

**ZÁPADOČESKÁ UNIVERZITA V PLZNI**  
**FAKULTA STROJNÍ**

Studijní program: B0715A270013 – Strojní inženýrství  
Studijní specializace: Průmyslové inženýrství a management

**BAKALÁŘSKÁ PRÁCE**

Vybrané technické příklady v Power Query

Autor: Tomáš Matějka  
Vedoucí práce: Doc. Ing. Pavel Raška, Ph.D.

Akademický rok 2023/2024

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Fakulta strojní

Akademický rok: 2023/2024

# ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Tomáš MATĚJKA**  
Osobní číslo: **S23B0240P**  
Studijní program: **B0715A270013 Strojní inženýrství**  
Specializace: **Průmyslové inženýrství a management**  
Téma práce: **Vybrané technické příklady v Power Query**  
Zadávací katedra: **Katedra průmyslového inženýrství a managementu**

## Zásady pro vypracování

- Úvod do řešené problematiky (popis příkladů z technické praxe)
- Analýza současného stavu (popis prostředí Power Query a jeho funkčnosti)
- Analýza řešení (analýza příkladů z technické praxe a možnosti jejich řešení pomocí Power Query)
- Implementace řešení (tvorba příkladů pomocí Power Query)
- Vyhodnocení a závěr

Rozsah bakalářské práce: **40 až 60 stran**  
Rozsah grafických prací: **–**  
Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. CLARK, Dan, Beginning Microsoft Power Bi: A Practical Guide to Self-Service Data Analytics. APress, 2020. ISBN 9781484256190.
2. ESCOBAR, Miguel, PULS, Ken, Master Your Data with Excel and Power BI. Holy Macro! Books, 2021. ISBN 9781615470587.
3. SEAMARK, Philip, Beginning DAX with Power BI. APress, 2018. ISBN 1484234766.
4. ALEXANDER, Michael, Excel Power Pivot & Power Query For Dummies. John Wiley & Sons Inc, 2022. ISBN 9781119844488.
5. FERRARI, Alberto, RUSSO, Marco, Analyzing Data with Power BI and Power Pivot for Excel. Microsoft Press, 2017. ISBN 9781509302765.

Vedoucí bakalářské práce: **Doc. Ing. Pavel Raška, Ph.D.**  
Katedra průmyslového inženýrství a managementu

Konzultant bakalářské práce: **Doc. Ing. Pavel Kopeček, CSc.**  
Katedra průmyslového inženýrství a managementu

Datum zadání bakalářské práce: **16. října 2023**  
Termín odevzdání bakalářské práce: **24. května 2024**

L.S.

---

**Doc. Ing. Vladimír Duchek, Ph.D.**  
děkan

---

**Doc. Ing. Michal Šimon, Ph.D.**  
vedoucí katedry

## **Prohlášení o autorství**

Předkládám tímto k posouzení a obhajobě bakalářskou práci zpracovanou na závěr studia na Fakultě strojní Západočeské univerzity v Plzni.

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této bakalářské práce.

V Plzni dne: .....

.....

podpis autora

## **Poděkování**

Tímto bych chtěl poděkovat mému vedoucímu bakalářské práce panu Doc. Ing. Pavlu Raškovi, Ph.D., za cenné rady, skvělý přístup a ochotu pomoc s řešením problémů, které se během psaní práce objevily. Dále bych chtěl poděkovat panu Doc. Ing. Pavlu Kopečkovi, CSc., především za rady ohledně práce s kusovníkovými hierarchiemi.

## ANOTAČNÍ LIST BAKALÁŘSKÉ PRÁCE

<b>AUTOR</b>	<b>Příjmení</b> Matějka	<b>Jméno</b> Tomáš	
<b>STUDIJNÍ PROGRAM</b>	B0715A270013 – Strojní inženýrství		
<b>VEDOUcí PRÁCE</b>	<b>Příjmení (včetně titulů)</b> Doc. Ing. Raška, Ph.D.	<b>Jméno</b> Pavel	
<b>PRACOVÍŠTĚ</b>	ZČU - FST - KPV		
<b>DRUH PRÁCE</b>	<b>DIPLOMOVÁ</b>	<b>BAKALÁŘSKÁ</b>	<b>Nehodící se škrtněte</b>
<b>NÁZEV PRÁCE</b>	Vybrané technické příklady v Power Query		

<b>FAKULTA</b>	strojní	<b>KATEDRA</b>	KPV	<b>ROK ODEVZD.</b>	2024
----------------	---------	----------------	-----	--------------------	------

### POČET STRAN (A4 a ekvivalentů A4)

<b>CELKEM</b>	68	<b>TEXTOVÁ ČÁST</b>	68	<b>GRAFICKÁ ČÁST</b>	0
---------------	----	---------------------	----	----------------------	---

<b>STRUČNÝ POPIS</b>	Bakalářská práce se zaměřuje na nástroje BI v rámci aplikace Excel a na jejich využití v praxi. Cílem práce je vytvořit sadu technických příkladů, které by sloužily jako podpůrný materiál ke studiu předmětu KPV/PVT2.
<b>KLÍČOVÁ SLOVA</b>	Excel, Power Query, Power Pivot, kusovník, datový model, úprava dat, vizualizace dat

## SUMMARY OF BACHELOR SHEET

<b>AUTHOR</b>	Surname Matějka	Name Tomáš	
<b>STUDY PROGRAMME</b>	B0715A270013 – Strojní inženýrství		
<b>SUPERVISOR</b>	Surname (Inclusive of Degrees) Doc. Ing. Raška, Ph.D.	Name Pavel	
<b>INSTITUTION</b>	ZČU - FST - KPV		
<b>TYPE OF WORK</b>	<del>DIPLOMA</del>	<b>BACHELOR</b>	Delete when not applicable
<b>TITLE OF THE WORK</b>	Technical examples in Power Query		

<b>FACULTY</b>	Mechanical Engineering	<b>DEPARTMENT</b>	Industrial Engineering	<b>SUBMITTED IN</b>	2024
----------------	------------------------	-------------------	------------------------	---------------------	------

### NUMBER OF PAGES (A4 and eq. A4)

<b>TOTALLY</b>	68	<b>TEXT PART</b>	68	<b>GRAPHICAL PART</b>	0
----------------	----	------------------	----	-----------------------	---

<b>BRIEF DESCRIPTION</b>	The bachelor thesis focuses on BI tools within Excel and their use in practice. The aim of the thesis is to create a set of technical examples that would serve as support material for the study of the subject KPV/PVT2.
<b>KEY WORDS</b>	Excel, Power Query, Power Pivot, kusovník, datový model, úprava dat, vizualizace dat

## Obsah

Přehled použitých zkratk a symbolů.....	8
Seznam obrázků .....	9
Seznam tabulek a grafů .....	10
1 Úvod do problematiky.....	11
2 Business Intelligence.....	12
2.1 Podstata Business Intelligence .....	12
2.1.1 Shromažďování a transformace dat.....	12
2.1.2 Analýza a průzkum dat.....	12
2.1.3 Vizualizace dat .....	13
2.1.4 Proces rozhodování .....	13
2.2 Příklady hlavních komponent BI.....	13
2.2.1 ETL.....	14
2.2.2 OLAP databáze .....	14
2.2.3 Datový sklad (DWH) .....	15
2.2.4 Datové tržiště (DMA).....	15
2.2.5 Klientské aplikace .....	15
2.2.6 Dočasné uložení dat (DSA).....	16
2.2.7 Dolování dat .....	16
2.2.8 Nástroje pro správu metadat.....	16
2.3 BI Softwary .....	17
3 Seznámení s prostředím Power Query .....	18
3.1 Power Query pro Excel .....	18
3.2 Získání a import dat.....	19
4 Seznámení s prostředím Power Pivot.....	22
4.1 Programovací jazyk DAX .....	22
4.2 Datový model .....	22
4.3 Power Pivot pro Excel.....	23
5 Představení technických příkladů.....	25
6 Příklad 1: Úprava dat v Power Query .....	26
6.1 Dotaz Položky .....	26
6.1.1 Načtení dat ze souboru Excel do prostředí Power Query.....	26
6.1.2 Úprava dotazu Položky v prostředí Power Query .....	27
6.2 Dotaz Pracoviště.....	32
6.3 Dotaz Časy operací.....	34
6.3.1 Načtení dat ze souboru CSV .....	34
6.3.2 Filtr dat .....	35



6.4	Dotaz Objednávky.....	36
6.5	Dotaz Zákazník .....	37
6.6	Spravování Dotazů a Připojení.....	37
7	Příklad 2: Úprava dat v Power Pivotu.....	40
7.1	Nastíněný postup příkladu.....	40
7.2	Import dat .....	42
7.3	Tvorba relací .....	43
7.4	Tvorba počítaných sloupců pomocí DAX.....	44
7.4.1	Cena jedné operace.....	44
7.4.2	Výrobní náklady .....	46
7.4.3	Celkové náklady .....	46
7.4.4	Celkové náklady na kolo .....	47
7.4.5	Vytvoření sloupce kusovníkové hierarchie .....	47
7.4.6	Cena položky.....	49
7.4.7	Zisk z prodeje .....	51
7.4.8	Funkce datum a čas .....	52
7.5	Výpočet míry.....	53
7.5.1	Průměrná cena objednávky .....	53
7.5.2	Celkový prodej .....	54
7.5.3	Počet kusů .....	54
8	Příklad 3: Tvorba kontingenčních tabulek a grafů.....	55
8.1	Zisk za jednotlivá období .....	55
8.1.1	Kontingenční tabulka Zisk za jednotlivá období.....	55
8.1.2	Kontingenční graf Zisk za jednotlivá období.....	57
8.2	Prodeje po jednotlivých značkách.....	58
8.2.1	Kontingenční tabulka Průměrná cena objednávky.....	58
8.2.2	Kontingenční graf Průměrná cena objednávky .....	59
8.2.3	Kontingenční tabulka Celkového prodeje a množství.....	59
8.2.4	Kontingenční graf Celkové prodeje a množství.....	60
8.3	Přehled objednávek a tržeb za určité období.....	61
	Závěr.....	63
	Seznam použité literatury .....	64
	Přílohy .....	65

## **Přehled použitých zkratk a symbolů**

BI	Business Intelligence
DAX	Data Analysis Expressions
DMA	Data mart
DSA	Data Staging Area
DSS	Decision Support Systems
DWH	Data warehouse
EIS	Executive Information Systems
ETL	Extraction, Transformation and Loading
MDX	MultiDimensional Expressions
MOLAP	Multidimensional Online Analytical Processing
OLAP	Online Analytical Processing
ROLAP	Relational Online Analytical Processing
SQL	Structured Query Language
SSIS	SQL Server Integration Services

## Seznam obrázků

Obrázek 1: Obecká koncepce architektury BI [5].....	13
Obrázek 2: OLAP Cube [7].....	14
Obrázek 3: Umístění Power Query v rámci Excelu .....	18
Obrázek 4: Nabídka datových zdrojů.....	19
Obrázek 5: Okno navigátoru .....	20
Obrázek 6: Nabídka importu dat .....	21
Obrázek 7: Umístění Power Pivotu v rámci Excelu.....	23
Obrázek 8: Import dat do Power Pivotu.....	24
Obrázek 9: Okno Navigátoru – dotaz Položky.....	26
Obrázek 10: Prostředí Power Query – dotaz Položky .....	27
Obrázek 11: Informace o výskytu prázdných řádků .....	28
Obrázek 12: Nahrazení hodnot – dotaz Položky .....	28
Obrázek 13: Rozdělení sloupce oddělovačem .....	29
Obrázek 14: Nástroj oříznutí .....	30
Obrázek 15: Nastavení datového typu sloupce – dotaz Položky .....	31
Obrázek 16: Přejmenované sloupce – dotaz Položky .....	31
Obrázek 17: Použitý postup – dotaz Položky.....	31
Obrázek 18: Import dat do Excelu .....	32
Obrázek 19: Import dat ze souboru Text/CSV.....	33
Obrázek 20: Okno navigátoru – dotaz Pracoviště.....	33
Obrázek 21: Prostředí Power Query – dotaz Pracoviště .....	34
Obrázek 22: Dotaz pracoviště výsledná podoba .....	34
Obrázek 23: Filtr sloupce .....	35
Obrázek 24: Výsledek aplikace filtru sloupce.....	35
Obrázek 25: Přejmenování sloupců – dotaz Objednávky .....	36
Obrázek 26: Změna datového typu podle národního prostředí .....	36
Obrázek 27: Rozdělení sloupců oddělovačem – dotaz Zákazník.....	37
Obrázek 28: Přejmenování sloupců – dotaz Zákazník.....	37
Obrázek 29: Sekce Dotazy a připojení.....	38
Obrázek 30: Lišta rychlého přístupu Dotazy a připojení .....	38
Obrázek 31: Definování hierarchie .....	40
Obrázek 32: Schematický diagram jízdního kola – hierarchické úrovně.....	41
Obrázek 33: Import dat do prostředí Power Pivotu.....	42
Obrázek 34: Zobrazení diagramu.....	43

Obrázek 35: Okno Úprava relace .....	44
Obrázek 36: Finální podoba dotazu ČasyOperací.....	45
Obrázek 37: Práce s funkcemi v DAX .....	45
Obrázek 38: Chyba funkce PATH .....	48
Obrázek 39: Nahrazení hodnoty (Power Query).....	48
Obrázek 40: Výsledek použití funkce PATH a PATHITEM.....	49
Obrázek 41: Ilustrační schematický diagram hierarchické kusovníkové úrovně.....	50
Obrázek 42: Výpočet nové míry .....	54
Obrázek 43: Vytvoření kontingenční tabulky .....	55
Obrázek 44: Lišta Seznam polí .....	56
Obrázek 45: Hodnoty pro kontingenční tabulku Zisk za jednotlivé období .....	56
Obrázek 46: Kontingenční tabulka Zisk za jednotlivá období.....	57
Obrázek 47: Časová osa Zisk za jednotlivá období .....	57
Obrázek 48: Kontingenční tabulka Průměrná cena objednávky .....	59
Obrázek 49: Kontingenční tabulka Celkové prodeje a množství.....	60
Obrázek 50: Nastavení datových řad pro kombinovaný graf.....	61
Obrázek 51: Průřez Kontingenčního grafu.....	62
Obrázek 52: Ukázka prostředí Power Query.....	65
Obrázek 53: Ukázka prostředí Power Pivot .....	66

## **Seznam tabulek a grafů**

Graf 1: Kontingenční graf Zisk za jednotlivá období .....	58
Graf 2: Kontingenční graf Průměrná cena objednávky.....	59
Graf 3: Kontingenční graf Celkové prodeje a množství .....	60
Graf 4: Kontingenční graf Přehled objednávek a tržeb za určité období.....	62

## 1 Úvod do problematiky

Současná společnost je obklopena daty kolem do kola. Takřka každý člověk se s nimi alespoň jednou za život setká. Není tedy divu, že přibývá lidí, kteří s nimi pracují na denní bázi. Společnosti kladou vyšší důraz na znalosti a schopnosti zaměstnanců právě v oblasti zpracování a implementace dat. I v důsledku toho dochází k vývoji nových softwarů a nástrojů, které současnou práci s daty usnadňují a přidávají další možnosti v oblasti zpracování dat. Jedním z příkladů jsou aplikace Self-Care BI, které uživateli umožňují z nahraných dat vytvořit např. vizualizace sloužící k určování dalších trendů, a to bez výrazných znalostí programování. Bez použití BI nástrojů by bylo nutné k dosažení stejných výsledků svěřit práci do rukou odborníkům z IT odvětví. Na trhu existuje řada BI programů, tato práce se blíže zaměří pouze na použití nástrojů od společnosti Microsoft, tedy konkrétně Power Query a Power Pivot. Oba zmíněné nástroje jsou k dispozici v rámci aplikace Excel.

Výsledkem této bakalářské práce, by měl být balíček příkladů z technické praxe, na kterých je vysvětlen postup a způsob použití zmíněných nástrojů Excelu. Konkrétně se tedy jedná o jeden komplexní příklad sestávající se z jednotlivých chronologicky řazených pod příkladů. V úvodu této práce je čtenář seznámen se samotným prostředím Power Query a Power Pivot, kde se prostředí nacházejí a jaké jsou jejich benefity oproti klasickému zpracování dat. Zmíněné informace budou dále sloužit jako odrazový můstek pro samotnou práci s těmito nástroji.

Zdrojem dat pro jednotlivé příklady bude databáze fiktivního výrobce kol, která byla vytvořena přímo pro tuto práci. Vstupní data obsahují informace o jednotlivých dílech kola, prodeji společnosti, zákaznících atd. Pro lepší orientaci v datech budou příklady realizovány na menším objemu dat. Postupy, popsané v této práci, však zůstávají stejné i pro rozsáhlejší databáze, kde se přínos BI softwarů již výrazně podepíše na výsledku práce.

Na základě prostudování tohoto dokumentu by čtenář měl pochopit základní principy a načerpat informace potřebné k dalšímu vzdělávání v rámci této problematiky. Zároveň lze znalosti z prostředí Excelu přenést i do dalších aplikací. Jednou z takových je např. Power BI od společnosti Microsoft, která se taktéž opírá o zmíněné nástroje. Základní principy obou aplikací jsou téměř identické a liší se převážně uživatelským rozhraním. Tento balíček dále může posloužit jako jeden z podpůrných materiálů pro výuku/studium předmětu KPV/PVT2.

## 2 Business Intelligence

Pojem Business Intelligence (dále jen BI) byl poprvé zmíněn v rámci článku *Cyclopædia of Commercial and Business Anecdotes* který byl publikován v roce 1865. V článku užil autor, Richard Millar Devens, tento termín k charakterizaci toho, jakým způsobem bankéř sir Henry Furnese využíval informace. Furnese si informace pečlivě shromažďoval a poté podle nich jednal před konkurencí, což mu přinášelo výhodu. [1]

O několik desítek let později v roce 1958, německý vědec Hans Peter Luhn ze společnosti IBM publikoval článek *A Business Intelligence System*, který popisuje potenciál využití BI technologie k shromažďování obchodních informací. V článku Luhn diskutuje např. o potřebě efektivního zpracování informací v podnikání, nebo o významu identifikace klíčových slov a klíčových frází v textu pro lepší porozumění obsahu. Koncepty uvedené v tomto článku se později staly klíčovými pro oblast BI. [2]

BI, jak je dnes chápána, se vyvinula ze systémů pro podporu rozhodování, které se začali používat během 60. let minulého století. V 80. letech minulého století došlo k velkému rozmachu BI. Byl vyvinut první systém pro správu databází DSS, rovněž se v této době poprvé objevují systémy jako OLAP, EIS, nebo například Datové sklady. Během této doby přicházejí na trh i první komerční produkty tohoto typu. [3]

### 2.1 Podstata Business Intelligence

Pojem BI lze chápat jako technologický proces analýzy dat a získávání potřebných informací na základě kterých se vytvářejí strategická rozhodnutí. Nástroje BI pracují s historickými a aktuálními daty na základě nichž prezentují zjištění v přehledných vizualizacích a tabulkách. [4]

Další z definic udává, že BI lze chápat jako sadu procesů, aplikací a technologií, jejichž cílem je účinně a účelně podporovat rozhodovací procesy ve firmě. Tyto procesy podporují činnosti podniků (analytické a plánovací) a jejich základy stojí na principech multidimenzionálních pohledů na data podniku. [5]

Úkolem BI je usilovat o optimalizaci obchodních rozhodnutí, která společností pomůžou zvýšit příjmy, zlepšit efektivitu provozu a získat konkurenční výhody oproti obchodním rivalům. K dosažení tohoto cíle BI využívá kombinace analytických nástrojů, nástrojů pro správu dat a účetnictví a různých metod pro správu a analýzu podnikových dat. [4]

Celý proces transformace nezpracovaných dat do utříděných srozumitelných přehledů lze shrnout do čtyř klíčových kroků, podle kterých BI postupuje.

1. **Shromažďování a transformace dat**
2. **Analýza dat**
3. **Vizualizace dat**
4. **Proces rozhodování** [4]

#### 2.1.1 Shromažďování a transformace dat

Obvyklým postupem nástrojů pro BI je využití metody extrakce, transformace a načítání k shlukování dat z různých zdrojů. Takto získaná data jsou následně transformována a upravena do patřičné podoby před samotným uložením do centrální paměti. To usnadňuje následné analýzy a tvorbu dotazů, protože jsou data uložena do komplexního souboru. [4],[6]

#### 2.1.2 Analýza a průzkum dat

V této fázi dochází k tzv. dolování dat neboli získávání dat pro potřebné analýzy. Nástroje BI obvykle používají řadu automatizací k urychlení procesu získání hledaných hodnot. Samotná

analýza dat probíhá za pomoci analytických a modelovacích nástrojů jako např. průzkumné, statistické, popisné a prediktivní nástroje. [4],[6]

### 2.1.3 Vizualizace dat

Nástroje BI umožňují z utříděných dat vytvářet vizualizace v podobě grafů, kontingenčních tabulek nebo např. map. Vizualizace přehledně demonstrují podniková data a umožňují jejich snadné pochopení a sdílení. Na základě toho lze např. porovnat aktuální data s daty z let minulých a predikovat tak budoucí vývoj. [4],[6]

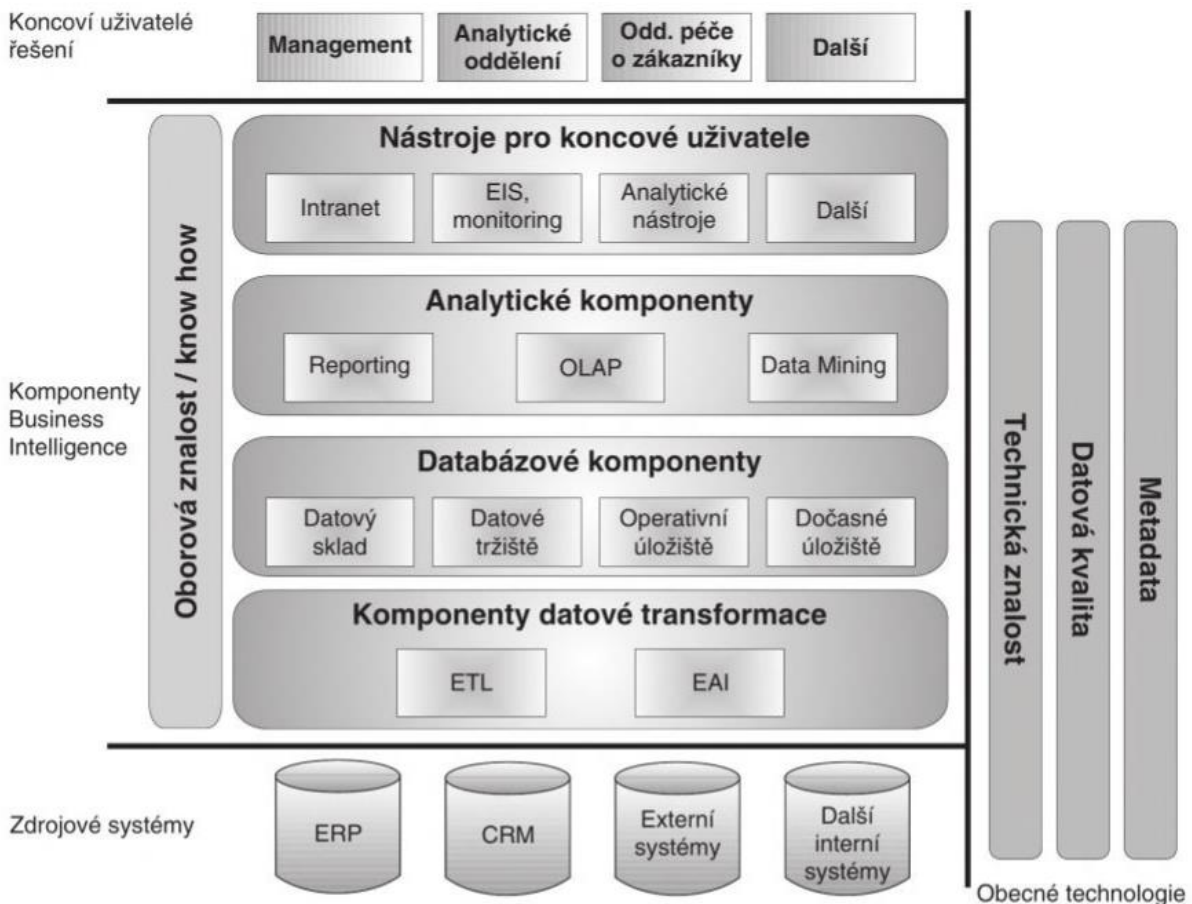
### 2.1.4 Proces rozhodování

Na základě vizualizace aktuálních a historických dat v kontextu s podnikovými aktivitami lze vytvářet rozhodnutí o budoucích záměrech. BI umožňuje upravovat a aktualizovat data v reálném čase a vytvářet dlouhodobé strategické změny. Na základě toho lze eliminovat neefektivitu, přizpůsobovat se změnám na trhu a řešit problémy zákazníků. [4],[6]

## 2.2 Příklady hlavních komponent BI

V této kapitole budou představeny jednotlivé komponenty BI a jejich uspořádání. V praxi lze narazit na nespočet různých variant řešení BI, a to od nejjednodušších až po rozsáhlá komplexní řešení. Každé řešení kombinuje různé technologie a produkty v rámci BI na základě potřeb daného podniku. Z toho důvodu budou představeny pouze základní, často používané komponenty BI.[5], [7]

Obecně lze koncepci architektury BI rozdělit do několika vrstev, které obsahují jednotlivé komponenty BI (viz Obrázek 1).



Obrázek 1: Obecná koncepce architektury BI [5]

### 2.2.1 ETL

V české literatuře taktéž známo pod pojmem *datová pumpa*. Jak sám název napovídá, úkolem ETL je získat data ze zdrojových systémů (*Extraction*), ta následně přetvořit do požadované formy a uspořádat (*Transformation*). Na závěr takto utříděná data nahrát do datových struktur (*Loading*). Tyto nástroje umožňují nahrání dat z jednoho, nebo i více datových zdrojů. S rozvojem analytických systémů, nabraly ETL systémy na důležitosti pro potřebu zajištění přenosu dat mezi jednotlivými systémy aplikací v rámci různých databázových prostředí. Nástroje ETL pracují s daty v tzv. dávkovém režimu neboli data jsou přenášena po jistých časových intervalech, řádově dnů či týdnů. [7], [5]

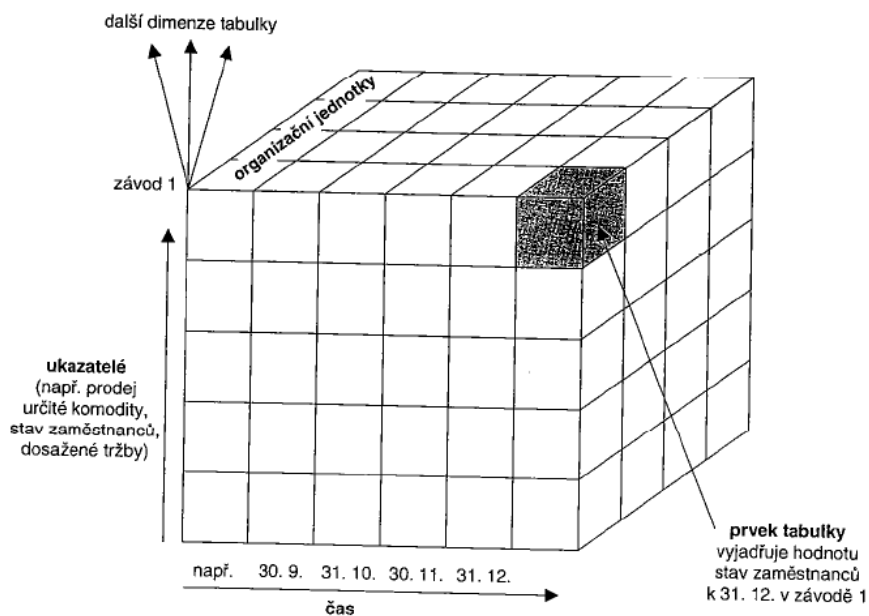
### 2.2.2 OLAP databáze

Pojem OLAP se poprvé objevil v 80. letech minulého století, jedná se o představitele multidimenzionální databáze, kde se pohlíží na data z vícero hledisek a jsou kladeny požadavky na optimalizované fyzické ukládání dat.

Jedná se o jednu z technologií v rámci BI založenou na multidimenzionálních databázích, která umožňuje analyzovat velké objemy dat a složité datové sestavy. Pracuje na principu multidimenzionální tabulky, která dokáže rychle a přehledně měnit jednotlivé dimenze, respektive pohledy na vytvářenou realitu.

OLAP databáze jsou založeny na principu stejnojmenné technologie, která ukládá data do multidimenzionálních databází. OLAP databáze se skládají z jedné nebo více OLAP kostek, které obsahují již předem rozřazená data podle jednotlivých hierarchických struktur. OLAP technologie se vyskytuje ve více variantách, nejčastěji však jako MOLAP anebo ROLAP.

- MOLAP – jedná se o variantu OLAP, která pracuje přímo s vícerozměrnou krychlí (*OLAP cube*). Pro většinu aplikací je tato varianta nejrychlejším a nejpraktičtějším typem analýzy multidimenzionálních dat.
- ROLAP – tato varianta pracuje přímo s daty v relačních tabulkách bez toho, aby došlo nejprve k organizaci dat do OLAP krychle. ROLAP najde své uplatnění především tehdy, je-li potřeba pracovat s velkým objemem dat ale již nejsou kladeny příliš vysoké nároky na výkon a flexibilitu.



Obrázek 2: OLAP Cube [7]



Základem multidimenzionálních databází je multidimenzionální tabulka nebo tzv. *OLAP cube*, viz *Obrázek 2*, která umožňuje měnit jednotlivé dimenze.

Z obrázku vyplývá, že standardními dvěma dimenzemi jsou ukazatelé ekonomické proměnné a čas. Zbylé dimenze jsou pro jednotlivé kostky definovány dle potřeby uživatele např.: zákazník, oblast, objednávky, zásoby, dodavatel atd. Jednotlivé dimenze jsou poté tvořeny z prvků dimenzí jako např.: konkrétní oblast, závody, zákazník, dodavatel apod. Prvky dimenzí jsou zpravidla uspořádány do skupin podle hierarchie. Nástroje BI poté zajišťují shromáždění hodnot na základě právě takto definovaných hierarchií. [7], [8]

### 2.2.3 Datový sklad (DWH)

Datový sklad je jednou z technologií uplatňovaných v rámci BI. Datové sklady řeší problém s neustále narůstajícím objemem používaných dat a potřebou pracovat v analytických aplikacích. Na rozdíl od již zmíněných MOLAP databází, datové sklady ukládají data do relačních databází. Data jsou pak dále seřazena podle STAR nebo SNOWFLAKE schémat anebo zůstávají uložené v relačních databázích dle klasického uspořádání. Ačkoliv tato data nejsou uložena primárně pro použití multidimenzionálních aplikací, jsou vyčleněna z produkčních databází a připravená k analýze. Vlastní multidimenzionální aplikace pak spojuje klasický datový sklad s multidimenzionálními aplikacemi (MOLAP, ROLAP), kde jsou data již uložena pouze ve zjednodušené formě, agregovaná podle jednotlivých dimenzí. [7], [5]

### 2.2.4 Datové tržiště (DMA)

Datová tržiště neboli Data Marty, jsou v principu to samé jako datové sklady. Od datových skladů se odlišují tím, že mají omezený přístup, jsou tedy dostupná jen pro určité skupiny (pobočky, oddělení, odvětví...). Principem je tak vytvoření decentralizovaných datových skladů, které budou postupně sloučeny do celopodnikového řešení. V některých případech Datová tržiště slouží i poté, co je celopodnikový datový sklad vytvořen, jako mezistupeň při transformaci dat z produkčních databází. Jedná se o specializovaný datový sklad orientovaný na řešení konkrétních problémů v daném okruhu uživatelů. [7], [5]

### 2.2.5 Klientské aplikace

V rámci BI lze rozdělit klientské aplikace do dvou základních skupin, a to Reporting a Manažerské aplikace. [7]

#### Reporting

Pod pojmem reporting si lze představit operace spojené s dotazováním se do databází prostřednictvím rozhraní těchto databází, jako např. SQL příkazy. V praxi se dále rozlišuje tzv. *standardní reporting* a *ad hoc reporting*. V rámci standardního reportingu se pracuje s předpřipravenými dotazy, které se spouštějí v určitých časových periodách. Zatím co u ad hoc reportingu uživatelé explicitně vytvářejí konkrétní dotazy, které jsou pak jednorázově formulovány na jednotlivé databáze. [7]

#### Manažerské aplikace (EIS)

Manažerské aplikace, nebo také EIS je jeden z typů klientských aplikací BI. Tyto aplikace v sobě integrují podstatné datové zdroje, s tím souvisejí specifické požadavky na prezentaci informací a jejich dostupnost pro vedoucí pracovníky a analytiku společnosti.

Výhody manažerských aplikací:

- jsou navrhovány především pro práci s manažerskými daty a pro podnikové použití
- Jsou uživatelsky přístupné a zajišťují, že data na výstupu budou mít vysokou vypovídací hodnotu
- přistupují k jednotlivým datům stejně jako k datům agregovaným

- disponují nástroji pro on-line analýzy jako jsou např. *drill up, drill down, identifikace výjimek*

Manažerské aplikace mohou být implementovány v různých technologických prostředích. S ohledem na potřebnou flexibilitu jsou obvykle vytvářena v rámci klientských aplikací namísto multidimenzionálních databází (MOLAP, ROLAP...). Vytváření a provoz těchto klientských manažerských aplikací se uskutečňuje s využitím různých prostředků, zejména:

- specializovaných produktů pro manažerské aplikace jako jsou např: ProClarity, Oracle Discoverer atd.
- kancelářských produktů jako je Excel nebo např. Access
- aplikací, které používají speciální programovací jazyky, např. MDX

[7]

### 2.2.6 Dočasné úložiště dat (DSA)

DSA je úložiště na které se ukládají netransformovaná data z výrobních systémů. Své využití nachází zejména u přetížených produkčních systémů. Data, se kterými je třeba pracovat, se nahrají z produkčního systému do dočasného úložiště, aby zbytečně nepřetěžovala paměť a poté se přesunou kam je třeba. [7]

### 2.2.7 Dolování dat

Dolování dat, nebo také Data mining, lze popsat jako proces automatického získávání užitečných informací z velkého datového úložiště za pomoci různých algoritmů. Technika dolování dat se využívá k prozkoumávání rozsáhlých databázových oblastí s cílem identifikovat nové a užitečné vzory, které by jinak mohly zůstat neobjeveny. Jednou z funkcí, které umožňuje je schopnost předpovídat budoucí vývoj na základě pozorovaných dat. Je třeba si však uvědomit, že dolování dat není vhodné pro všechny úlohy týkající se získávání informací, například pro vyhledávání jednotlivých záznamů pomocí systému správy databází, nebo obecně úkoly spojené s vyhledáváním informací. Tyto techniky byly vyvinuty s cílem vylepšit systémy získávání dat. [7], [5]

Existuje mnoho nástrojů pro dolování dat, zahrnující ty, které jsou navrženy pro odborníky s dovednostmi v oblasti statistiky, stejně jako ty, které jsou vhodné pro řídicí pracovníky. Nicméně cíl dolování dat zůstává podobný většině úloh v oblasti BI, což je poskytování strategických informací různým manažerům v organizaci. To, čím se liší dolování dat od jiných statistických nástrojů, je jeho zaměření na různé uživatelské profily. Statistické úlohy spojené s dolováním dat jsou prováděny automaticky na základě přesně definovaných algoritmů, což umožňuje, aby i manažeři bez specializovaných znalostí v oblasti statistiky byli cílovými uživateli těchto nástrojů.[7], [5]

Příklady technik používaných k data miningu:

- **Rozhodovací stromy** – zobrazení dat do podoby stromu, prediktivní model, každý kořen představuje kritérium rozdělení dat do dalších větví
- **Neuronové sítě** – tvorba prediktivních modelů
- **Generické algoritmy** – uplatňují princip evoluce na řešené úlohy
- **Clustering a klasifikace** – data členěna do skupin podle podobností, v rámci skupin se definují klíčové atributy v podobě klasifikačních kritérií [7], [9]

### 2.2.8 Nástroje pro správu metadat

Tyto nástroje začali nabírat na důležitosti s implementací BI. Metadata lze popsat jako data o datech, konkrétně v rámci BI řešení dokumentují konkrétní použité informační systémy podniku. Metadata tedy sbírají data a popisují veškeré informační systémy, které jsou použity.

Z hlediska BI lze za metadata označit především datové modely, popisy funkcí, pravidel, reportů apod. [7]

## 2.3 BI Softwary

BI softwary tvoří skupinu počítačových aplikací, které zpracovávají a analyzují podniková data za účelem vytváření přehledů a porozumění dat. První prototypy těchto softwarů se objevily již během 80. a 90. let minulého století. Společnosti jako *IBM* nebo *Siebel* byly jedny z prvních které pracovali na jejich vývoji. Po roce 2001 začaly přicházet na trh první BI softwary které vypadaly tak, jak je známe dnes. Pro většinu podnikatelů však tyto softwary byly příliš drahé a náročné na obsluhu. Byly vzneseny požadavky na to, aby uživatelé mohli extrahovat data nezávisle ze svých systémů a ukládat je na dostupná místa. Také aby všichni uživatelé měli přístup k nástrojům potřebných pro tvorbu vizualizací a úpravě těchto dat. To se změnilo s příchodem cloudových uložišť v roce 2006, ta nabízejí správu dat v hostovaných uložištích, díky tomu klesla i cena softwarů a ty se zpřístupnili vícero uživatelům. S rozvojem cloudových uložišť přišel i rozmach BI softwarů. V rychlé době, kdy se pracuje se stále většími objemy dat a je třeba uchovávat více a více informací, vývojáři přicházejí s novými a účinnějšími nástroji. [10]

V dnešní době již používá BI softwary k práci s daty většina velkých i menších podniků, řada z nich je dostupná v zdarma v základních verzích anebo v rámci licenčních balíčků s ostatními aplikacemi. Níže uvedené softwary jsou pouze ukázkou některých z nejčastěji používaných

- **Tableau Software**
- **IBM Cognos Analytics**
- **QlikTech**
  - QlikView
  - QlikSense
- **SAP BusinessObjects**
- **Looker**
- **Domo**
- **Microsoft**
  - SQL Server Reporting Services
  - SQL Server Analysis Services
  - PerformancePoint Server 2007
  - Proclarity
  - Power Pivot
  - Power Query
  - Power BI

Tato práce je dále zaměřena na popis BI nástrojů nabízené společností Microsoft, respektive nástrojů Power Pivot a Power Query. [11], [10]

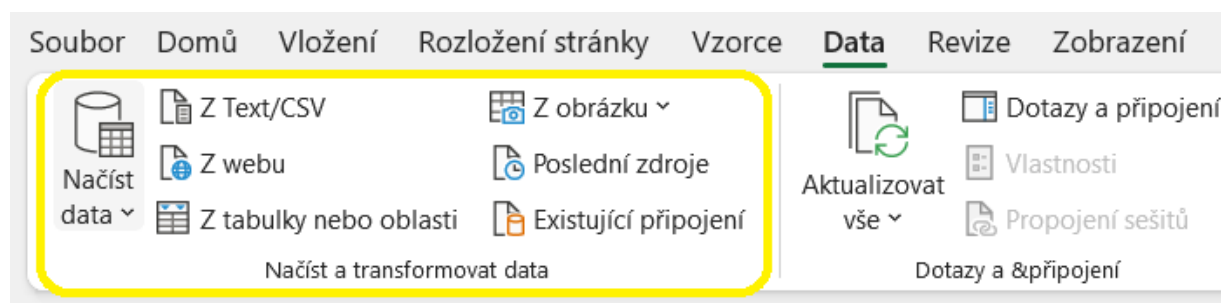
### 3 Seznámení s prostředím Power Query

Jedná se o nástroj pro transformaci a přípravu dat vyvinutý společností Microsoft. Smyslem jeho použití je zjednodušení a zrychlení práce s daty. Uživatel tak nutně nepotřebuje disponovat znalostmi z oboru IT. Nástroj Power Query přichází s přehledným grafickým rozhraním v rámci něž, je možno data jednoduše importovat (a to i z vícero zdrojů) a následně tato data transformovat do požadované podoby. V rámci prostředí Editoru Power Query probíhá primární příprava dat. Uživatelé zde mohou nahrát data z různých typů zdrojů a následně tato data třídit, selektovat, upravovat jejich vzhled anebo kupříkladu kombinovat data z jednotlivých zdrojů. Možnosti úpravy dat jsou stejné pro všechny typy zdrojů. Díky grafickému prostředí je použití editoru dostupné pro většinu uživatelů bez znalostí programovacích jazyků. Během práce s nástrojem Power Query jsou automaticky zaznamenávány veškeré provedené změny do kódu pomocí programovacího jazyka M. Aplikace toto vykonává bez vědomí uživatele, ten pouze zvolí z nabídky dostupných operací, jak mají být data upravena. Tento relativně nový jazyk se výrazně liší od klasických jazyků používaných v rámci prostředí Excelu. Uživatel může do samotného kódu nahlédnout za pomoci Rozšířeného editoru. Pokud uživatel nenalezne v rámci uživatelského rozhraní funkce a transformace uspokojující jeho potřeby, je vhodné použít právě zmíněný Rozšířený editor. Práce s ním však vyžaduje znalost jazyka M a předchozí zkušenost s používáním aplikace. [2], [13], [14]

Společnost Microsoft nabízí nástroj Power Query v rámci hned několika programů, jako např.: Excel, Power BI, Power Apps Azure Data Factory a mnoho dalších.

#### 3.1 Power Query pro Excel

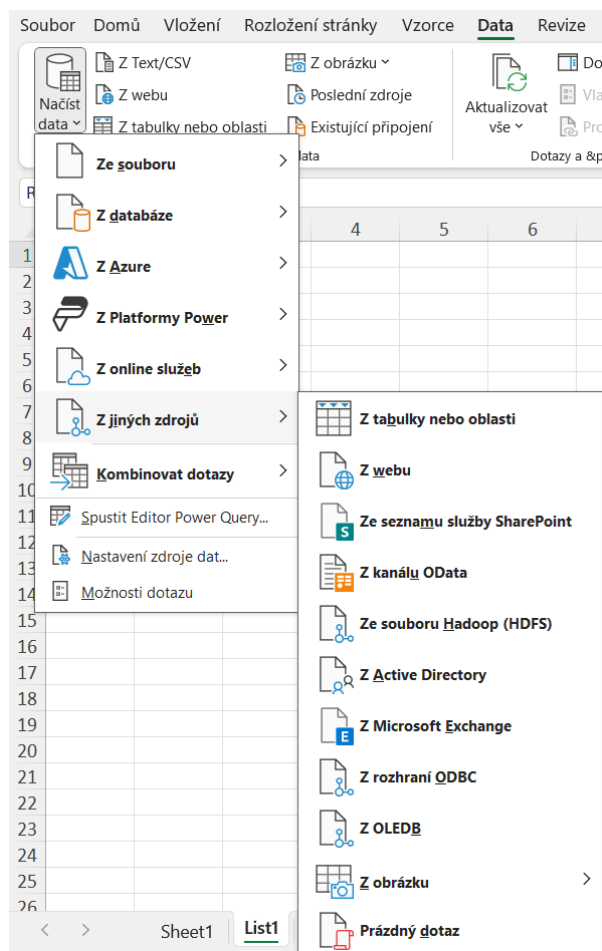
Power Query během své existence prošel řadou změn a aktualizací. Poprvé se objevil ve verzi Excel 2010, v této verzi, stejně jako ve verzi Excel 2013, není nástroj Power Query přímo součástí aplikace. Pro jeho použití v těchto verzích je nutné jej stáhnout a nainstalovat. Společnost Microsoft jej nabízí zdarma ke stažení na svých webových stránkách společně s návodem na jeho instalaci. Po jeho instalaci se objeví v samostatné kartě pod názvem *Power Query*. S nástupem verze Excel 2016 se nástroj Power Query dočkal řady změn. Power Query se stalo primárním prostředím pro import a čištění dat a nahradilo tak starší průvodce importem dat. Nově jej najdeme na kartě *Data* pod názvem *Načíst&Transformovat*, viz *Obrázek 3*. V rámci této karty nalezneme také vylepšené funkce importu dat a nové podokno *Dotazy a připojení*. Od roku 2019 je Power Query dostupný v rámci Excelu i pro uživatele operačního systému *Mac*. [13], [14]



Obrázek 3: Umístění Power Query v rámci Excelu

## 3.2 Získání a import dat

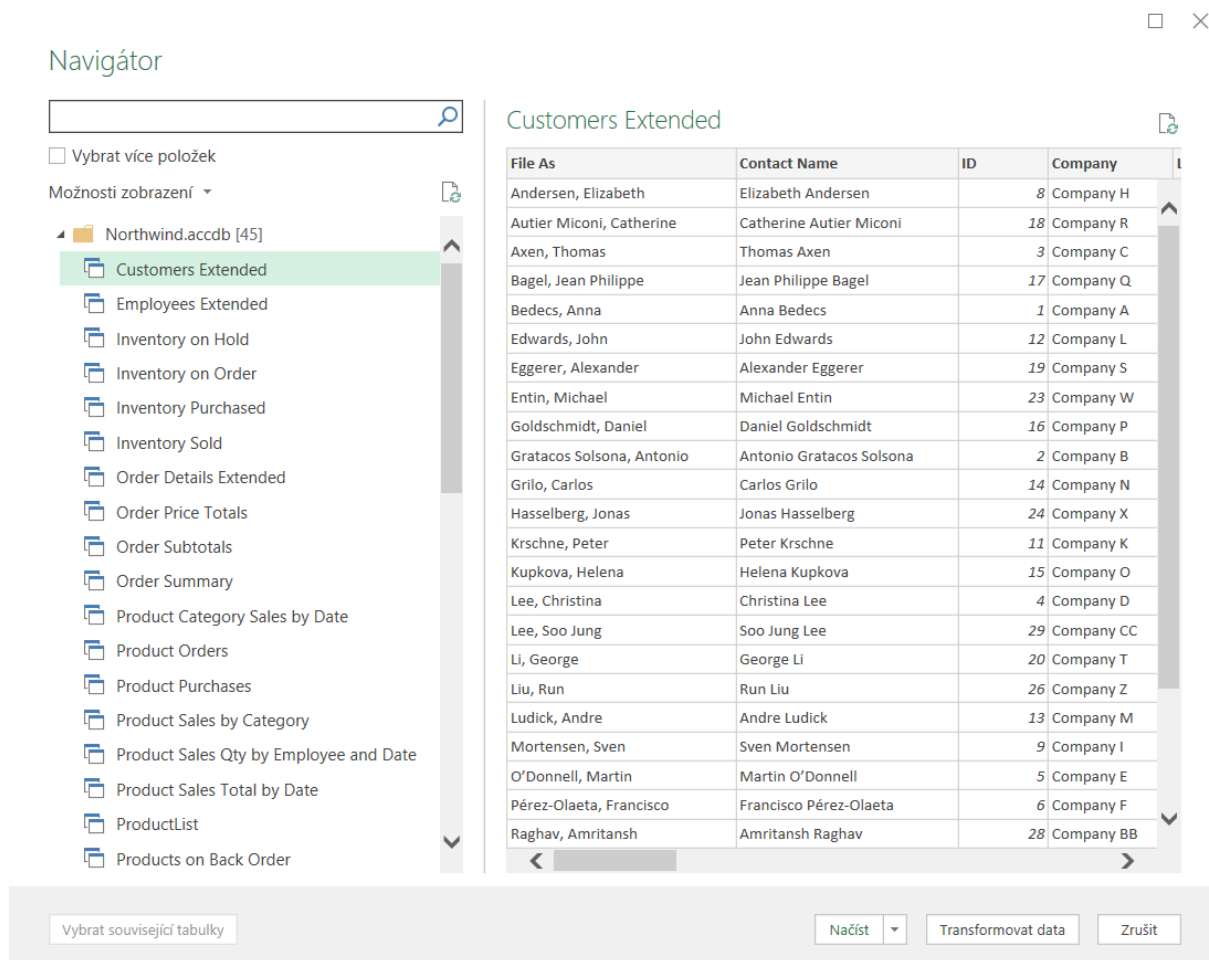
Pokud byste chtěli kombinovat a transformovat data z vícero zdrojů bez použití nástroje Power Query, stálo by to mnohé úsilí. Data by musela být spracována pomocí nástrojů, jako např. SSIS, které vyžadují širší znalost informatiky. Jednou z alternativ k tomuto způsobu je právě zmíněný self-service BI ETL nástroj Power Query. Pro nahrání dat do Excelu pomocí tohoto nástroje přejděte na kartu *Data* a rozklikněte nabídku *Načíst data*.



Obrázek 4: Nabídka datových zdrojů

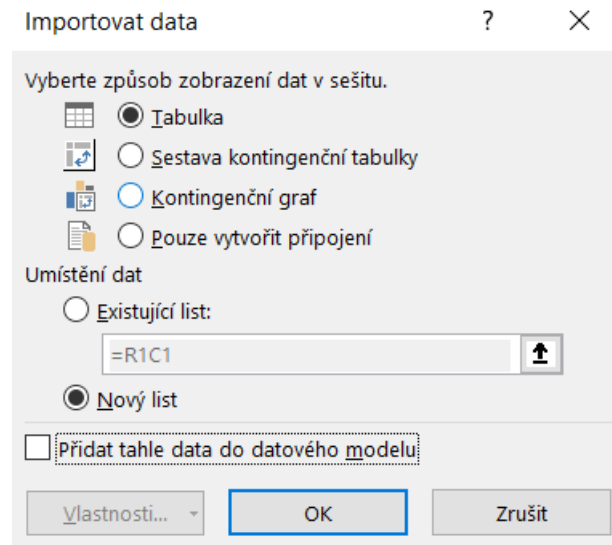
Zobrazí se seznam s různými datovými zdroji, odkud lze data přenést. Jak je patrné z obrázku (viz Obrázek 4), data lze získat například z databáze, různých aplikací, webu a spousty jiných zdrojů. Jednotlivé způsoby připojení ke zdrojům dat vyžadují různý způsob připojení. Například pokud chce uživatel nahrát data z webového zdroje, je zapotřebí znát jeho URL adresu, pomocí které se ke zdroji připojí. Jakmile dojde k připojení ke zdroji, spustí se okno editoru Power Query. Pokud jsou data přebírána ze zdroje obsahujícího více tabulek, před načtením samotného editoru se zobrazí okno navigátoru, viz Obrázek 5. V rámci navigátoru jsou vlevo ve sloupci zobrazeny veškeré dotazy, které byly nalezeny a napravo je pak zobrazen náhled aktuálně otevřeného dotazu. V levém sloupečku lze zvolit které konkrétní tabulky mají být vybrány. Pokud z náhledu vpravo je vidět že data nejsou utříděna, nebo nevyhovují zadaným požadavkům, lze je pomocí tlačítka *Transformovat* nahrát do prostředí Power Query. Zde se data dále upravují a třídí do požadované podoby. Pokud jsou již vstupní data upravena je možno je rovnou nahrát do Excelu prostřednictvím tlačítka *načíst* (viz. okno navigátoru).[12]

Při zvolení možnosti *Transformovat data* se uživatel přesune do okna Power Query (detailní pohled viz přílohy Obrázek 52). Na levé straně okna se nachází karta *Dotazy*, ve které se nachází všechny nahrané tabulky/dotazy mezi nimiž lze přepínat. Konkrétní úpravy se pak aplikují vždy pro jednotlivý dotaz. Na horní liště lze nalézt všemožné nástroje pro úpravu a práci s daty, k těm však podrobněji až u jednotlivých příkladů. Ve sloupci na pravé straně se postupně vypisuje postup provedených úprav, na jehož základě lze snadno nahlédnout do historie úprav a pozměnit je. Poté, co jsou data připravena k importu, zbývá jen postup uložit a načíst do Excelu. To se realizuje kliknutím na šipku u tlačítka *Zavřít a načíst* na kartě *Soubory* a zvolením možnosti *Zavřít a načíst do ....* Následně se zobrazí okno s možnostmi zobrazení dat v sešitu, viz Obrázek 6. [14], [12] [16]



Obrázek 5: Okno navigátoru

V dolní části okna se ještě nachází zaškrtnuté pole s dotazem, zdali uživatel chce data nahrát do datového modelu. Při zaškrtnutí tohoto pole se data nahrají rovnou do datového modelu v prostředí Power pivotu.



Obrázek 6: Nabídka importu dat

## 4 Seznámení s prostředím Power Pivot

Power pivot je jedním z hlavních nástrojů používaných v rámci Self Service Business intelligence společnosti Microsofts. Dalo by se říci, že je jakým si hnacím motorem. Prostředí Power Pivot umožňuje uživateli jednoduše modelovat a upravovat data do potřebné podoby a následně tato data analyzovat např. pomocí kontingenčních tabulek, nebo přehledných vizualizací. Pro úpravu dat Power Pivot využívá programovací jazyk DAX, pomocí něhož lze vytvářet podmínky a funkce. Jednou z velkých výhod doplňku Power Pivot je, schopnost plynule pracovat s rozsáhlým množstvím dat, řádově desítkami až stovkami milionů řádků. Za tím vším stojí *xVelocity In-memory Analytics Engine*. Jedním z klíčových rozdílů je způsob ukládání dat. Oproti klasickému ukládání dat do řádků, Power Pivot využívá systému ukládání dat do sloupců. Sloupcová databáze ukládá data do oddělených sloupců a pracuje pouze s daty v potřebných sloupcích, není tak nutné číst data ze všech sloupců. Díky tomu databáze výrazně šetří místo na uložení a umožňuje tak plynulou práci i s větším objemem dat. Také umožňuje rychlejší vyhledávání či aktualizaci dat v databázi. Při použití klasického ukládání dat do řádků zabere vyhledávání v databázi mnohem více času, poněvadž dochází k častému opakování stejných názvů, např. příjmení atd. Další bezprostřední výhodou tohoto enginu je způsob načítání dat. Data jsou nejprve načtena do RAM paměti počítače a následně potom jsou nad nimi prováděny dotazy. Výsledkem je rychlejší zpracování dat, a to především při použití 64-bitového operačního systému a aplikace. [12], [14]

Před tím, než jsou prováděny úkony s doplňkem Power Pivot, je nezbytné do něj naimportovat data. Data lze nahrát hned z několika zdrojů, např. z databáze, datových kanálů, webu nebo textových dokumentů. Pokud je potřeba data nejprve setřídít a upravit, je vhodné před samotným nahráním použít nástroj Power Query. [12]

### 4.1 Programovací jazyk DAX

DAX je programovací jazyk používaný v rámci rozhraní Power BI a Power Pivot pro Excel. DAX zahrnuje funkce, operátory a hodnoty k provádění pokročilejších dotazů a výpočtů s daty v příslušných tabulkách a sloupcích datového modelu. Tento jazyk byl vyvinut speciálně pro práci s tabulkovými datovými modely, na jejímž principu pracuje mimo jiné i Power Pivot. Tvorba vzorců v rámci jazyka DAX je velmi podobná tvorbě vzorců v rámci samotného Excelu, některé příkazy jsou dokonce stejné pro oba případy. V čem se však liší je, že DAX vztahuje vzorec na celý sloupec, zatímco vzorce vytvořené v Excelu jsou vztaheny pouze na jednotlivé buňky. Pokud tedy uživatel v rámci DAX použije vzorec k úpravě jednoho řádku v daném sloupci, vzorec se aplikuje na všechny řádky příslušného sloupce. [12], [14]

Při vytváření vzorců je potřeba brát zřetel na datový typ sloupců, aby nedocházelo k zbytečným chybám. DAX disponuje rozsáhlou sadou operátorů pro práci s různými datovými typy a řešit tak celou řadu úloh. Do této sady lze zahrnout např. operátory pro práci s aritmetickými, porovnávacími, nebo logickými úlohami. [12], [14]

### 4.2 Datový model

Před tím, než jsou prováděny operace pomocí nástroje Power Pivot je vhodné mít alespoň základní znalost o datových modelech. Datový model se zpravidla skládá z tabulek, sloupců, datových typů a relací mezi tabulkami. Tabulky v rámci datového modelu jsou zpravidla konstruovány tak, aby uchovávaly data pro podnikové entity. Například údaje o zaměstnancích jsou obsaženy v tabulce *Zaměstnanci* a údaje o objednávkách jsou obsaženy v tabulce *Objednávky*. Tabulky se pak skládají z jednotlivých sloupců, které definují atributy dané entity. Například v jednotlivých sloupcích tabulky *Zaměstnanci*, budou informace o jménu, příjmení,



datu narození zaměstnanců atd. Každý z těchto sloupců je definován datovým typem, který závisí na tom, jaké informace jsou v něm obsaženy, např. jméno zaměstnance je datový typ *text*, datum narození je *datum* a rok narození *celé číslo*. Každý řádek v tabulce by měl být jedinečný, aby nedocházelo k duplicitám a chybám. [12], [14]

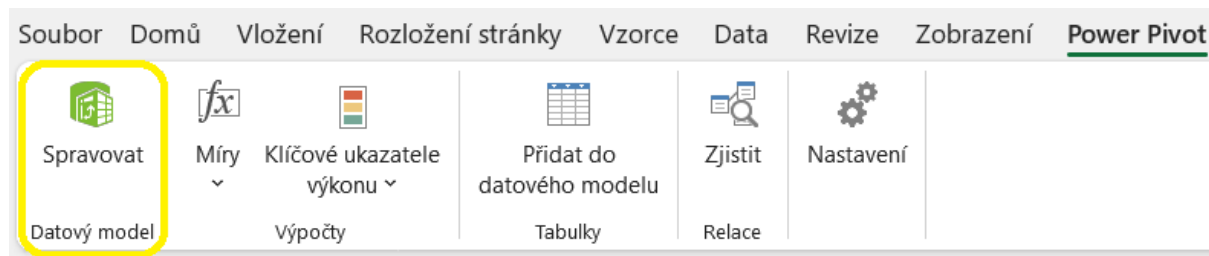
Pro to, abychom získaly „dobrý“ datový model je třeba jej normalizovat. To později pomůže urychlit práci Power Pivota a zvýší tak efektivitu práce. Normalizaci lze chápat jako proces organizování a třídění dat. Jedním z typů je třeba uchovávat ve sloupci pouze jednu informaci, jako např. rozdělit sloupeček se jménem zákazníka na dva sloupce, jméno a příjmení. Vyhnout se míchání dat, které spolu nesouvisí, např. údaje data o zákazníkovi s daty o zásobách ve skladu. Pokud datový model obsahuje nenormalizovaná data, je možné upravit za pomoci Power Query. [12], [14]

Poté co je datový model normalizován, je třeba určit, jak jsou jednotlivé tabulky mezi sebou provázány. Kupříkladu chceme-li získat data o prodeji v rámci jednotlivých věkových skupin, je potřeba propojit tabulky *zákazníci*, *prodeje* a *produkty*. Jednotlivé tabulky jsou propojovány na základě tzv. *klíčů*. Každý řádek v tabulce je definován jedním nebo vícero sloupci, které identifikují řádek. Tak je definován *primární klíč*. V některých případech lze primární klíč určit jednoduše, např. číslo zákazníka, jindy je třeba data podrobit analýze k nalezení toho správného primárního klíče. Pokud chceme vytvořit relaci mezi jednotlivými tabulkami, je třeba aby primární klíč z jedné tabulky byl obsažen v druhé tabulce ve formě tzv. *cizího klíče*. Poté co jsou identifikovány klíče v jednotlivých tabulkách je možné začít vytvářet relace pomocí nástroje Power Pivot. [12], [14]

To, jak nainportovat data do datového modelu, je popsáno v kapitole 3.2.

### 4.3 Power Pivot pro Excel

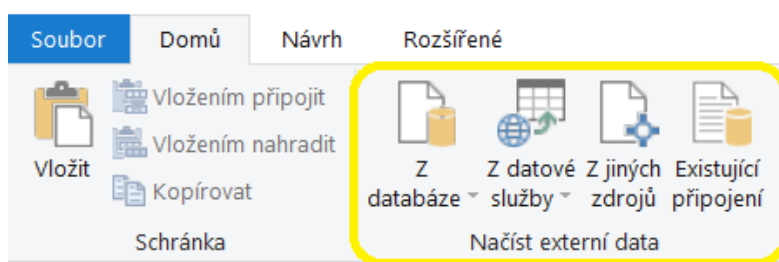
Jak již bylo řečeno, Power Pivot je nástroj vyvinutý společností Microsoft a je dostupný i v rámci aplikace Excel. Stejně jako Power Query i Power Pivot se poprvé objevil ve verzi Excel 2010. Pro použití v této verzi je však nutné jej stáhnout z webu společnosti Microsoft a nainstalovat. Pro použití ve verzi Excel 2013 a novější, je již součástí Excelu, je zapotřebí jej aktivovat prostřednictvím nastavení. Po zaktivnění se v náhledu aplikace objeví karta s názvem *Power Pivot*. Samotné prostředí nástroje Power Pivot spustíme po kliknutí na ikonu *Spravovat*, která se nachází v sekci *Datový model* na kartě *Power Pivot*, viz. *Obrázek 7*. Po kliknutí na ikonu dojde k načtení nového okna (Power Pivot pro Excel), které je připojené k stávajícímu Excelovskému sešitu. Mezi těmito okny lze jednoduše přepínat. V rámci okna Power Pivot probíhají úpravy s daty, výpočty nových sloupců, vytváření relací mezi tabulkami, nastavení hierarchií atd., zatímco jejich vizualizace prostřednictvím kontingenčních tabulek a grafů, probíhá v prostředí samotného Excelu. [12], [15]



Obrázek 7: Umístění Power Pivota v rámci Excelu

Data lze do prostředí Power Pivot nahrát opět několika způsoby, první z nich byl již zmíněn v kapitole 3.2, tedy nahrání dat do datového modelu rovnou po úpravě v prostředí Power Query. Pokud data po práci v Power Query nebyla nahrána rovnou do datového modelu, lze je do něj nahrát zpětně. Na kartě Data po kliknutí na políčko Dotazy a připojení se zobrazí lišta se seznamem všech již nahraných tabulek a databází. Kliknutím pravým tlačítkem myši na zvolená data se objeví nabídka, kliknutím na *Načíst do...* se zobrazí okno, viz *Obrázek 6*. Zde lze zaškrtnout možnost *Přidat tahle data do datového modelu*. [12], [15]

Další z možností je, data nahrát z externích zdrojů rovnou do okna Power Pivotu, a to zvolením jedné z možností nahrání uvedených na domovské kartě, viz *Obrázek 8*. Tato možnost připadá v úvahu pouze pokud jsou zdrojová data již upravená a připravená pro import. V odlišném případě je zapotřebí data nejprve upravit v Power Query, aby bylo možné s nimi dále účinně pracovat. Propojení mezi Power Pivotem a zdrojem dat setrvává pouze po dobu přebírání dat, jakmile jsou data převedena do datového modelu spojení je ukončeno. [12], [15]



**Obrázek 8: Import dat do Power Pivotu**

Prostředí Power Pivotu lze vidět na obrázku (viz přílohy *Obrázek 53*), viz. příloha. Horní lišta obsahuje čtyři karty (*Soubor*, *Domů*, *Návrh* a *Rozšířené*), na domovské kartě mimo jiné lze přepínat mezi *Zobrazením dat* a *Zobrazením diagramu*. V *Zobrazení dat* jsou data zobrazena v jednotlivých tabulkách, v tomto prostředí se data mohou dále upravovat. Nahraná data jsou zobrazena ve sloupcích se zeleným záhlavím, zatímco sloupce vytvořené v Power Pivotu mají záhlaví černé, pro lepší přehlednost. V *Zobrazení diagramu* lze vidět jednotlivé tabulky a relace mezi nimi. V tomto prostředí dochází právě k tvorbě relací a nastavení hierarchií. Ukázka jednotlivých nástrojů a práce s nimi bude podrobněji vysvětlena v praktické části práce. [12], [15]

## 5 Představení technických příkladů

Poté co v teoretické části bylo popsáno prostředí a jednotlivé nástroje, které jsou v této práci použity, nastal čas na představení samotného příkladu. Cílem práce je vytvořit sadu příkladů, respektive jeden komplexní příklad rozdělený na jednotlivé dílčí pod příklady. Je vhodné provázat logickou souvislost jednotlivých dílčích příkladů, na kterých budou demonstrovány jednotlivé funkcionality použitého prostředí. Tento příklad bude vycházet z fiktivní databáze podniku zabývajícího se výrobou a prodejem jízdních kol a komponentů. Výstupem by měla být databáze obsahující data o jednotlivých prodávaných položkách, postupu jejich výroby (pokud jsou vyráběny), zákaznicích a jejich objednávkách. Postup práce je pak chronologicky rozdělen do třech dílčích příkladů. První z nich popisuje práci v prostředí Power Query, druhý v prostředí Power Pivotu a poslední tvorbu kontingenčních tabulek a grafů. Napříč jednotlivými příklady budou provedeny následující operace:

- Upravení a třídění vstupních dat
- Sjednocení dat do datového modelu
- Vytvoření kusovníkové hierarchie a výpočet cen jednotlivých položek
- Určení cen a zisku z jednotlivých objednávek
- Výpočet měř a sloupců k zobrazení klíčových ukazatelů a trendů
- Vytvoření kontingenčních tabulek a grafů s průřezy

## 6 Příklad 1: Úprava dat v Power Query

Jak již bylo nastíněno, prvním krokem bude úprava dat v Power Query. Tato kapitola podrobně popisuje provedené úpravy na základě dotazů. Každá podkapitola se zabývá úpravou jednoho konkrétního dotazu. Během této kapitoly by se měl uživatel seznámit se způsoby nahrání dat, nastavením datového typu, nahrazením hodnot, rozdělením/sloučením sloupců a dalšími užitečnými operacemi.

### 6.1 Dotaz Položky

Prvním krokem bude provedení dotazu týkající se importu položek kusovníku, tedy všechny komponenty, ze kterých se skládá jízdní kolo.

#### 6.1.1 Načtení dat ze souboru Excel do prostředí Power Query

V novém Excelovském sešitu, pojmenovaném *Příklad Kola*, přejděte na kartu *Data*, skupina *Načíst a transformovat* a po kliknutí na šipku u tlačítka *Načíst data* se rozbalí seznam s různými způsoby importu dat. První tabulka se nachází v souboru typu *\*.xlsx*, tedy v Excelovském sešitu. Pro import zvolte variantu získat data *Ze souboru* a poté klikněte na *Ze sešitu aplikace Excel*. Dojde k zobrazení *Průzkumníka souborů*, pomocí něhož vyhledejte umístění souboru *Položky vstupní data.xlsx* a klikněte na tlačítko *Importovat*. Následně dojde k zobrazení okna *Navigátoru*, ve kterém si uživatel může projít jednotlivá data obsažená ve zdrojovém souboru. Na levé straně navigátoru se nachází lišta se seznamem dotazů. Po kliknutí na libovolný dotaz

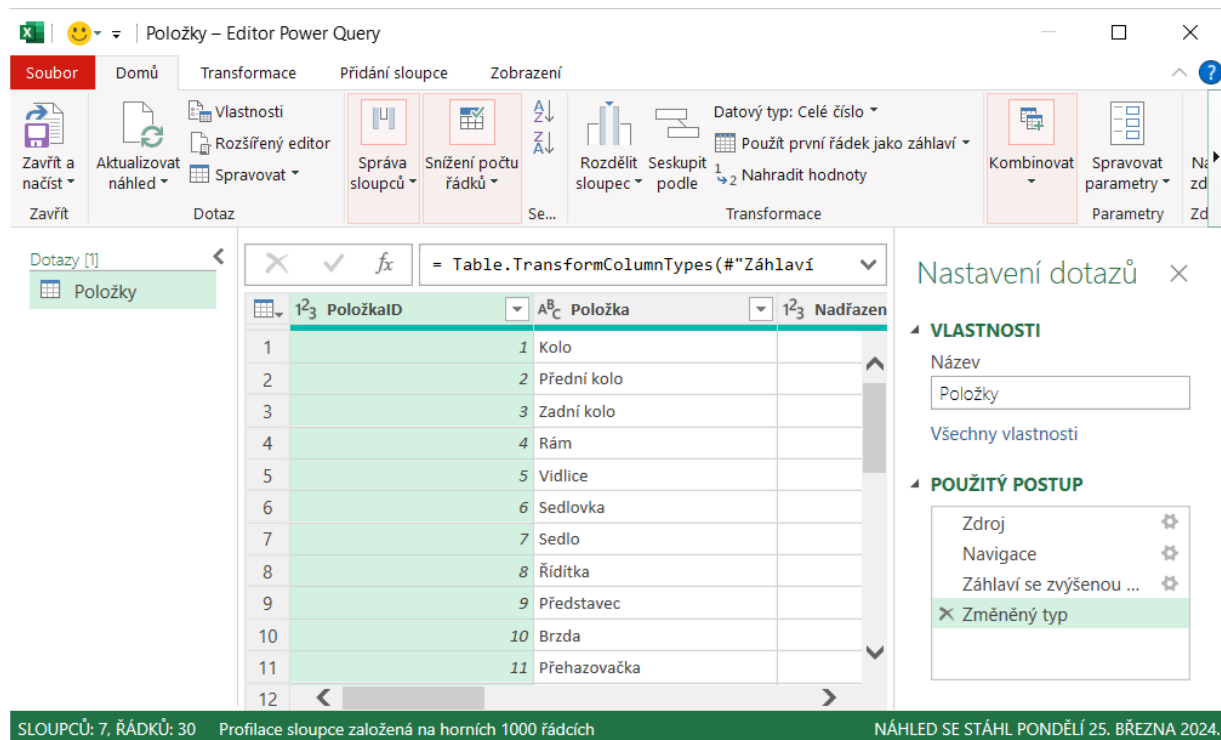
The screenshot shows the Power Query Navigator window. On the left, there is a search bar and a list of data sources. The selected source is 'Položky' from the file 'Položky vstupní data.xlsx'. On the right, a table displays the data for the selected query.

PoložkaID	Položka	NadřazenáPoložka	NázevPoložky
1	Kolo		VELO - SuperSpeed 29 Pro
2	Přední kolo	1	VELO - 320 Wheel Al pro 29 F
3	Zadní kolo	1	VELO - 320 Wheel Al pro 29 F
4	Rám	1	VELO - HT Frame 29
5	Vidlice	1	DT Swiss - F 232 "29"
6	Sedlovka	1	VELO - Seatpost Al 31,6mm
7	Sedlo	1	Selle Italia - Comfort 720
8	Řídítka	1	VELO - Handlebar Al 760mm
9	Představec	1	VELO - HandlebarStem Al 80r
10	Brzda	1	SHIMANO - Brzdová páka SLX
11	Přehazovačka	1	SRAM - GX 11s-black
12	Řazení	1	SRAM - GX 11s-right
13	Kazeta	1	SRAM - XG-1175, 10-42z
14	Převodník	1	SRAM - X-sync 11s, 32z
15	Řetěz	1	KMC - 1090-11s
16	Kliky	1	TRUVATIV - Dub Bboost 175r
17	Přední kotouč	2	SHIMANO - BR5600 180mm
18	Zadní kotouč	3	SHIMANO - BR5600 160mm
19	Přední třmen	2	SHIMANO-BR5600
20	Zadní třmen	3	SHIMANO-BR5600
21	Přední ráfek	2	VELO - 320 Rim Al pro 29 F
22	Zadní ráfek	3	VELO - 320 Rim Al pro 29 R
23	Zadní náboj	3	DT Swiss - Hub 360r

At the bottom of the window, there are buttons for 'Načíst', 'Transformovat data', and 'Zrušit'.

Obrázek 9: Okno Navigátoru – dotaz Položky

ze seznamu se v pravé části navigátoru zobrazí náhled tabulky. Pro výběr více tabulek je nutno v levé liště zaškrtnout pole *Vybrat více položek*, tím se zobrazí zaškrtnovací pole u všech obsažených dotazů a jeho zaškrtnutím lze vybírat vícero položek ze seznamu. V případě souboru *Položky vstupní data* se v levé liště zobrazí pouze jediný dotaz s názvem *Položky*. Pro načtení dat do prostředí Power Query jej označte a stiskněte tlačítko *Transformovat data*, viz *Obrázek 9*. Nyní došlo k načtení dat do prostředí Power Query, které se otevřelo v novém okně.



**Obrázek 10: Prostředí Power Query – dotaz Položky**

Prostředí Power Query (viz *Obrázek 10*) je na první pohled podobné prostředí Excelu. Ve vrchní části se nachází seznam karet, pod nímž se rozkládají skupiny s jednotlivými nástroji. Pomocí těchto nástrojů dochází k oné úpravě dat do finální podoby a mnoha jiného. Pod panelem nástrojů se nacházejí data uspořádaná do tabulky. Na levé straně v záložce *Dotazy* pak lze přepínat mezi jednotlivými tabulkami, podobně jako tomu je u přepínání mezi listy v aplikaci Excel. Na opačné straně se nachází lišta se záznamem použitých a provedených úprav. Jelikož nástroj Power Query nedisponuje tlačítkem *Zpět*, je třeba veškeré opravy učinit právě v této liště. Poslední z důležitých částí prostředí je lišta ve spod samotného okna, jedná se o tzv. *Stavový řádek*, který obsahuje např. informace ohledně počtu řádků, sloupců a čase zobrazení.

### 6.1.2 Úprava dotazu Položky v prostředí Power Query

Poté, co jsou data úspěšně načtena, je možné s nimi začít pracovat. V následující kapitole bude představeno několik užitečných operací, pomocí nich lze data upravit a zpřehlednit.

Při pohledu zpět do dotazu *Položky* (viz *Obrázek 10*) je patrné, že v liště použitý postup se již nacházejí proběhlé změny. Tyto změny provedl nástroj Power Query automaticky. Konkrétně v případě tohoto dotazu došlo k převodu prvního řádku na záhlaví a změně datového typu určitých sloupců. Může však nastat situace, že Power Query neprovede automaticky tyto operace a je potřeba je provést ručně (např. jsou-li data importována z textového dokumentu).

Nastavení záhlaví a datového typu lze provést za pomoci nástrojů ve skupině *Transformace* na domovské kartě.

Ačkoliv došlo k automatickému určení datového typu, lze si povšimnout, že u některých sloupců stále není datový typ jednoznačně definován, nebo dokonce neodpovídá požadavkům. Je proto doporučeno tyto změny kontrolovat. Při pohledu do sloupců *NadřazenáPoložka*, *NákupníCena* a *Cena materiáluna*, se zde vyskytuje kromě číselných hodnot i text. Jelikož ale dané sloupce mají být datového typu *Celé číslo*, respektive *Měna*, je zapotřebí tento text převést na číselnou hodnotu. O tom ale až později.

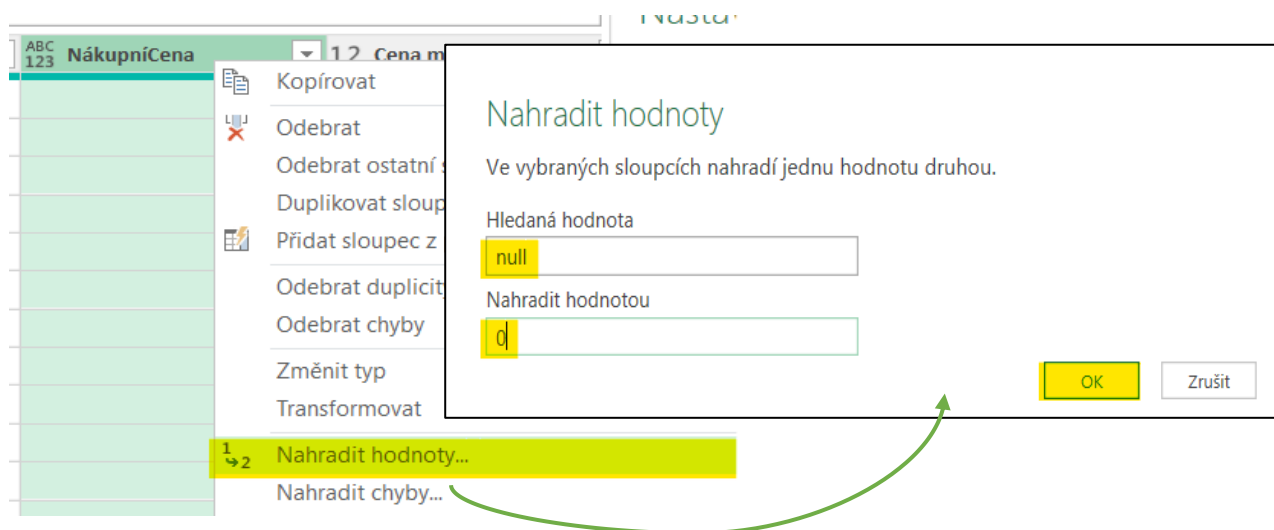
1 <sup>23</sup> NadřazenáPoložka	A <sup>B</sup> C NázevPoložky	1.2 NákupníCena	1 <sup>23</sup> Cena materiáluna
	SuperSpeed 29 Pro	null	null
	20 Wheel Al pro 29 F	null	null
	20 Wheel Al pro 29 R	null	null
	HT Frame 29	null	1200
	5 - F 232 "29"	5200	null
	Seatpost Al 31,6mm	null	280

Obrázek 11: Informace o výskytu prázdných řádků

Při najetí kurzoru myši na zelenou lištu pod záhlavím sloupců (viz Obrázek 11) se zobrazí informační panel, který říká kolik řádků je platných, kolik je chybných a kolik prázdných. Ve zmíněných třech sloupcích se vyskytují prázdné řádky znázorněné textem *null*. Aby bylo možné v budoucnu se sloupci počítat, je zapotřebí aby obsahovaly pouze hodnoty jednoho datového typu. Toho se docílí změnou hodnot ve sloupci v prostředí Power Query (nikoliv v originálním datovém zdroji).

### 6.1.2.1 Nahrazení hodnot

Pomocí pravého tlačítka myši klepněte na sloupec, ve kterém chcete zaměnit hodnoty, a poté zvolte z nabídky pole *Nahradit hodnoty* (viz Obrázek 12). Zobrazí se nová karta, kde dojde k definování zaměněných hodnot. Do pole *Hledaná hodnota* vepište „*null*“ a do pole *Nahradit hodnotou* vepište „*0*“, následně potvrďte operaci tlačítkem *OK*. Pomocí zmíněného návodu



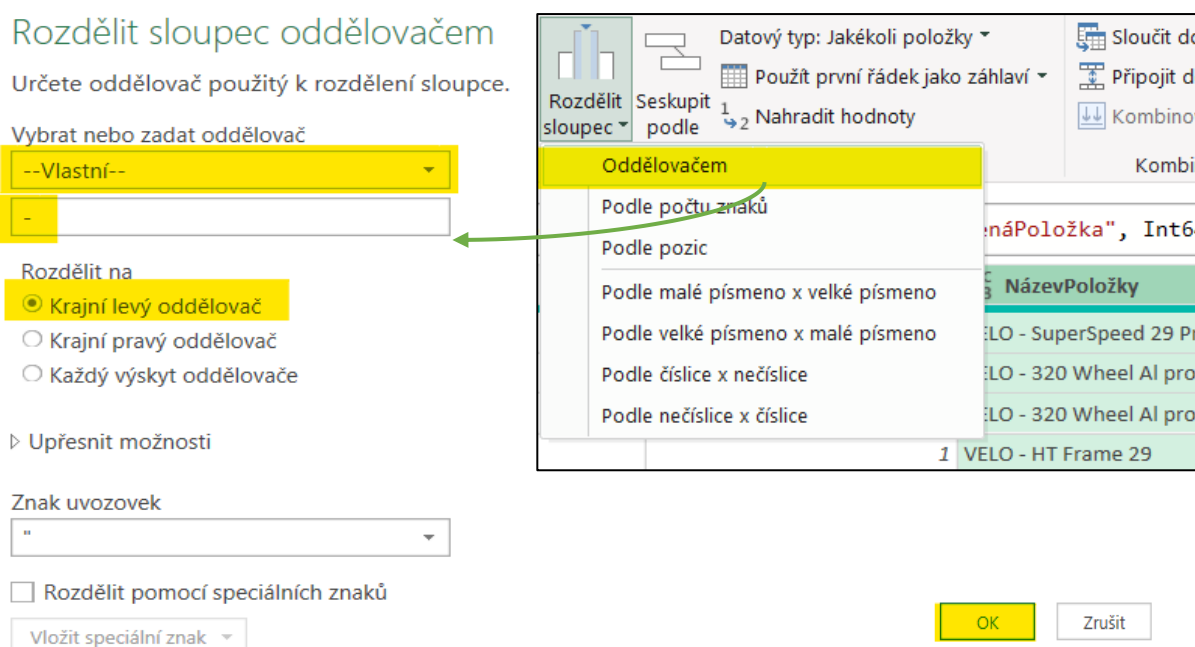
Obrázek 12: Nahrazení hodnot – dotaz Položky

nahradíte hodnoty ve sloupcích *NadřazenáPoložka*, *NákupníCena* a *Cena materiáluna*.

**TIP:** pokud chcete zaměnit stejné hodnoty ve vícero sloupcích, přidržetím klávesy **CTRL** klikněte levým tlačítkem myši na příslušné sloupce, pro jejich označení, a poté klepněte pravým tlačítkem do libovolného ze zvolených sloupců. Dále pokračujte podle výše uvedeného postupu. Změna se tak aplikuje na všechny zvolené sloupce. Pokud však chcete zaměnit v každém sloupci jiné hodnoty, je zapotřebí postupovat u každého sloupce jednotlivě.

### 6.1.2.2 Rozdělení sloupce pomocí oddělovače

Další z mnoha užitečných funkcí Power Query je rozdělování sloupců. Pokud je nutné rozdělit sloupec na vícero jednotlivých sloupců, např. sloupec *Jméno* na *Jméno* a *Příjmení*, lze použít nástroj *Rozdělit sloupec*. Ten lze najít ve skupině *Transformace* na kartě *Domů*. V případě tabulky *Položky*, je požadováno rozdělení sloupce *NázevPoložky* na dva nové. Tento sloupec totiž obsahuje informaci o názvu výrobce (vlevo) a označení produktu (vpravo za pomlčkou). Pokud sloupec obsahuje vícero informací, je vhodné jej rozdělit, aby v každém sloupci byla obsažena pouze jedna informace. Předejde se tím případným komplikacím při tvorbě počítaných sloupců v Power Pivotu či finálních vizuálů.



Obrázek 13: Rozdělení sloupce oddělovačem

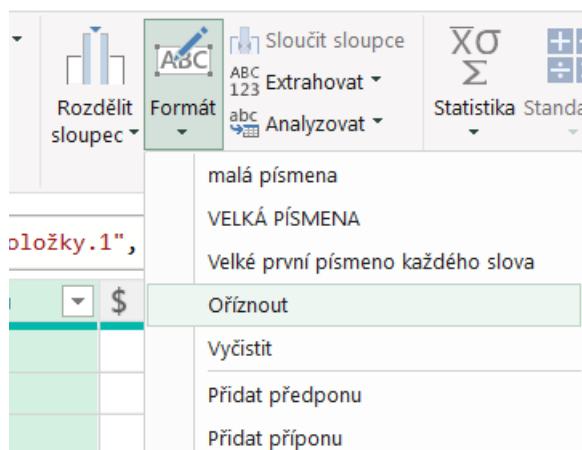
K rozdělení sloupce použijte možnost *Rozdělit sloupec oddělovačem*, a to klepnutím levého tlačítka myši na šipku u nástroje *Rozdělit sloupec* a zvolením první možnosti ze seznamu. Po zvolení způsobu rozdělení se zobrazí okno s upřesňujícími informacemi, viz *Obrázek 13*. V sekci *Vybrat nebo zadat oddělovač* zvolte možnost *Vlastní* a následně definujte oddělovač. V tomto případě je oddělovačem (separátorem) pomlčka. Poté zaškrtněte pole *Krajní levý oddělovač* a potvrďte tlačítkem *OK*. Je potřeba dbát na to, že zvolený oddělovač se musí vyskytovat ve všech řádcích sloupce, jinak by mohlo dojít k chybě. Zvolením krajního levého oddělovače dojde k oddělení sloupců v místě prvního výskytu definovaného oddělovače z levé strany. Pokud by se v každém řádku vyskytoval pouze jediný oddělovač, bylo by možné zvolit krajní pravý oddělovač nebo každý výskyt oddělovače. Jelikož se ale v některých řádcích sloupce *NázevPoložky* vyskytuje vícero pomlček, je jediným správným řešením použití krajního levého oddělovače. Pro každý případ je potřeba pečlivě promyslet jaké nastavení oddělovače je potřeba použít.

Nástroj *Rozdělit sloupec* lze také nalézt v nabídce, která se zobrazí při stisknutí pravého tlačítka myši na příslušný sloupec.

### 6.1.2.3 Odstranění mezery před textem

Po rozdělení sloupce *NázevPoložky* vznikly dva nové sloupce (*NázevPoložky.1* a *NázevPoložky.2*). Na první pohled vypadají oba sloupce v pořádku, avšak lze si povšimnout že ve druhém sloupci jsou některé hodnoty lehce posunuté oproti zbylým. To je způsobeno tím, že některé řádky původního sloupce *NázevPoložky* měly za oddělovačem mezeru a některé ne. Tím pádem se část hodnot vložila s mezerou před textem. Jedná se spíše o kosmetickou vadu, ale v některých případech by i tato drobnost mohla způsobit problémy při dalších úpravách.

Pro odstranění mezery před textem lze využít další z nástrojů implementovaných v prostředí Power Query. Jedná se o funkci oříznutí. Ta projde řádky vybraného sloupce a pokud se v některém z nich vyskytne mezeru před textem, odstraní ji. Nástroj se nachází na kartě *Transformace* ve skupině *Sloupec Text*. Pro oříznutí sloupce *NázevPoložky.2* nejprve označte sloupec a poté klikněte na šipku pod polem *Formát*, tím se rozbalí nabídka, v níž zvolte nástroj *Oříznout*. Postup je vykreslen na obrázku níže, viz *Obrázek 14*.



Obrázek 14: Nástroj oříznutí

### 6.1.2.4 Změna názvu sloupce

Pokud je uživatel nespokojen s názvem některého ze sloupců, lze jej jednoduše změnit. Název se dá změnit dvojitým poklepnutím do jeho záhlaví anebo kliknutím pravého tlačítka myši na sloupec a zvolit možnost *Přejmenovat*.

Pro tabulku *Položky* změňte názvy těchto sloupců:

- *NázevPoložky.1* → *Výrobce*
- *NázevPoložky.2* → *OznačeníProduktu*
- *NákupníCena* → *NákupníCena [1 ks]*
- *Cena materiáluna* → *CenaMateriálu [1 ks]*

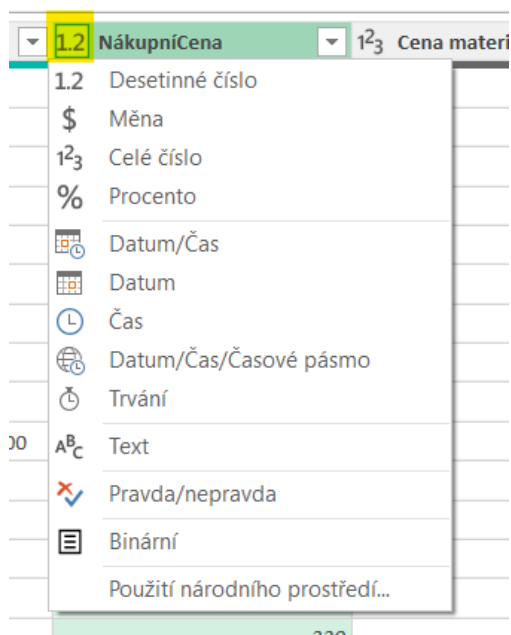
**TIP:** Pro vložení hranatých závorek použijte klávesovou zkratku **pravý ALT + f** nebo **g** (při rozložení české klávesnice).

### 6.1.2.5 Změna datového typu sloupce

Nyní, když jsou všechny hodnoty ve sloupcích řádně upraveny, je možné nastavit potřebné datové typy sloupců. Toho lze docílit jednoduše poklepnutím levého tlačítka myši na symbol



vlevo od názvu sloupce. Ten symbolizují aktuální datový typ. Po klepnutí na zmíněný symbol se zobrazí seznam s možnostmi nastavení datového typu viz *Obrázek 15*. Kliknutím na zvolený



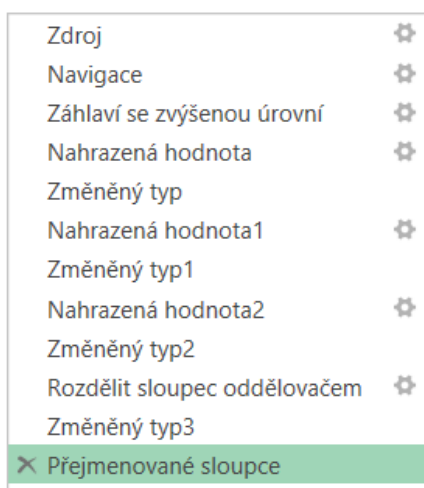
**Obrázek 15: Nastavení datového typu sloupce – dotaz Položky**

datový typ se změní všechny hodnoty daného sloupce. Změnu datového typu lze taktéž provést pomocí tlačítka *Datový typ* ve skupině *Transformace* na kartě *Domů*. Pro tabulku *Položky* postupně projděte všechny sloupce a změňte jejich typ, pokud již není správně nastaven, na typ odpovídající obrázku níže (viz *Obrázek 16*).



**Obrázek 16: Přejmenované sloupce – dotaz Položky**

#### POUŽITÝ POSTUP



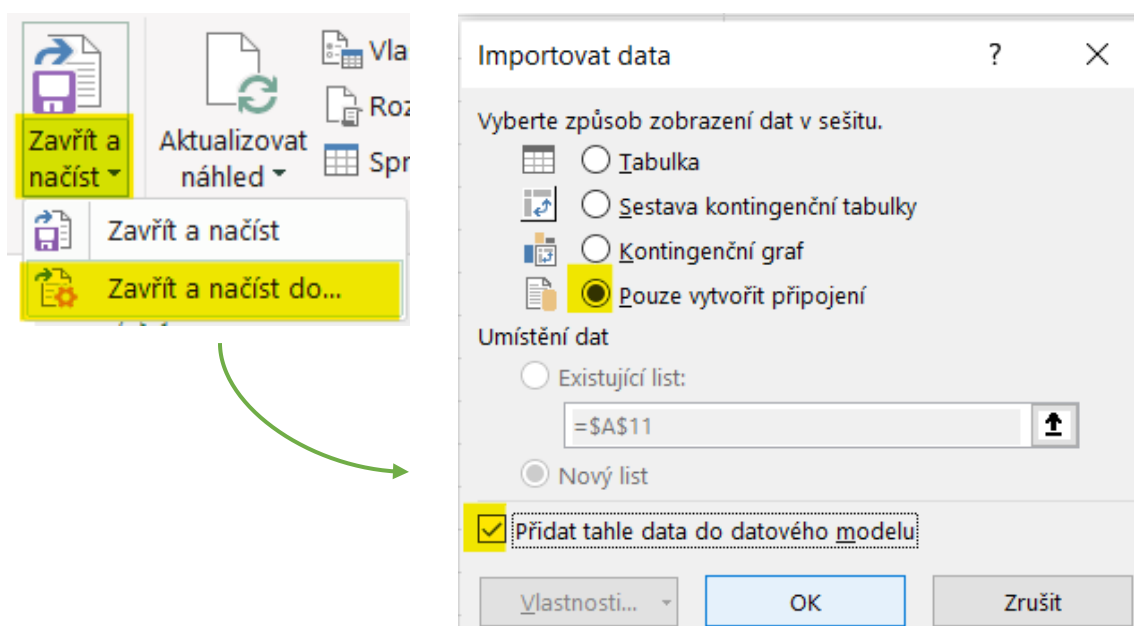
**Obrázek 17: Použitý postup – dotaz Položky**

Nyní jsou data v tabulce *Položky* „vyčištěná“ a upravená na požadovanou podobu. Postup úprav pro tuto tabulku by pak měl vypadat následovně – viz *Obrázek 17*.

Výsledný postup se může lišit pořadím u některých provedených úprav, položky seznamu by však měly být stejné.

### 6.1.2.6 Import dat do sešitu

Takto upravená data jsou připravena pro import do Excelu. Přejděte na kartu *Domů* a v levé části ve skupině *Zavřít*, klepněte na šipku u pole *Zavřít a načíst* a zvolte možnost *Zavřít a načíst do....* Tím dojde k zavření okna prostředí Power Query a zobrazí se okno *Importovat data*, viz *Obrázek 18*. Zvolte možnost *Pouze vytvořit připojení* a zaškrtněte že chcete *Přidat tahle data do datového modelu*, potvrďte tlačítkem *OK*. Tímto krokem se vytvořilo připojení mezi Excelovským sešitem a zdrojem dat. Data nejsou přímo nahrána v sešitu, ale pouze se odkazují na zdrojový soubor. Tato varianta je efektivní zejména při práci s větším objemem dat, kdy pak nedochází k výraznému zvětšení velikosti Excelovského souboru.



Obrázek 18: Import dat do Excelu

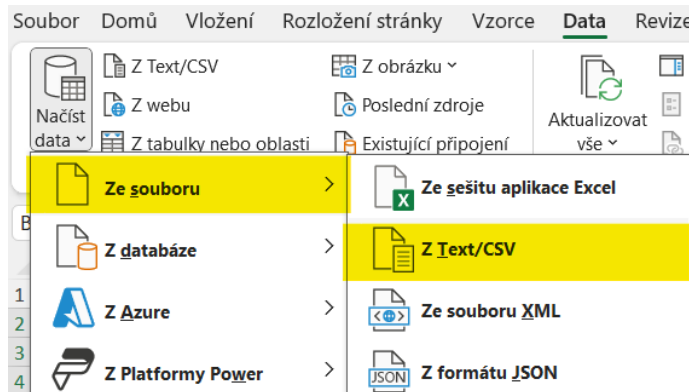
Zaškrtnutím pole *Přidat tahle data do datového modelu*, se data automaticky nahrála do prostředí Power Pivot. Pokud se s daty bude dále zacházet v prostředí Power Pivotu je vhodné je takto rovnou naimportovat.

## 6.2 Dotaz Pracoviště

V následujícím kroku dojde k nahrání a úpravě dotazu *Pracoviště*. Tento dotaz obsahuje informace o hodinových sazbách jednotlivých pracovišť, které slouží k výpočtu nákladů vyráběných položek.

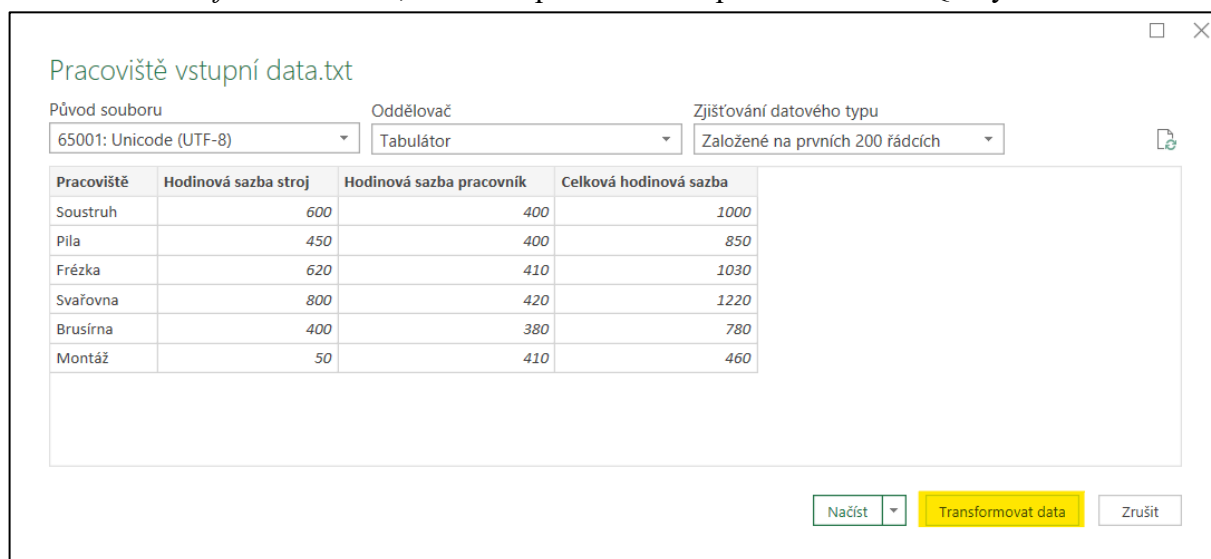
## Načtení dat z textového souboru

Postup načtení dat z textového dokumentu je obdobný, jako tomu bylo v případě souboru Excel. Pro import dat přejděte do prostředí aplikace Excel a na kartě *Data* ve skupině *Načíst a transformovat data* klepněte na pole *Načíst data*, *Ze souboru* a zvolte možnost *Z Text/CSV*, viz. *Obrázek 19*.



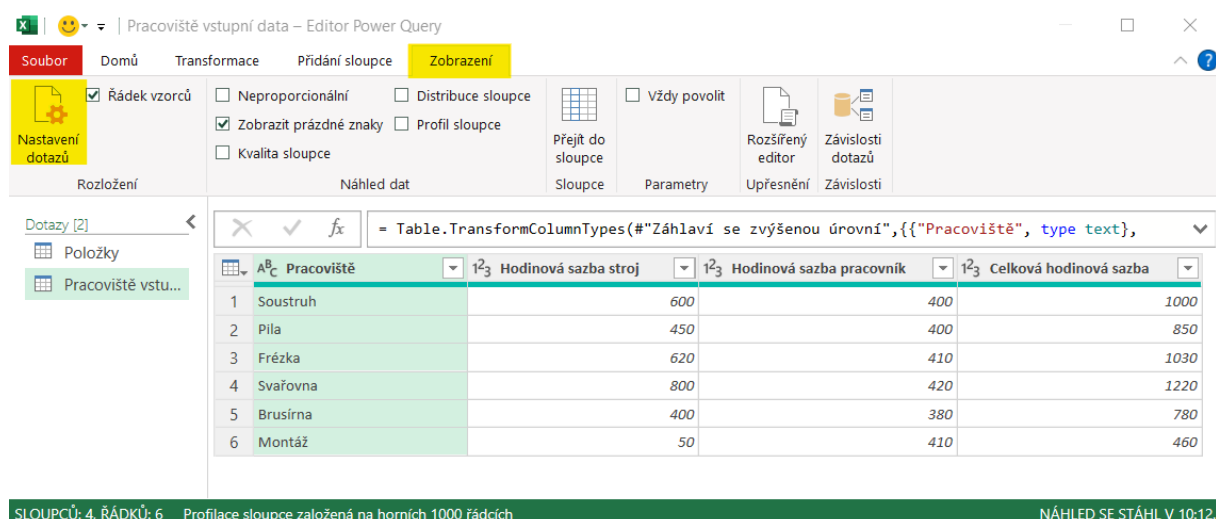
Obrázek 19: Import dat ze souboru Text/CSV

Najděte stažený soubor, *Pracoviště vstupní data.txt*, na vašem disku a naimportujte jej. Následně dojde k zobrazení okna navigátoru (viz *Obrázek 20*), které je podobné dialogovému oknu z předchozí kapitoly. Tentokrát není na výběr z vícero tabulek k importu, tudíž se naimportují veškerá data z textového dokumentu. Po zobrazení okna navigátoru pouze klepněte na tlačítko *Transformovat data*, čímž se opět ocitnete v prostředí Power Query.



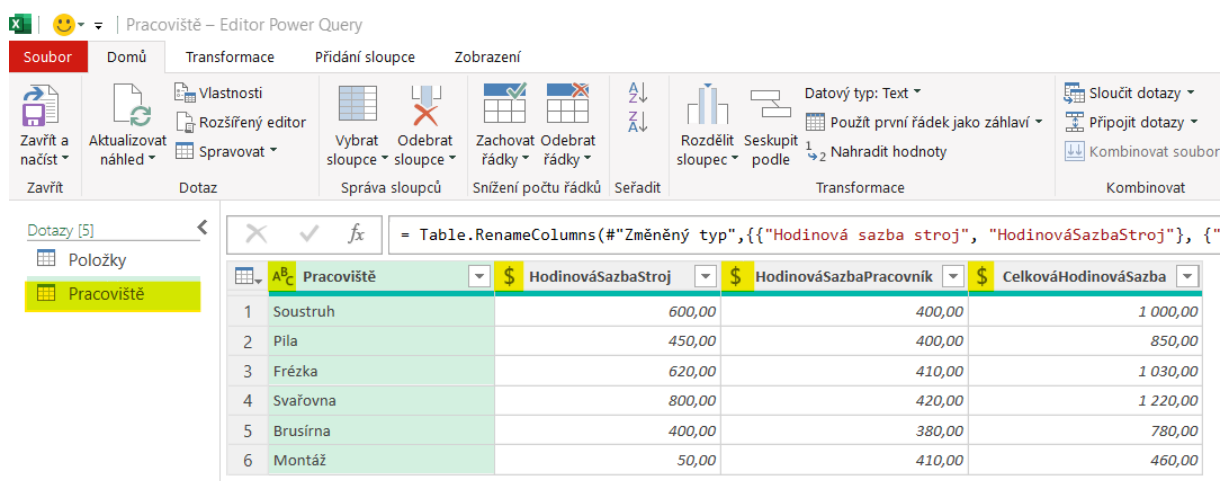
Obrázek 20: Okno navigátoru – dotaz Pracoviště

V náhledu prostředí Power Query, viz *Obrázek 21*, si lze všimnout, že na pravé straně chybí lišta *Nastavení dotazů*. Pokud nastane situace, že se vám po otevření Power Pivotu nezobrazí, nebo si jí nechtěně zavřete, lze jí zase jednoduše zobrazit (na kartě *Zobrazení* poklepejte na pole *Nastavení dotazů* ve skupině *Rozložení*). Dalším rozdílem oproti minulému importu je ten, že se na kartě *Dotazy*, na levé straně okna, vyskytují oba naimportované dotazy. Kdykoli tak otevřete Power Query, je možné jednoduše přepínat mezi všemi dotazy co byly do sešitu naimportovány a znovu je upravovat.



Obrázek 21: Prostředí Power Query – dotaz Pracoviště

Data po nahrání jsou téměř v požadované podobě, tudíž stačí provést pouze drobné úpravy a posléze data načíst. Nejprve změňte název dotazu na *Pracoviště*. Přejmenování dotazu lze provést dvojklikem nebo přes pravé tlačítko a výběrem možnosti *přejmenovat*, stejně jak tomu bylo u záhlaví sloupce. Následně přejmenujte jednotlivé sloupce a změňte datový typ podle obrázku níže (Obrázek 22). Hodnoty ve sloupcích jsou v pořádku, tudíž zbývá jen načíst data do datového modelu viz. *Kapitola 6.1.2.6*.



Obrázek 22: Dotaz pracoviště výsledná podoba

## 6.3 Dotaz Časy operací

Jako další na řadě je dotaz *ČasyOperací*. V rámci něj jsou zaznamenána data určující kolik času daný výrobek stráví na daném Pracovišti. Společně s daty obsaženými v dotazu *Pracoviště* jsou použita k výpočtu nákladů na vyráběné položky.

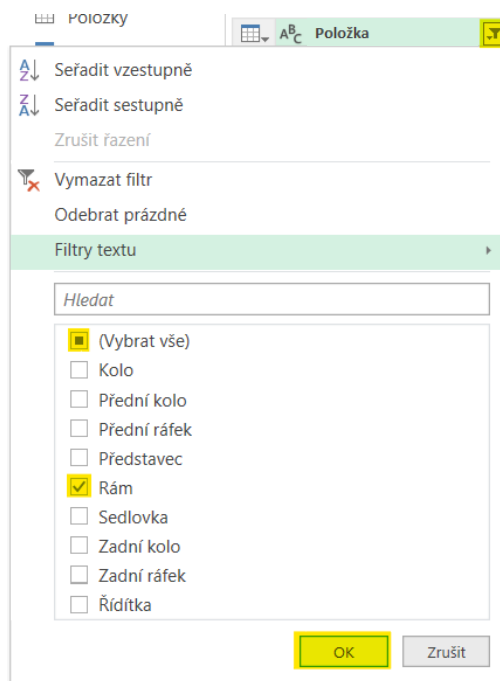
### 6.3.1 Načtení dat ze souboru CSV

Pro nahrání dat ze souboru CSV se postupuje stejně jako v případě textového dokumentu. Opět tedy zvolte pole *Z Text/CSV* ve skupině *Načíst a transformovat* na kartě *Data* a otevřete soubor *Časy operací vstupní data.csv*. V okně navigátoru zvolte, podobně jako u předchozího souboru, *Transformovat data*.

V prostředí Power Query podle předchozích návodů přejmenujte název dotazu *Časy operací vstupní data* na *ČasyOperací*. Podobným způsobem přejmenujte i jednotlivé záhlaví sloupců, tedy odstraňte mezery v názvech a změňte první písmeno každého slova na velké (např. *DobaJednéOperace*). Dále netřeba dalších úprav.

### 6.3.2 Filtr dat

Před načtením dat by však bylo vhodné si vyzkoušet další funkci Power Query, a to možnost nastavení filtru. Pro filtrování řádků v konkrétním sloupci přejděte do záhlaví sloupce a klepněte na šipku v pravé části záhlaví. Tím dojde k zobrazení lišty s možnostmi řazení a filtrování řádků, viz *Obrázek 23*. Lze například abecedně seřadit řádky, odebrat prázdné řádky anebo třeba použít některé filtry z nabídky filtrů textu. V dolní části lišty se nachází seznam všech hodnot vyskytujících se v jednotlivých řádcích sloupce. Pokud chcete zobrazit pouze některé z uvedených hodnot, lze filtrovat pomocí zaškrtnutí jednotlivých polí vedle hodnot. Vybrané hodnoty jsou pak vidět v zobrazení dotazu. V případě dotazu *ČasyOperací* vyfiltrujte všechny řádky obsahující hodnotu *Rám* ve sloupci *Položka*. Nejprve od označte pole (*Vybrat vše*), poté zaškrtněte hodnotu *Rám* a potvrďte tlačítkem *OK*. Výsledkem jsou tři řádky ukazující, kolik času zabrala jedna operace na jednotlivém pracovišti v případě výroby rámu (*Obrázek 24*).



Obrázek 23: Filtr sloupce

Nyní jsou všechny úpravy hotovy a data jsou připravena k načtení. Na kartě *Domů* klikněte na pole *Zavřít a načíst do...*, následně zvolte variantu *Pouze vytvořit připojení* a přidejte data do datového modelu.

	A <sup>B</sup> <sub>C</sub> Položka	A <sup>B</sup> <sub>C</sub> Pracoviště	1 <sup>2</sup> <sub>3</sub> DobaJednéOperace [min.]
1	Rám	Pila	25
2	Rám	Svařovna	30
3	Rám	Brusírna	15

Obrázek 24: Výsledek aplikace filtru sloupce

## 6.4 Dotaz Objednávky

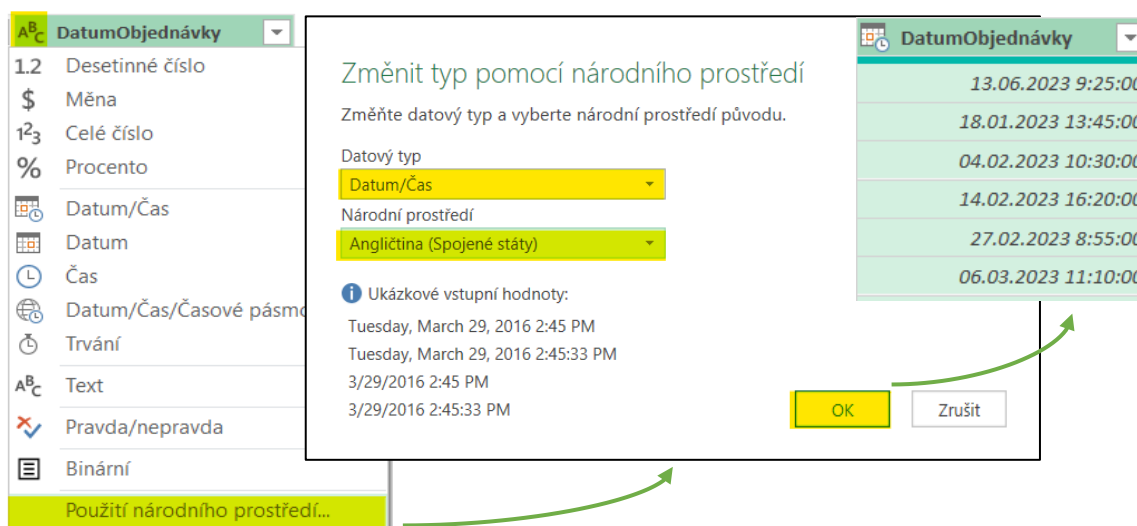
Další v pořadí je dotaz *Objednávky*. Ten obsahuje informace o veškerých objednávkách za určité období. Pro načtení dat zvolte variantu *Ze sešitu aplikace Excel* a vyberte soubor *Objednávky vstupní data.xlsx* (postup nahrání dat ze souboru Excel je popsán v Kapitole 6.1.1). V okně navigátoru zvolte dotaz *List1* a zvolte možnost *Transformovat data*. Dvojitým kliknutím na název dotazu jej změňte na *Objednávky*. Dále postupujte podle obrázku níže (*Obrázek 25*) a přejmenujte požadované sloupce dotazu.



Obrázek 25: Přejmenování sloupců – dotaz *Objednávky*

### Změna datového typu pomocí národního prostředí

Poslední, co zbývá provést, je změna datového typu sloupců. V případě dotazu *Objednávky*, jde pouze o poslední sloupec, tedy *DatumObjednávky*. Jak sám název napovídá, sloupec obsahuje informace o datu vytvoření objednávky, je tedy třeba jej převést na datový typ *Datum/čas*. Všimněte si, že pokud zvolíte z nabídky požadovaný datový typ a potvrdíte výběr, u více než poloviny řádků se objeví chyba. Ta nastala díky tomu, že data ve sloupci jsou zadána ve formátu používaném v USA, kde je prohozené pořadí dnů s pořadím měsíců. Power Query tedy zobrazil pouze datумы do 12 dne každého měsíce a zbylé vyhodnotil jako chybu. K tomu, aby bylo možné změnit datový typ, je zapotřebí použít volby národního prostředí. Po rozbalení lišty s datovými typy zvolte možnost úplně dole, *Použití národního prostředí...*. Následně dojde



Obrázek 26: Změna datového typu podle národního prostředí

k zobrazení karty, ve které se zvolí požadovaný datový typ a národní prostředí, ve kterém jsou data naimportována. Po zadání správných informací, dojde k změně zobrazení správných hodnot v českém formátu (*dd/mm/yyyy*), viz *Obrázek 26*.

Takto upravená data opět načtete jako u předchozího dotazu (pouze vytvořit připojení a načíst do datového modelu).

## 6.5 Dotaz Zákazník

Tabulka s informacemi o jednotlivých zákaznících se nachází v excelovském sešitu *Zákazníci vstupní data.xlsx*. Pro načtení dat do Power Query opět využijte možnost načtení *Ze sešitu aplikace Excel*. Zvolte dotaz *List1* a možnost *Transformace dat*. V prostředí Power Query změňte název dotazu na *Zákazníci*. Dále je zapotřebí rozdělit sloupec *Adresa* na tři dílčí sloupce, a to *Město*, *UliceČísloPopisné* a *PSČ*. Sloupec rozdělte pomocí oddělovače (postup viz kapitola 6.1.2.2) a zvolte možnost *Každý výskyt oddělovače* a jako oddělovač *Čárka*. Výsledný vzhled je zobrazen na obrázku níže (Obrázek 27).

Rozdělit sloupec oddělovačem

Určete oddělovač použitý k rozdělení sloupce.

Vybrat nebo zadat oddělovač

Čárka

Rozdělit na

Krajní levý oddělovač

Krajní pravý oddělovač

Každý výskyt oddělovače

Město	UliceČísloPopisné	PSČ
Praha 3	Smíchovské Nádraží 236	10023
Trhové Sviny	Družstevní 236	55623
Olomouc	Pod Věží 123	56232
Šumperk	Zahradní 156	21521
Černý Důl	Pod Šachtou 23	54972
Řeporyje	U starostů 456	12156
Vimperk	Pivovarská 69	38501
Valašské Meziříčí	Dálniční 1	22438
Pacov	Národního Povstání 23	42356
Rožmytál	Příčná 56	26345

Obrázek 27: Rozdělení sloupců oddělovačem – dotaz Zákazník

Nyní už zbývá jen upravit názvy zbylých sloupců a je hotovo. Názvy sloupců změňte podle následujícího obrázku (viz Obrázek 28). Po provedení veškerých úprav data opět načtete do datového modelu a vytvořte připojení s Excelovským sešitem.

Jméno	e-mail	tel.	Zákaznické číslo

Jméno	e-mail	TelefonníČíslo	ZákazníkID

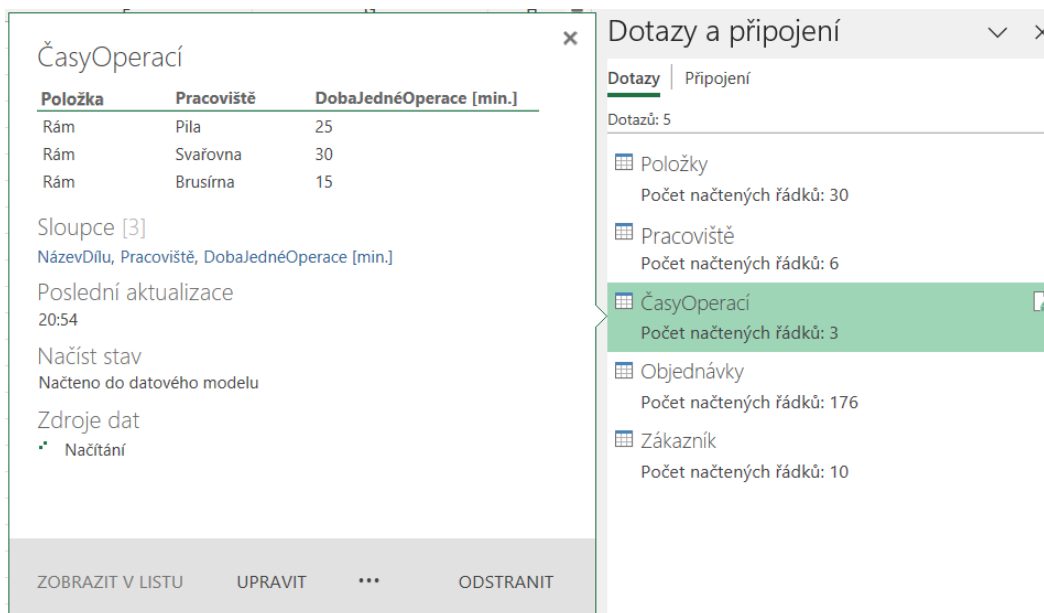
Obrázek 28: Přejmenování sloupců – dotaz Zákazník

## 6.6 Spravování Dotazů a Připojení

Nyní, když jsou potřebná data načtena, je dobré vědět, kde je nalézt a jak je dále transformovat. Ve všech případech načtení dat bylo vytvořené pouze připojení, tím pádem po zavření okna Power Query nejsou k náhledu žádná data. To je v pořádku. Na místo nahrání dat do Excelu došlo pouze k propojení se zdrojem dat. Pro zobrazení veškerých dotazů a připojení v daném souboru Excelu, přejděte na kartu *Data* a ve skupině *Dotazy a připojení* klikněte na stejnojmenné pole. Na pravé straně sešitu by se měla objevit lišta zobrazující jednotlivé dotazy a připojení. Na obrázku (viz Obrázek 29) lze vidět, že bylo připojeno pět dotazů. Při přejetí kurzoru na některý z dotazů se zobrazí jeho náhled se stručnými informacemi. V případě, že je potřeba data dodatečně upravit, dvojitým poklepáním na příslušný dotaz se otevře v prostředí Power Query.

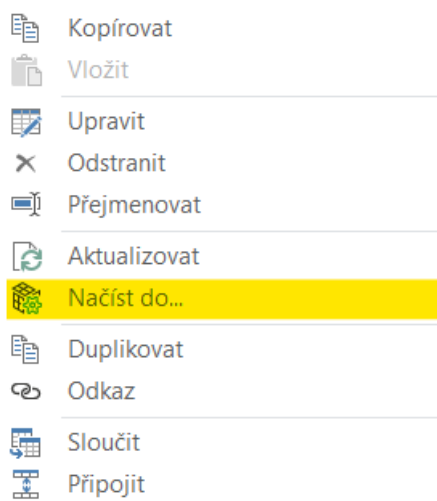
Při pohledu na dotaz *ČasyOperací*, je vidět, že jsou nahrány pouze 3 řádky. Pro demonstrování příklad však bude potřeba načíst všechny řádky. Pro úpravu tak přejděte do Prostředí Power Query a ve zmíněném dotazu zrušte filtr ve sloupci *Položka*. Toho docílíte klepnutím na postup *Filtrované řádky* v liště *Použitý postup* a následným zvolením (*vybrat vše*) ve filtru sloupce (viz kapitola 6.3.2). Poté opět klikněte na tlačítko *Zavřít a načíst* v levém horním rohu. Nyní při

zobrazení lišty *Dotazy a připojení* je vidět, že se počet načtených řádků pro dotaz *ČasyOperací* změnil na 22 řádků.



Obrázek 29: Sekce Dotazy a připojení

Lišta *Dotazy a připojení* nabízí řadu nástrojů rychlé úpravy dotazů. Při klepnutí pravým tlačítkem myši na dotaz se zobrazí seznam nástrojů pro rychlou úpravu, aniž by uživatel musel přecházet do okna Power Query. Zejména pak nástroj *Načíst do...* (viz Obrázek 30), může být užitečný. Pokud tak dojde ke špatně zvolenému způsobu nahrání dat, zvolením nástroje *Načíst do...* se zobrazí okno importu dat (Obrázek 18), ve kterém můžete redefinovat způsob zobrazení dat v sešitu.



Obrázek 30: Lišta rychlého přístupu  
Dotazy a připojení

Pro dotaz *Pracoviště* změňte způsob zobrazení dat na *Tabulka* a zrušte možnost *přidání dat do datového modelu*. Tímto dojde k nahrání tabulky pouze do excelovského sešitu, a nikoliv už do datového modelu. V příkladě však budou s těmito daty prováděny další operace, proto je potřeba tabulku nahrát i do datového modelu. Nahrání tabulky bude demonstrováno dalším



z možných způsobů. Pokud máte naformátovanou tabulku již v sešitu, přejděte na kartu *Power Pivot* a ve skupině *Tabulky* klepněte na pole *Přidat do datového modelu*. Před importem je potřeba tabulku označit (klepnutím do libovolného pole tabulky levým tlačítkem myši).

## 7 Příklad 2: Úprava dat v Power Pivotu

V předchozích kapitolách bylo ukázáno, jak lze data před importem do Excelu upravit do přehledné a použitelné formy pomocí nástroje Power Query. Nyní jsou vstupní data nahrána a připravena pro další práci. Ta se bude odehrávat v prostředí Power Pivot Excel. V rámci této kapitoly se uživatel seznámí s prostředím Power Pivot, s importem externích dat, s jednotlivými nástroji, tvorbou relací mezi tabulkami a v neposlední řadě i s tvorbou počítaných sloupců a měř pomocí jazyka DAX.

Do prostředí Power Pivotu se dostanete klepnutím na stejnojmennou kartu a následným zvolením pole *Spravovat* ve skupině *Datový model*. Pokud se na liště karet nenachází karta *Power Pivot*, je třeba ji zpřístupnit v nastavení Excelu. Postup, jak zpřístupnit Power Pivot je popsán v praktické části (viz kapitola 4.3). Power Pivot se pak zobrazí v samostatném okně jako tomu bylo i v případě Power Query. Po otevření Power Pivotu si lze všimnout že prostředí mnohem více připomíná klasické rozhraní Excelu. V horní části se opět nachází pás karet s jednotlivými nástroji pro úpravu dat. Ve spod okna leží lišta s názvy jednotlivých dotazů, mezi nimi lze přepínat podobně jako mezi listy sešitu v Excelu. Prostředí Power Pivotu je pak podrobněji popsáno v Kapitole 4.3.

### 7.1 Nastíněný postup příkladu

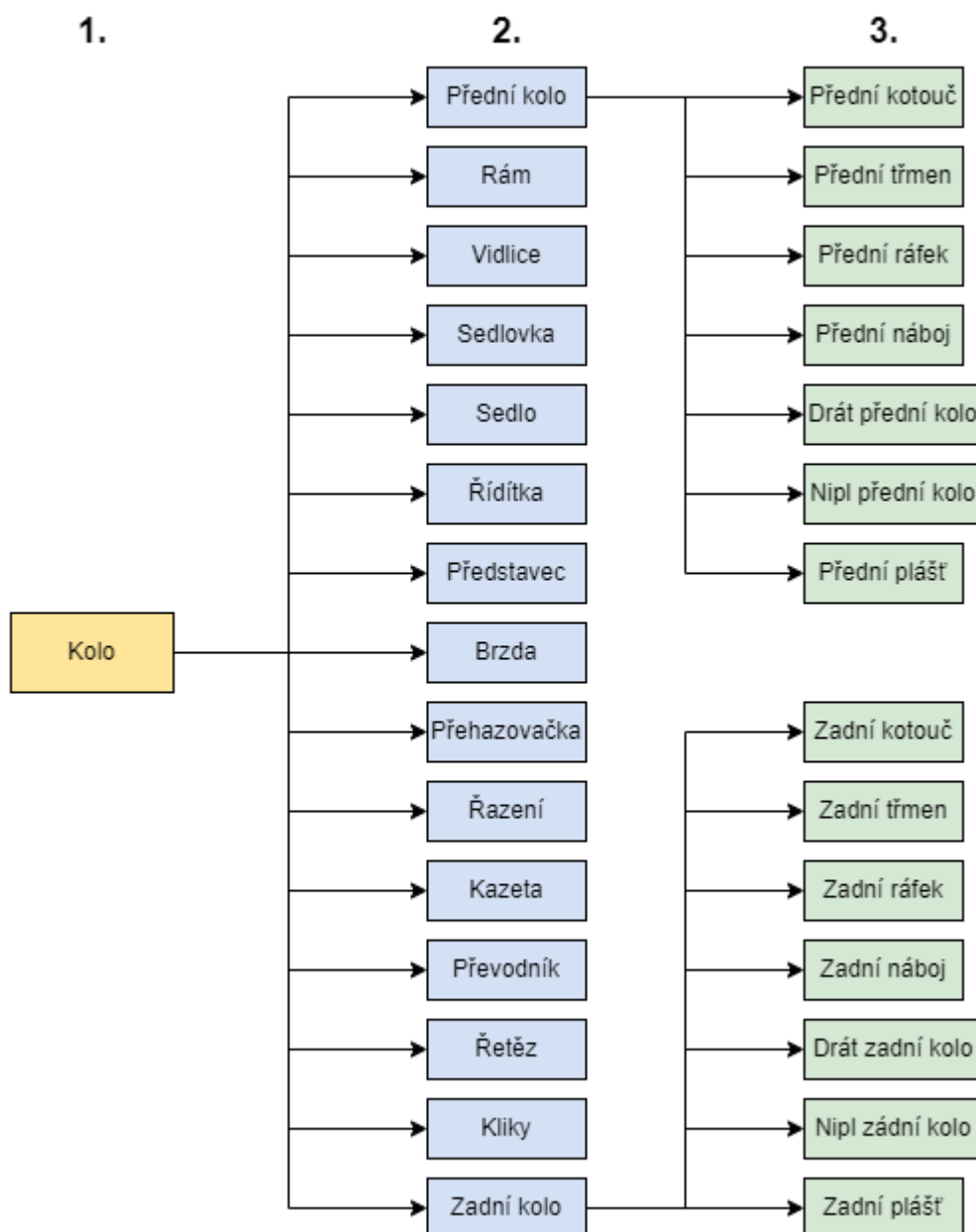
Po načtení okna Power Pivotu se zobrazí naimportované dotazy: *Položky*, *Pracoviště*, *ČasyOperací*, *Objednávky*, *Zákazník*. Pomocí těchto dat se budou počítat nové sloupce k určení cen jednotlivých položek obsažených v kusovníku (tabulka *Položky*), cen objednávek a dalších potřebných informací. Díky tomu bude možno data filtrovat v kontingenčních tabulkách a vizuálech např. podle jednotlivých úrovní kusovníku.

Data z tabulky *Položky* mají jasně definovanou hierarchii pomocí sloupců *PoložkaID* a *NadřazenáPoložka* (v anglické literatuře označována jako *parent-child hierarchy*). Sloupec *PoložkaID* udává pořadové číslo každé položce kusovníku. V rámci tohoto sloupce nedochází k výskytu stejné hodnoty dvakrát (každé položce je přiřazena právě jedna hodnota). To samé ale neplatí pro sloupec *NadřazenáPoložka*. Tento sloupec obsahuje číselné hodnoty, ze sloupce

PoložkaID	Položka	NadřazenáPoložka
1	Kolo	
2	Přední kolo	1
3	Zadní kolo	1
4	Rám	1

Obrázek 31: Definování hierarchie

*PoložkaID*, rovnající se nejbližší nadřazené položce pro položku v daném řádku. Např. položka *Kolo* je přímo nadřazené položkám *Přední kolo*, *Zadní kolo* a *Rám*. Jelikož *PoložkaID* položky *Kolo* je rovna jedné, hodnota ve sloupci *NadřazenáPoložka* u položek *Přední kolo*, *Zadní kolo* a *Rám* bude taktéž rovna jedné, viz *Obrázek 31*. Toto značení pokračuje dál pro všechny položky. Pro zadaný příklad se v kusovníku vyskytuje 30 položek rozdělených do tří úrovní. V první úrovni se nachází finální výrobek, *Kolo*, který se sestává z jednotlivých komponent. V druhé úrovni jsou komponenty, ze kterých se již montuje samotné kolo, konkrétně 15. V poslední, třetí, úrovni se nacházejí zbylé jednodušší komponenty, ze kterých se nejdříve vyrobí větší součásti (úroveň 2), ze kterých se později smontuje kolo (úroveň 1). Pro lepší představu je celá soustava vykreslena v následujícím diagramu viz *Obrázek 32*.



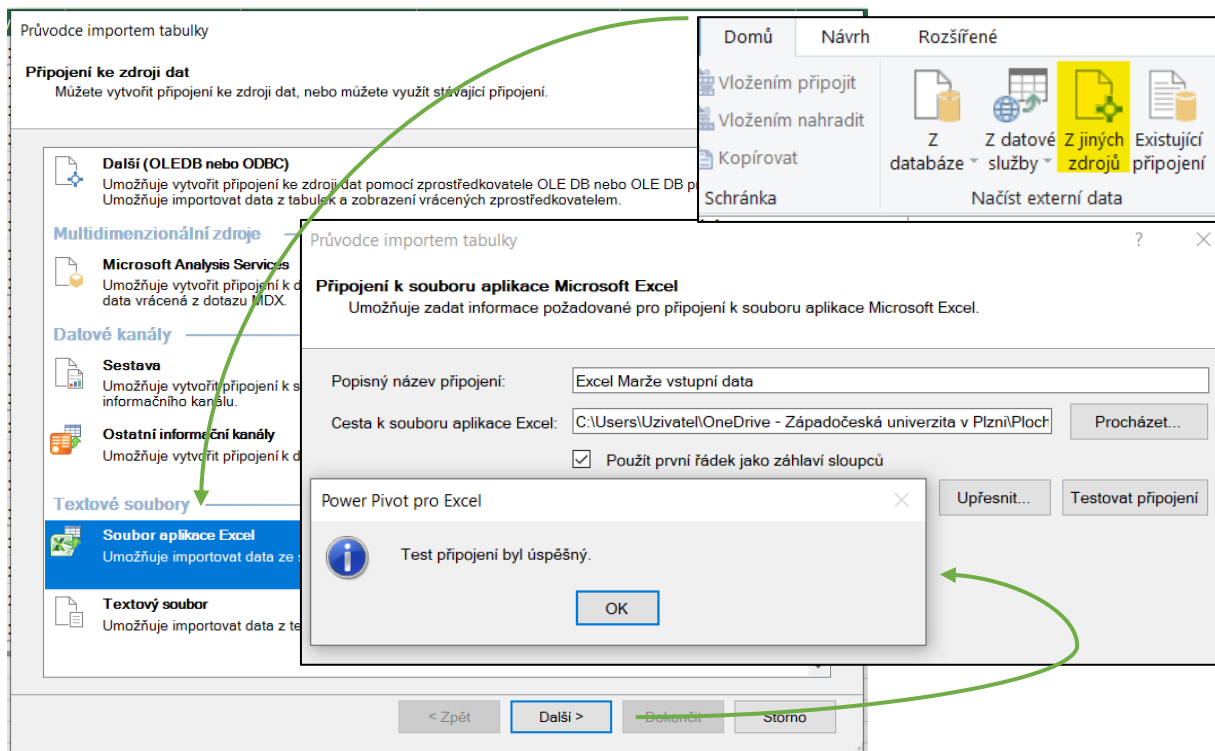
Obrázek 32: Schematický diagram jízdního kola – hierarchické úrovně

V dotazu *Položky* se dále nacházejí sloupce určující, zda se jedná o vyráběnou, či nakupovanou položku. Ačkoliv to není nikde přímo uvedeno, která položka je která, dá se to vyčíst z dat ve sloupcích *NákupníCena [1ks]* a *CenaMateriálu [1ks]*. Všechny nakupované položky kusovníku totiž mají ve sloupci *NákupníCena [1ks]* nenulovou hodnotu, respektive vyráběné položky mají nenulovou hodnotu ve sloupci *CenaMateriálu[1ks]*. Co se týče nakupovaných položek, tak u nich je jejich finální cena známa, tedy cena nakupované položky je rovna cenně uvedené v příslušném řádku sloupce *NákupníCena [1ks]*. Jelikož žádná nakupovaná položka v kusovníku se již neskládá z nižších položek, není potřeba k její cenně připočítávat i cenu položek ze kterých se položka skládá. To vše je již zahrnuto v nákupní cenně. Ovšem složitější situace nastává v případě vyráběných položek. Zde je potřeba vypočítat cenu jednotlivých položek na základě známých hodnot, tedy ceny materiálu, trvání operace (soustružení,

montáž...) a režijních nákladů na stroj a obsluhu. Jelikož tato data se nacházejí v rozdílných dotazech, je nutné nejprve vytvořit relace mezi jednotlivými dotazy.

## 7.2 Import dat

Před tím, než se začne s daty pracovat, je potřeba do datového modelu nahrát ještě jednu tabulku. Tentokrát jsou data v tabulce již v upraveném stavu, a tím pádem je možné je načíst rovnou do prostředí Power Pivotu. Asi nejjednodušší způsob načtení dat, je zvolení jedné z možností ze skupiny *Načíst externí data* přímo na domovské kartě Power Pivotu. Lze vybírat hned z několika možností, pro tento případ budou data importována ze sešitu Excelu. Pro nahrání patřičného souboru přejděte do zmíněné skupiny a klepněte na ikonu *Z jiných zdrojů* a z nabídky poté vyberte možnost *Soubor aplikace Excel*. Poté dojde k zobrazení karty Průvodce importem, ve které se vyplní umístění cílového souboru. Zvolte soubor *Marže vstupní data.xlsx* a tlačítkem *Testovat připojení* ověřte správnost připojení. Jelikož načítaná tabulka obsahuje záhlaví zvolte možnost *Použít první řádek jako záhlaví sloupců*. Pokud je vše v pořádku, klikněte na tlačítko *Další*, zvolte požadovanou tabulku (v tomto případě je pouze jedna) a potvrďte tlačítkem *Dokončit*, viz *Obrázek 33*.

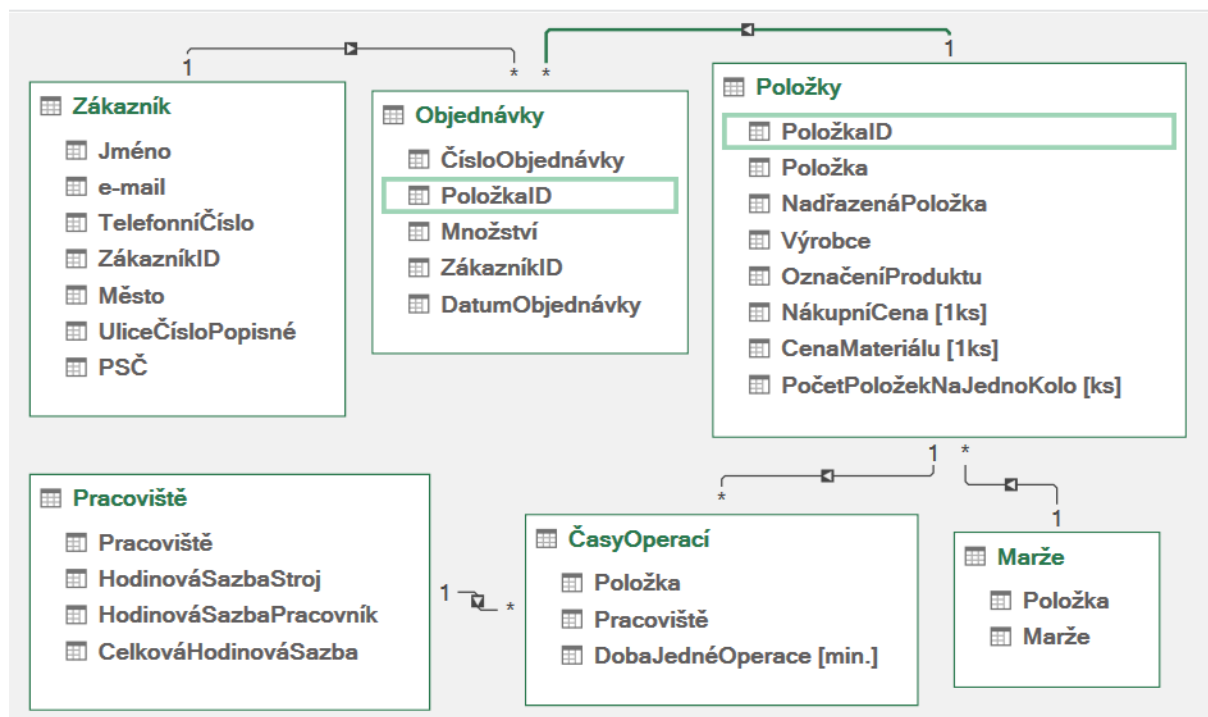


Obrázek 33: Import dat do prostředí Power Pivotu

Tabulku se úspěšně podařilo nahrát do datového modelu a lze jí vidět mezi zbylými dotazy. Tabulka obsahuje celkem čtyři sloupce, *Položka*, *Marže* a dva prázdné sloupce. Zmíněné prázdné sloupce zde nemají co dělat, tudíž je z datového modelu odstraníte (klepnutím pravého tlačítka myši na označené sloupce a zvolením možnosti *Odstranit sloupce*). Nakonec zbývá už jen přejmenovat dotaz na *Marže* a je hotovo (pomocí dvojklíku na název dotazu v liště dole).

### 7.3 Tvorba relací

Pomocí relací je možné pro výpočet sloupců a měř používat i data z jiných dotazů. V důsledku propojení jednotlivých tabulek vznikne vazba, která k sobě přiřazuje konkrétní hodnoty z propojených tabulek na základě klíče. K tomu, aby bylo možné vytvořit relaci mezi dvěma tabulkami, je nutné v každé z nich najít sloupec obsahující stejné hodnoty. Pro práci s relacemi je nutné se přepnout na *Zobrazení diagramu*. Toto tlačítko se nachází na kartě *Domů* ve skupině *Zobrazit*. Po přepnutí do okna diagramu se zobrazí jednotlivé dotazy v podobě „*dlaždice*“, každá z nich obsahuje název dotazu a názvy jednotlivých sloupců. Po nahrání dotazů do modelu nejsou data mezi sebou nikterak provázána, jsou tak vidět jen zmiňované „*dlaždice*“. Na obrázku níže jsou vykreslené dotazy již propojeny, relace jsou symbolizovány spojnicí mezi jednotlivými dotazy. Pro vytvoření relace stačí jednoduše vzít název sloupce z jednoho dotazu a přetáhnout jej na sloupec z druhého dotazu. Power Pivot pak následně vytvoří relaci mezi zvolenými dotazy a automaticky jí udělí typ. Z obrázku je zřejmé, že např. dotazy *Položky* a *Pracoviště* mají oba vazbu *jednu k mnoha* s tabulkou *ČasyOperací*. To je symbolizováno znaky jednotky a hvězdy na každém z konců spojnice. Po najetí kurzoru myši na relaci se označí sloupce, pomocí nichž se dané dotazy propojili (např. dotazy *Položky* a *Objednávky* jsou propojeny pomocí sloupců *Položka ID*, viz *Obrázek 34*).



Obrázek 34: Zobrazení diagramu

Pokud vám způsob tvorby relací nevyhovuje, lze relaci vytvořit taktéž přes klepnutí pravého tlačítka na zvolený dotaz a vybrání pole *Vytvořit relaci...* ze seznamu. Zobrazí se vám okno s manuální volbou dotazů a sloupců pomocí nichž mají být propojeny. Výhodou tohoto použití je, že uživatel vidí náhled daného sloupce, a tudíž při rozdílných názvech sloupců si může zkontrolovat, zdali hodnoty ve sloupcích jsou shodné. Zmíněné okno se taktéž zobrazí pomocí dvojkliku levým tlačítkem myši na již vytvořenou relaci, s tím rozdílem, že už je nadefinována. Ukázka manuální tvorby/úpravy relací je demonstrována na obrázku níže (viz *Obrázek 35*). Po Nastavení relací, viz *Obrázek 34*, se znovu přepněte do *Zobrazení dat* (skupina *Zobrazit*, karta *Domů*).

Úprava relace

Vyberte tabulky a sloupce, které se k sobě vztahují.

ČasyOperací

DobaJednéOperace [min.]	NázevDílu	Pracoviště
120	Kolo	Montáž
25	Rám	Pila
30	Rám	Svařovna
15	Rám	Brusárna
18	Přední kolo	Montáž

Položky

CenaMateriálu [1ks]	NadřazenáPoložka	NákupníCena [1ks]	OznačeníProduktu	PočetPoložekNaJednoKolo [ks]	Položka	PoložkaID	Výrobce
0	0	0	SuperSpeed 29 Pro	1	Kolo	1	VELO
0	1	0	320 Wheel Al pro 29 F	1	Přední kolo	2	VELO
0	1	0	320 Wheel Al pro 29 R	1	Zadní kolo	3	VELO
1200	1	0	HT Frame 29	1	Rám	4	VELO
0	1	5200	F 232 29	1	Vídlice	5	DT Swiss

Aktivní

OK Zrušit

Obrázek 35: Okno Úprava relace

## 7.4 Tvorba počítaných sloupců pomocí DAX

Po nastavení potřebných relací je možné se pustit do tvorby nových sloupců. V nadcházející kapitole bude představeno fungování jazyka DAX a jak pomocí něj vytvářet počítané sloupce. V rámci tohoto příkladu budou představeny pouze ukázkové výpočty, pro další práci s DAX je doporučeno nastudování informací z vícero zdrojů, pro plné pochopení problematiky.

### 7.4.1 Cena jedné operace

Nejprve bude potřeba vytvořit nový sloupec *CenaJednéOperace* v dotazu *ČasyOperací*. V tomto sloupci bude zobrazena cena za konkrétní operaci pro jeden kus vyráběné položky. Výsledek se získá vynásobením doby trvání jedné operace celkovou hodinovou sazbou (stroj + pracovník), je však potřeba dbát na jednotky. Pro vytvoření sloupce přejděte na dotaz *ČasyOperací* a dvakrát poklepejte do záhlaví sloupce *Přidat sloupec* (vždy se nachází za posledním sloupcem dotazu úplně vpravo). Tím dojde k vytvoření nového prázdného sloupce. Takto vytvořený sloupec se automaticky pojmenuje, k jeho přejmenování stačí znovu dvakrát poklepat do záhlaví, teď již vytvořeného sloupce a přejmenovat jej na *CenaJednéOperace*. Nyní už nezbyvá nic jiného než napsat funkci pro zobrazení patřičných hodnot. Označte si počítaný sloupec a poté klepněte do řádku nad ním se symbolem  $f_x$  na začátku. Do tohoto řádku pak vepište následující funkci:

```
=RELATED('Pracoviště'[CelkováHodinováSazba])*ČasyOperací[DobaJednéOperace [min.]]/60
```

V základě se jedná o funkci prostého násobení a dělení jednotlivých sloupců. Jelikož se ale násobí hodnotou z jiného tabulky, je potřeba použít funkci *RELATED*, ta navrací hodnotu

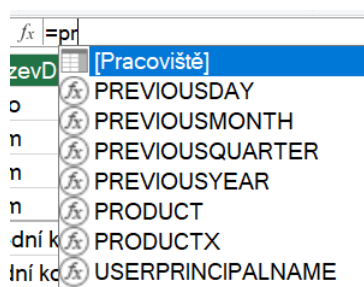
v relaci z jiné tabulky. Pro jasnou identifikaci sloupce se nejprve píše název dotazu do jednoduchých uvozovek a následně pak název sloupce do hranatých závorek. [17]

Po vepsání funkce a potvrzení pomocí tlačítka *Enter* se načtou hodnoty do všech řádků počítaného sloupce. Pokud funkce není správně definována, nebo kód obsahuje chyby, na místo hodnot se pak ve sloupci objeví text **#CHYBA**. Po aplikaci funkce by měl Dotaz *ČasyOperací* vypadat jako je tomu na obrázku (viz Obrázek 36). Jediné, co zbývá je upravit datový typ vypočteného sloupce na měnu (Kč). Toho docílíte klepnutím na šipku vedle ikonky dolaru ve skupině *Formátovat* na domovské kartě a zvolíte české koruny. Při pohledu na zmiňovaný obrázek si lze všimnout, že vypočtený sloupec má odlišnou barvu záhlaví oproti původně nahraným. Neznamena to, že by byl sloupec v nepořádku, označuje se tak pouze pro snazší rozlišení sloupců v tabulce (původní – zelené, vytvořené – černé).

	Praco...	NázevDílu	DobaJednéOperace [min.]	CenaJednéOperace
1	Montáž	Kolo	120	920,00 Kč
2	Pila	Rám	25	354,17 Kč
3	Svařovna	Rám	30	610,00 Kč
4	Brusírna	Rám	15	195,00 Kč

Obrázek 36: Finální podoba dotazu *ČasyOperací*

**TIP:** Při psaní kódu není potřeba vypisovat celé názvy ručně nebo vypisovat speciální znaky. Většinou, po vepsání počátečních písmen, se zobrazí nabídka s možnostmi. Pokud se zde vyskytuje funkce/název který hledáte, klepněte na něj myší a poté stiskněte klávesu *Tab*. Tím dojde k automatickému vepsání do kódu. Jak je vidět na obrázku (viz Obrázek 37), po vepsání písmen „pr“ došlo k zobrazení všech funkcí, názvů tabulek a sloupců obsahující tento kořen slova.



Obrázek 37: Práce s funkcemi v DAX

### 7.4.2 Výrobní náklady

Na základě znalosti ceny jedné operace, vypočtené v předchozí kapitole, je možné určit výrobní náklady vyráběných položek. K vypočtení takového sloupce však bude zapotřebí použití složitější funkce, sestávající se z několika funkcí vnořených do sebe. Požadavek je, aby funkce prošla všechny řádky sloupce *CenaJednéOperace* z tabulky *ČasyOperací* a sečetla časy operací pro každou položku kusovníku. Hodnoty se pak následně uloží do nově vytvořeného sloupce v tabulce *Položky*. V dotazu *Položky* vytvořte nový sloupec a pojmenujte jej *VýrobníNáklady [Iks]*, do něj následně opište tento kód:

```
=  
IF(  
  ISBLANK(  
    SUMX(  
      RELATEDTABLE('ČasyOperací');  
      'ČasyOperací'[CenaJednéOperace]  
    )  
  );  
  0;  
  SUMX(  
    RELATEDTABLE('ČasyOperací');  
    'ČasyOperací'[CenaJednéOperace]  
  )  
)
```

Kód je rozepsán do následující podoby z důvodu lepší přehlednosti. Jde o zažitou konvenci programátorů, kdy každý příkaz je vypsán na samostatné stránce. V praxi se preferuje tento zápis, nicméně lze kód zapsat jako v předchozí kapitole do jednoho řádku. Mezery a prázdné řádky nemají vliv na funkčnost kódu. Pro odřádkování funkce je potřeba použít kombinaci kláves **Shift** + **Enter**, samotný **Enter** by potvrdil rozepsanou funkci a došlo by k hlášení chyby.

Funkce **IF** ověřuje podmínku, zda je hodnota součtu (**SUMX**) sloupce *CenaJednéOperace* z připojené tabulky (**RELATEDTABLE**) podle jednotlivých položek nulová (**ISBLANK**), či nikoliv. Pokud **ANO**, vepíše 0, pokud **NE**, vepíše součet cen jednotlivých operací pro položku. Tím pádem funkce navrátí hodnotu součtu pro všechny vyráběné položky, a naopak u všech nakupovaných položek navrátí hodnotu 0. Nakupované položky se totiž nevyskytují v tabulce *ČasyOperací*. [17]

Po vepsání kódu získáte ve sloupci opět hodnoty datového typu *Desetinné číslo*. Stejně jako v předchozí kapitole (viz *Kapitola 7.4.1*), změňte datový typ sloupce na *Měna (Kč)*. Stejným způsobem změňte datový typ i pro sloupce *NákupníCena [Iks]* a *CenaMateriálu [Iks]* v dotazu *Položky*.

### 7.4.3 Celkové náklady

Poté, co je známa cena všech nakupovaných položek, materiálu a operací, je možné definovat sloupec ukazující cenu/náklady každé položky. Nový sloupec se jednoduše získá sečtením sloupců *NákupníCena [Iks]*, *CenaMateriálu [Iks]* a *VýrobníNáklady [Iks]*. Vytvořte nový sloupec a pojmenujte jej *CelkovéNáklady [Iks]*. Sloupec vypočtete pomocí následující funkce:



```
=  
'Položky'[NákupníCena [1ks]]  
+ 'Položky'[CenaMateriálu [1ks]]  
+ 'Položky'[VýrobníNáklady [1ks]]
```

Nezapomeňte převést datový typ na měnu v českých korunách.

**TIP:** na místo vepisování názvu sloupce do funkce, stačí klepnout do požadovaného sloupce a jeho název se do kódu vepíše automaticky.

#### 7.4.4 Celkové náklady na kolo

Jako další se vypočte sloupec obsahující data o ceně jednotlivých položek potřebných na smontování právě jednoho kola. Jednoduše se vynásobí cena položky počtem kusů obsažených v kusovníku. Pro vytvoření nového sloupce *CelkovéNákladyNaKolo* použijte následující funkci:

```
=  
'Položky'[CelkovéNáklady [1ks]]  
* 'Položky'[PočetPoložekNaJednoKolo [ks]]
```

Datový typ sloupce změňte na měnu v českých korunách.

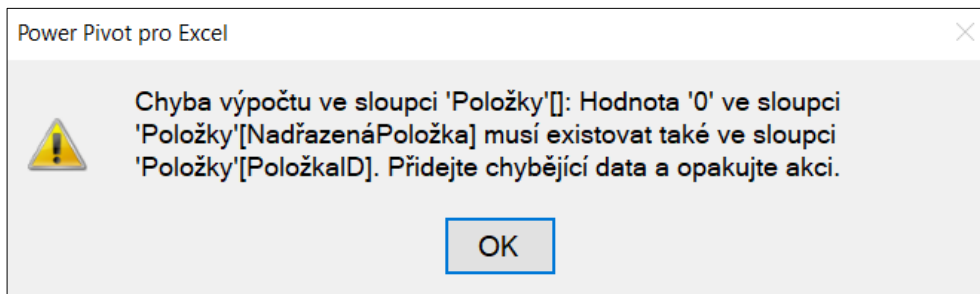
#### 7.4.5 Vytvoření sloupce kusovníkové hierarchie

V nadcházející kapitole bude nastíněn postup vytvoření sloupců pro definování hierarchie v kusovníku a určení vyšší/nížší položky. Jak již bylo popsáno v *Kapitole 7.1*, kusovníková hierarchie je definována sloupci *PoložkaID* a *NadřazenáPoložka*. Pro další práci je však potřeba rozepsat položky do jednotlivých úrovní, viz *Obrázek 32*. K tomu bude použito funkce **PATH**, která pracuje s tzv. *Parent-Child* hierarchií. Funkce vepíše všechny nadřazené položky (*Parents*) k položce v daném řádku. Ty budou zaznamenány pomocí pořadových čísel kusovníku (sloupec *PoložkaID*) a odděleny svislou čarou. Pořadí jednotlivých úrovní jde chronologicky zleva doprava s tím, že první číslo určuje nejvyšší úroveň kusovníku (v tomto případě kolo). Naopak poslední číslo vždy definuje položku v daném řádku a jedná se o položku, která není nadřazenou pro žádnou z dalších položek (*Child*). [17]

Vytvořte nový počítaný sloupec, do kterého opište následující kód:

```
=  
PATH(  
    'Položky'[PoložkaID];  
    'Položky'[NadřazenáPoložka]  
)
```

Po vepsání a potvrzení funkce, dojde v řádcích sloupce k zobrazení textu #CHYBA. Nyní je třeba zjistit, kde ona chyba nastala, při najetí kurzoru na sloupec se zobrazí nabídka s možností *Zobrazit chybu...*, po rozkliknutí dojde k zobrazení okna s popisem konkrétní chyby, která nastala. V tomto případě se v jednom ze sloupců vyskytla hodnota, která se nevyskytuje v druhém sloupci, na který odkazuje (viz Obrázek 38), tím pádem je potřeba hodnotu ve sloupci upravit.



Obrázek 38: Chyba funkce PATH

K tomu, aby bylo možné upravit hodnotu v sloupci *NadřazenáPoložka*, je zapotřebí přejít zpět do prostředí Power Query. V prostředí Power Pivotu není možné upravovat originální data ze zdroje, protože se v něm fyzicky nenacházejí, jsou k němu pouze připojena ze svých zdrojů. Tím pádem je nutné upravit data ve zdrojovém dokumentu, a to například v nástroji Power Query.

Nejprve uložte provedený postup a zavřete okno Power Pivotu. Nyní přejděte do prostředí Power Query, známým postupem přes kartu *Data, Dotazy a Připojení* a zvolte dotaz *Položky*. V sloupci *NadřazenáPoložka* je zapotřebí se zbavit „0“ v prvním řádku. V rámci *Kapitoly 6.1.2.1* byla učiněna operace, ve které se nahradili prázdné hodnoty ve všech sloupcích tabulky, ve kterých byly obsaženy. Nyní je potřeba učinit reverzní operaci, tedy změnit „0“ na prázdnou hodnotu. Pomocí návodu ve zmíněné kapitole nahraďte hodnotu „0“ ve sloupci *NadřazenáPoložka* hodnotou „null“, symbolizující prázdnou hodnotu (viz Obrázek 39).

### Nahradit hodnoty

Ve vybraných sloupcích nahradí jednu hodnotu druhou.

Hledaná hodnota

Nahradit hodnotou

OK

Zrušit

Obrázek 39: Nahrazení hodnoty (Power Query)

Další z možností, jak provést tuto operaci je, přejít do seznamu použitého postupu a vymazat krok nahrazení prázdné hodnoty za „0“. Má to však několik háčeků. Zaprvé, pokud jste v tomto kroku označili vícero sloupců najednou, změna se bude týkat všech zvolených sloupců. Zadruhé, je zde riziko, že smazání kroku z postupu ovlivní kroky následující a data tak nebudou naformátována tak, jak měla být.

Po nahrazení hodnoty zpět na prázdné pole, klikněte na ikonku diskety v levém horním rohu (*Zavřít a načíst*) a vraťte se zpět do prostředí Power Pivot. Klepněte na pole *Aktualizovat*

nacházející se na domovské kartě. Nyní by měl první řádek upraveného sloupce být prázdný, tím pádem by funkce **PATH** měla do stejnojmenného sloupce vypsat požadované hodnoty.

V dalším kroku rozdělte vytvořený sloupec po jednotlivých úrovních (co sloupec to úroveň), tím vzniknou tři nové sloupce. Pro rozdělení sloupce **PATH** bude použita funkce **PATHITEM**, která do nového sloupce vepíše pro každý řádek hodnotu ze zvolené úrovně, pokud v dané úrovni nějakou hodnotu má. V opačném případě navrátí prázdnou hodnotu. V dotazu *Položky* vytvořte sloupce **PATHLevel1**, **PATHLevel2** a **PATHLevel3**. Do prvního ze sloupců vložte funkci:

```
=
PATHITEM(
    'Položky'[PATH];
    1;
    1
)
```

Kde funkce **PATHITEM** navrátí hodnoty z 1. úrovně sloupce **PATH**, a to v podobě typu *integer* (celé číslo). Právě první z číslic ve funkci udává úroveň hierarchie a druhé číslo zase datový typ ve kterém se data zobrazí. Pro další dva sloupce aplikujte stejný vzorec akorát s tím rozdílem, že první z číslic změňte podle hledané úrovně na „2“ a poté „3“. Druhé z čísel zůstává stejné, pokud by bylo třeba data nahrát v podobě typu *string* (text), vepíše se namísto jedničky nula. [17]

PoložkaID	Položka	PATH	PATHLevel1	PATHLevel2	PATHLevel3
1	Kolo	1	1		
2	Přední kolo	1 2	1	2	
3	Zadní kolo	1 3	1	3	
17	Přední kotouč	1 2 17	1	2	17
18	Zadní kotouč	1 3 18	1	3	18

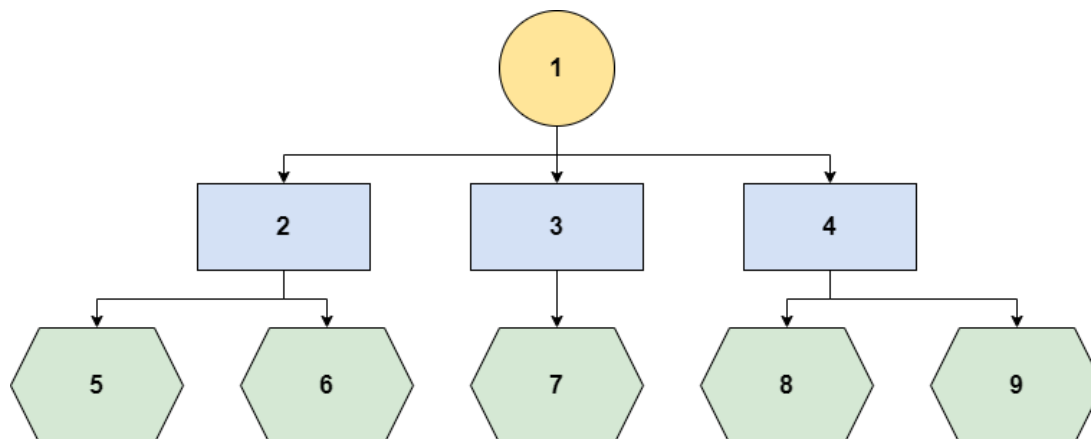
Obrázek 40: Výsledek použití funkce **PATH** a **PATHITEM**

Výsledkem by měly být 4 sloupce (**PATH**, **PATHLevel1**, **PATHLevel2** a **PATHLevel3**) definující, v jaké úrovni se daná položka nachází a jaké má vyšší/nížší položky. Vypočtené sloupce by pak měli vypadat, jako je tomu na obrázku (*viz Obrázek 40*). V obrázku jsou použity jen některé sloupce z tabulky pro lepší přehlednost, dále jsou vyfiltrovány jen některé položky pro lepší demonstraci. Při pohledu do řádku položky *Přední kotouč* je vidět, že se položka nachází ve 3. úrovni (ID položky se nachází ve třetím sloupci rozkladu). Jelikož třetí úroveň je poslední, znamená to, že *Přední kotouč* již nemá žádnou podřazenou položku. Na druhou stranu ze sloupce **PATHLevel2** vyplývá, že nadřazenou položkou pro *Přední kotouč* je *Přední kolo* (podle ID položky). Nakonec, celé *Přední kolo* má nadřazenou položku *Kolo* znázorněné hodnotou „1“ ve sloupci **PATHLevel1**, která odpovídá ID položky *Kolo*. Stejným postupem lze analyzovat všechny ostatní položky. Např. *Zadní kotouč* má nadřazenou položku *Zadní kolo* a ta zase položku *Kolo*.

#### 7.4.6 Cena položky

Cílem této kapitoly je, vytvořit sloupec, kde v každém řádku bude zobrazena skutečná cena každé položky. V ceně dané položky by měly být zahrnuty i ceny všech nižších položek, pokud položka nějaké má. Na obrázku níže (*viz Obrázek 41*) se nachází ukázka zjednodušeného diagramu kusovníku, podobného tomu počítanému. Pro určení ceny jednotlivých polí (1-9) je

nejprve nutné analyzovat z jakých dílčích cen se jednotlivé položky skládají. Nejjednodušší postup je začít od položek v nejnižší úrovni, pro ty platí, že mají pouze nadřazené položky. V případě zjednodušeného diagramu (viz Obrázek 41) se jedná o zelená hexagonální pole. Jejich cena je rovna nákupní ceně nebo celkovým výrobním nákladům (materiál + výroba) v závislosti, zda je položka vyráběná či nakupovaná. Při určování ceny položek ve druhé úrovni



Obrázek 41: Ilustrační schematický diagram hierarchické kusovníkové úrovně

(modré obdélníky) se však situace lehce komplikuje. K ceně samotné položky (např. náklady na montáž) už je třeba přičíst i cenu nižších položek. Tímto způsobem se získají ceny všech polí v druhé úrovni a stejný princip se opakuje pro vyšší úrovně.

Zkuste spočítat cenu položky „4“ a „1“ z diagramu, pokud cena každé položky odpovídá číslu zobrazeném na položce (např. cena pole „6“ = 6Kč). **Výsledek:** „4“ = 21Kč; „1“ = 45Kč

Zpět k příkladu. Z diagramu, viz. Obrázek 32, je zapotřebí zjistit cenu položek *Přední kolo*, *Zadní kolo* a *Kolo*. Pro zbylé položky je cena známa. Jelikož ale Power Pivot aplikuje funkce vždy na celé sloupce, není tato informace až tak důležitá, protože počet neznámých položek neovlivní tvar funkce.

Před vypočtením sloupce s cenami položek se nejprve provede výpočet pro jednotlivé úrovně. Vzniknou tak tři nové sloupce, každý definující cenu položek vyskytující se právě v dané úrovni. Hodnoty z těchto sloupců se pak sečtou do již zmiňovaného sloupce určujícího ceny položek. Vytvořte nový sloupec s názvem *CenaLevel1* a vepište do něj následující funkci:

```

=
VAR HledanaHodnota = 'Položky'[PoložkaID]
VAR ShodneRadky = FILTER(
    'Položky';
    'Položky'[PATHLevel1] = HledanaHodnota
)
RETURN
SUMX(
    ShodneRadky;
    'Položky'[CelkovéNákladyNaKolo]
)
    
```

Funkce projde řádky sloupce *PATHLevel1*, ve kterém se vyskytují číselné hodnoty odpovídající pořadí položek v kusovníku (*PoložkaID*). Postupně najde všechny řádky, které obsahují konkrétní hodnotu ze sloupce *PoložkaID*, následně sečte hodnoty ze sloupce *CelkovéNákladyNaKolo* odpovídající nalezeným řádkům. Součet těchto hodnot pak vepíše do počítaného sloupce, do řádku odpovídajícímu hledané hodnotě *PoložkyID*. Například, funkce hledá všechny řádky sloupce *PATHLevel1* obsahující hodnotu „1“, která odpovídá položce *Kolo*. Jelikož v úrovni jedna se nachází pouze položka *Kolo*, viz *Obrázek 32*. Všechny řádky sloupce obsahují jen a pouze hodnotu „1“. Následně funkce přejde do sloupce *CelkovéNákladyNaKolo* a sečte hodnoty nacházející se v nalezených řádcích. Výsledný součet se pak vepíše do 1. řádku počítaného sloupce odpovídající položce *Kolo*. [17]

Stejný postup aplikujte na další dva sloupce, tedy sloupce *CenaLevel2* a *CenaLevel3*. Funkce se liší pouze tím, že prochází jiný sloupec. V důsledku toho je nutné ve funkci přepsat název sloupce *PATHLevel1* na *PATHLevel2*, respektive *PATHLevel3*.

Po vytvoření všech tří sloupců jsou známy ceny všech položek kusovníku. Ceny položek jsou však vepsány napříč třemi nově vytvořenými sloupci, každá ve sloupci odpovídající úrovni, ve které se položka nachází. Je tedy třeba tato data dostat do jednoho sloupce, kde budou všechna pohromadě. Vytvořte tedy nový počítaný sloupec *Cena*, který se vypočítá následující funkcí:

```
=('Položky'[CenaLevel1]  
+'Položky'[CenaLevel2]  
+'Položky'[CenaLevel3]  
)  
/'Položky'[PočetPoložekNaJednoKolo [ks]]]
```

Sloupec *Cena* se získá prostým sečtením tří sloupců *CenaLevel1*, *CenaLevel2*, *CenaLevel3* a následným vydělením hodnotami ze sloupce *PočetPoložekNaJednoKolo*. Jelikož předešlé tři sloupce byly vypočteny z nákladů pro celé kolo, je třeba jejich výsledek podělit právě počtem kusů, potřebných na montáž jednoho kola. Pokud by tak nebylo učiněno, cena položky by neodpovídala jednomu kusu, ale všem kusům potřebných k montáži jednoho kola. U většiny dílů zůstane cena stejná, ale např. u položky *Drát přední kolo* se cena změní ze 14Kč na 0,5Kč, jelikož na montáž předního kola je potřeba 28 drátů.

#### 7.4.7 Zisk z prodeje

Nyní, když jsou známy ceny jednotlivých položek, je možné určit zisk z prodeje každé položky. Cílem této kapitoly bude vypočítat zisk z každé dokončené objednávky. K tomu, aby bylo možné vypočítat zisk, je nutné znát cenu nákupu/výroby a cenu prodeje. V předchozí kapitole byla vypočtena cena jednotlivých položek kusovníku (sloupec *Cena*), tato cena představuje nákupní cenu, respektive cenu na výrobu položky. Prodejní cena položky je však neznámá. K výpočtu prodejní ceny, a tedy i zisku bude zapotřebí použít marži, která je zadána ve stejnojmenné tabulce. Marže je definována podle vzorce (1) jako podíl zisku a prodejní ceny. Na základě tohoto vzorce lze odvodit nový vztah (2) pro výpočet prodejní ceny. Uvedené vzorce platí pro hodnoty marže typu desetinné číslo, při zadání marže v procentech je třeba pravou stranu rovnice (1) vynásobit stem. [18]

$$\text{Marže} = \frac{\text{Prodej} - \text{Nákup}}{\text{Prodej}} \quad (1)$$

$$\text{Prodej} = \frac{\text{Nákup}}{1 - \text{Marže}} \quad [\text{Kč}] \quad (2)$$

Na základě rovnice pro prodejní cenu vypočtete nový sloupec *ProdejníCena* v dotazu *Položky*. Pro výpočet použijte následující vzorec:

='Položky'[CenaPoložky]/(1-RELATED('Marže'[Marže]))

K navrácení hodnot z tabulky *Marže* je zapotřebí použít funkci **RELATED**.

Poté, co jsou známy ceny nákupu/výroby a prodeje všech položek kusovníku, je na řadě určit ceny jednotlivých objednávek. Za použití funkce **RELATED** a základních matematických operací vytvořte dva nové sloupce v dotazu *Objednávky*. Nové sloupce vytvoříte pomocí následujících funkcí:

Cena:=RELATED('Položky'[Cena])  
\*'Objednávky'[Množství]

ProdejníCena:=RELATED('Položky'[ProdejníCena])  
\*'Objednávky'[Množství]

Požadovaný sloupec *Zisk* se pak dostane odečtením sloupce *Cena* od sloupce *ProdejníCena*. Výsledná funkce pak vypadá následovně:

='Objednávky'[ProdejníCena] - 'Objednávky'[Cena]

#### 7.4.8 Funkce datum a čas

Následující kapitola je zaměřena na využití jazyka DAX a funkcí které nabízí, k úpravě dat typu datum, respektive čas. Konkrétně se budou tyto úpravy týkat sloupce *DatumObjednávky* v dotazu *Objednávky*. K tomu, aby bylo možné data ve vizualizacích a kontingenčních tabulkách jednoduše filtrovat, je vhodné informaci o datu/času rozložit na vícero dílčích informací. Uživatel se pak lépe orientuje v datech a může jednoduše pomocí filtrů sloupce nahlédnout, kolik záznamů tabulka má např. v daném měsíci.

##### Měsíc

Prvním krokem bude vytvoření sloupce udávající měsíc, kdy byla objednávka vytvořena. Je zde požadavek, aby hodnoty byly zobrazeny jako text. Některé hodnoty, jako třeba právě měsíce, lze zobrazit jak v číselném formátu, tak i jako text. Vytvořte tedy nový počítaný sloupec s názvem *Měsíc*, který vepíše název konkrétního měsíce podle následující funkce:

Měsíc: =FORMAT('Objednávky'[DatumObjednávky],"mmmm")

V podstatě jde jen o upravení data do požadovaného formátu, podobně jako tomu je v Power Query. Na rozdíl od Power Query, zde zůstal původní sloupec ve stejném formátu a definované změny se aplikovali v nově vytvořeném sloupci. Funkce **FORMAT** tedy vezme data ze zvoleného sloupce (*DatumObjednávky*) a podle definovaného formátu („*mmmm*“) vloží data do nového sloupce. Výsledkem jsou hodnoty ve formátu *leden, únor, březen atd.* Písmeno „*m*“ z definice formátu, udává že se jedná o měsíc (z anglického *month*) a počet písmen zase formát ve kterém má být hodnota zobrazena. Zkuste pozměnit formát funkce postupně na „*mmm*“, „*mm*“ a „*m*“. Všimněte si, že pokud zadáte pouze jedno písmeno, zobrazí se hodnoty jako pořadové číslo měsíce. Pomocí funkce **FORMAT** tak lze nadefinovat různé zobrazení dat různých datových typů. [17]

## Číslo Týdne

Další ze zajímavých funkcí v nabídce DAX, je funkce **WEEKNUM**. Ta navrácí hodnotu, která už není tak jednoznačně patrná z datumu, a tou je číslo týdne. Pokud je tedy potřeba vědět informace v rámci jednotlivých týdnů, je vhodné použít právě tuto funkci. [17]

Do nově vytvořeného sloupce vložte následující kód s využitím funkce **WEEKNUM**:

```
ČísloTýdne:=WEEKNUM('Objednávky'[DatumObjednávky];2)
```

U funkcí pracujících s týdny je třeba myslet na to, že ne ve všech kulturách je definice týdne jednotná. Např. v některých anglicky mluvících zemích týden začíná nedělí a končí sobotou. Tím pádem je důležité správně definovat začátek a konec týdne. Číslo „2“ v rámci kódu uvedeného výše definuje týden, tak jak je v Česku zvykem. Při psaní kódu vyjede list možností s popisem jednotlivých variant, tudíž není třeba si čísla pamatovat z hlavy.

## Pořadí dne v týdnu

Následuje funkce velmi podobná té předchozí, s tím rozdílem, že nenavrácí čísla týdnů, ale čísla dnů v týdnu. Jedná se o funkci **WEEKDAY**. [17]

Pro určení čísla symbolizující den v týdnu opět vytvořte nový sloupec a vložte následující funkci:

```
PořadíDneVTýdnu:=WEEKDAY('Objednávky'[DatumObjednávky];2)
```

Funkce obsahuje stejné hodnoty jako ta předchozí. Jediný rozdíl, který není vidět je, že nyní je na výběr ze tří možností formátu týdne. Kromě lišícího se začátku a konce týdne se zde liší i číslování, to lze nastavit od 1 do 7, ale také od 0 do 6. Pro středoevropský formát však stále platí číslo „2“ ve vzorci.

## 7.5 Výpočet míry

Kromě počítaných sloupců lze použít jazyk DAX také k výpočtu měr. Míry se od sloupců liší tím, že zpravidla reprezentuje souhrn dat ze sloupci, či zvolených sloupců. Základním příkladem takové míry může být např. součet hodnot, průměrná hodnota, modus, medián atd. Míry jsou dalším z velmi důležitých prvků pro vytvoření dobrých vizuálů. V některých případech se lze dostat ke stejné hodnotě počítáním nového sloupce, ale většinou se preferuje v takovém případě vytvořit míru. Díky tomu že se míra vepíše pouze jednou do příslušného pole, nedochází ke zbytečnému zvětšování modelu. Dalším z důvodů, proč používat míry je ten, že v některých případech není jiná možnost (ani výpočtem nového sloupce) jak se k potřebným datům dostat.

V rámci tohoto příkladu budou vytvořeny 3 jednoduché míry pro demonstraci jejich tvorby. Tyto míry budou následně zakomponovány do kontingenčních tabulek a grafů.

### 7.5.1 Průměrná cena objednávky

První míra bude obsahovat informaci o průměrné ceně objednávky. Na základě této hodnoty je možno porovnávat jednotlivé objednávky a vyhodnocovat o kolik se lišily jejich ceny od té průměrné. Pro vytvoření míry přejděte do okna Power Pivotu a ve skupině *Zobrazit* na domovské kartě zaškrtněte pole *Oblast Výpočtu*. Nyní by se měla zobrazit prázdná část sešitu v dolní části okna. Pomocí kurzoru myši lze upravovat její výšku v závislosti na potřebách uživatele. Oblast výpočtu lze použít pro výpočet nových měr a psaní poznámek. Míru pak jednoduše vytvoříte vepsáním funkce do libovolného pole. Z pravidla se míry umísťují pod sloupec, ze kterého byla použita data pro výpočet.

Pro výpočet míry *PrůCenaObj* vložte do libovolného pole, pod sloupcem *Prodejní Cena* v dotazu *Objednávky*, následující kód:

$$\text{PrůCenaObj} = \text{AVERAGE}([\text{ProdejníCena}])$$

Pro psaní kódu opět platí stejná pravidla jako u počítaných sloupců. Nezapomeňte za název míry připojit dvojtečku před rovnítko, aby nedošlo k chybě. Funkce **AVERAGE** navrátí průměrnou hodnotu ze sloupce *ProdejníCena*. [17]

### 7.5.2 Celkový prodej

Pro vytvoření druhé míry se bude postupovat obdobným způsobem. Cílem je vytvořit míru obsahující informaci o celkové ceně prodaného zboží za celé období. Jedná se o sečtení hodnot ze sloupce *ProdejníCena* v dotazu *Objednávky*. Opět umístěte míru pod sloupec *ProdejníCena*. Pro vytvoření sloupce použijte následující kód:

$$\text{CelkovýProdej} = \text{SUM}(\text{'Objednávky'}[\text{ProdejníCena}])$$

### 7.5.3 Počet kusů

Poslední z potřebných měř je míra obsahující informaci o celkovém počtu prodaných kusů. Pro tento případ bude použit jiný ze způsobů tvorby měř.

V okně Excelu přejděte na kartu *Power Pivot* a v záložce *Výpočty* zvolte možnost *Nová míra...*, dojde k zobrazení okna, ve kterém se nadefinuje nová míra. Pro vytvoření nové míry *PočetKusů* vyplňte tabulku podle obrázku níže (*Obrázek 42*) a potvrďte tlačítkem *OK*.

Míra

Název tabulky:

Název míry:

Popis:

Vzorec:

Možnosti formátování

Kategorie:

Formát:

Použít oddělovač tisíců (.)

Obrázek 42: Výpočet nové míry



## 8 Příklad 3: Tvorba kontingenčních tabulek a grafů

Následující kapitola se zaměřuje na tvorbu a práci s kontingenčními tabulkami a grafy (dále jen jako *kont. tabulkami*). Ty mají za úkol čtenáři poskytnout jednoduchý a přehledný přístup ke konkrétním datům obsažených v databázi. Jako zdrojová data pro tvorbu kont. tabulek a grafů bude použita vytvořená databáze v sešitu *Příklad kola*. Z databáze vytvořené v Power Pivotu budou postupně nahrána potřebná data do kont. tabulek v excelovském sešitu. V těchto tabulkách bude shrnut např. přehled tržeb za jednotlivá období. Tato data budou následně zobrazena v kont. grafech pro lepší představení.

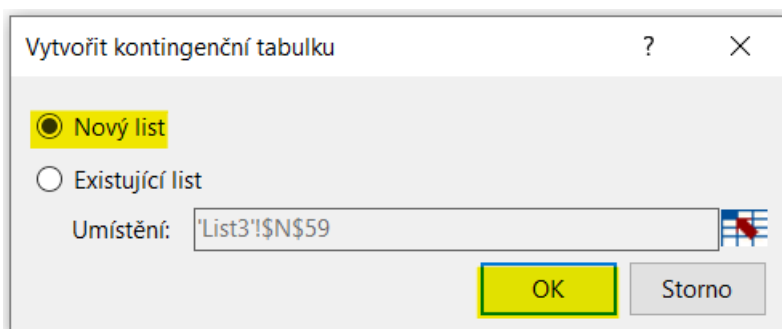
### 8.1 Zisk za jednotlivá období

Prvním úkolem bude vytvořit kont. tabulku a graf zobrazující zisky za určité období, konkrétně za jednotlivé měsíce a dny v týdnu. Pomocí těchto dat lze získat přehled o tom, kdy se podniku daří prodávat a kdy naopak klesá poptávka po produktech. Na základě analýzy těchto dat napříč jednotlivými obdobími, lze predikovat budoucí vývoj tržeb a zisku. Například co se týče prodeje kol a jejich komponentů, zcela jistě budou zisky vyšší během letní sezóny než během zimních měsíců. Dále mohou prodeje ovlivnit například dny v týdnu nebo třeba i počasí (převážně, jedná-li se o kamennou prodejnu a ne e-shop)

V rámci příkladu jsou poskytnuta data za první pololetí 2023. Jelikož všechny objednávky byly provedeny ve stejném roce, budou zisky rozděleny pouze do měsíců a dnů v týdnu. V případě, že by se v databázi vyskytovali objednávky z let předešlých bylo by nutné zisk rozdělit i v rámci jednotlivých let.

#### 8.1.1 Kontingenční tabulka Zisk za jednotlivá období

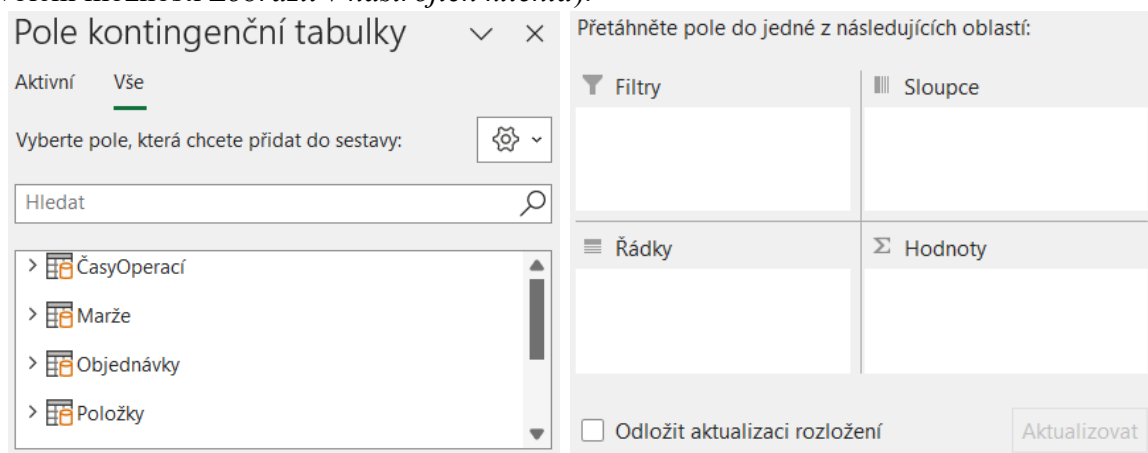
K vytvoření nové kont. tabulky přejděte do prostředí Power Pivotu a na domovské kartě klikněte na pole *Kontingenční tabulka*. Dojde k přepnutí do prostředí excelovského sešitu a zobrazení okna *Vytvořit kontingenční tabulku*. Ve zmíněném okně pak zvolte variantu *Nový list* a potvrďte výběr tlačítkem *OK*, viz. *Obrázek 43*. Tímto krokem došlo k nahrání prázdné kont. tabulky do nového listu sešitu *Příklad kola.xlsx*. Zvolením varianty *Existující list* lze vybrat pozici na již vytvořeném listě, kam se má tabulka naimportovat. V případě tohoto příkladu bude každá kont. tabulka zobrazena na novém listu pro lepší přehlednost.



Obrázek 43: Vytvoření kontingenční tabulky

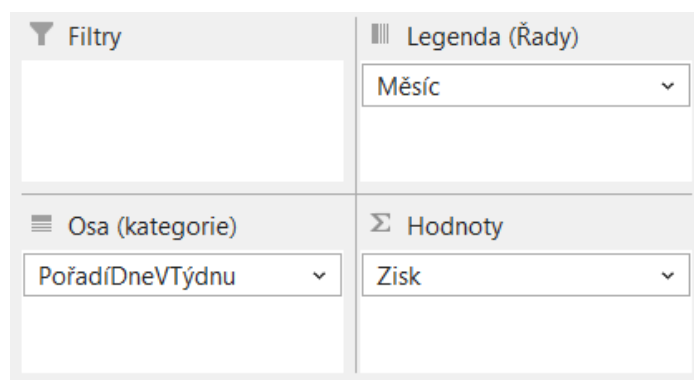
Nově vytvořená kont. tabulka neobsahuje žádná data, ta je třeba vybrat a ručně definovat ze *Seznamu polí*, viz *Obrázek 44*. Seznam polí je lišta obsahující list všech tabulek/dotazů nahraných v Power Pivotu a jednotlivá pole pro definování tabulky (*Filtry*, *Sloupce*, *Řádky a Hodnoty*). Lišta se seznamem polí se automaticky zobrazí po vytvoření kont. tabulky. Pokud se tak nestane, je možné ji zapnout pomocí tlačítka *Seznam polí* ve skupině *Zobrazit* na kartě *Analýza kontingenční tabulky*.

Po rozkliknutí šipky u libovolného dotazu se zobrazí seznam všech aktivních sloupců a měř daného dotazu. Pokud bude třeba pracovat s daty, které jsou v prostředí Power Pivotu skryta, je zapotřebí je nejprve znovu aktivovat (klepnutím pravého tlačítka myši na skrytý sloupec a zvolení možnosti *Zobrazit v nástrojích klienta*).



Obrázek 44: Lišta Seznam polí

Pro vytvoření kont. tabulky *Zisk za jednotlivé období*, budou použita následující data: *Měsíc*, *PořadíDneVTýdnu* a *Zisk* (všechna z dotazu *Objednávky*). Rozklikněte dotaz *Objednávky* v seznamu polí a postupně přetáhněte zmíněná data do příslušných polí, viz *Obrázek 45*. Nyní se zvolená data zobrazí v kont. tabulce. První sloupec obsahuje čísla od 1 do 7, která představují jednotlivé dny v týdnu. Další sloupce pak reprezentují jednotlivé měsíce a zobrazují sumu zisků pro každý den týdne (po, út, ...). Poslední řádek a sloupec tabulky obsahuje součet zisků za jednotlivé měsíce, respektive dny v týdnu. Tato kont. tabulka udává celkový zisk za jednotlivé



Obrázek 45: Hodnoty pro kontingenční tabulku *Zisk za jednotlivé období*

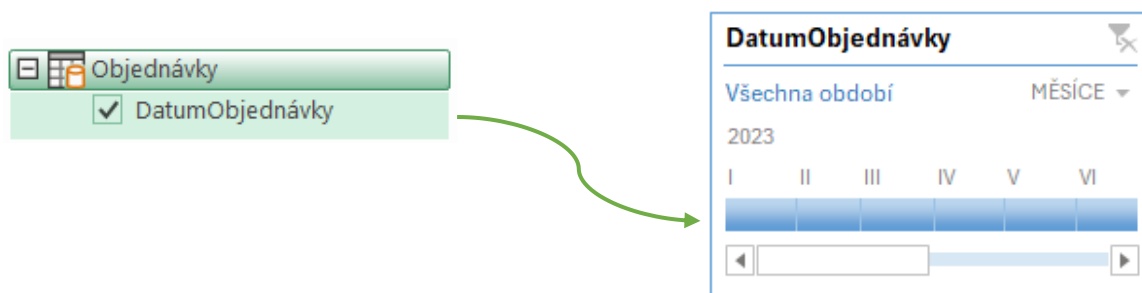
dny v měsíci, tedy např. celkový zisk na všechny pondělky v lednu. Tabulka byla vytvořena za účelem demonstrace rozdělení dat pro jednotlivé dny v týdnu, nicméně v praxi tato data nebudou mít příliš vypovídající hodnotu, a to zejména kvůli proměnnému počtu dní v měsíci. Pro lepší výpovědní hodnotu se mohou data v prvním sloupci nahradit např. za jednotlivé týdny v měsíci, nebo přímo konkrétní dny.

Zobrazenou kont. tabulku zbývá už jen naformátovat do vhodné podoby a je hotovo. Na kartě *Návrh* nastavte design podle vlastního uvážení, tak aby byla zachována přehlednost kont. tabulky. Dále přejmenujte záhlaví sloupců dle obrázku (viz *Obrázek 46*).

Zisk za jednotlivá období	Měsíce							
Den v týdnu	led	úno	bře	dub	kvě	čvn	Celkový součet	
1	2 489,06 Kč	11 193,13 Kč	31 503,84 Kč	180,51 Kč	3 039,87 Kč	2 196,03 Kč	50 602,45 Kč	
2	3 433,13 Kč	21 154,04 Kč	22 720,56 Kč	2 112,00 Kč	260,19 Kč	11 362,93 Kč	61 042,85 Kč	
3	11 281,33 Kč	3 549,76 Kč	557,35 Kč	12 501,07 Kč	7 626,85 Kč	479,82 Kč	35 996,17 Kč	
4	479,82 Kč		2 157,88 Kč				2 637,70 Kč	
5	2 469,62 Kč	1 107,93 Kč	11 739,81 Kč	2 443,61 Kč	1 478,28 Kč	4 545,12 Kč	23 784,36 Kč	
6	1 992,69 Kč	9 957,19 Kč	2 909,19 Kč	1 236,51 Kč	3 481,84 Kč		19 577,42 Kč	
7	1 716,53 Kč	2 317,41 Kč	361,03 Kč	211,54 Kč	1 418,72 Kč	3 297,18 Kč	9 322,40 Kč	
<b>Celkový součet</b>	<b>23 862,18 Kč</b>	<b>49 279,46 Kč</b>	<b>71 949,65 Kč</b>	<b>18 685,24 Kč</b>	<b>17 305,75 Kč</b>	<b>21 881,07 Kč</b>	<b>202 963,36 Kč</b>	

Obrázek 46: Kontingenční tabulka Zisk za jednotlivá období

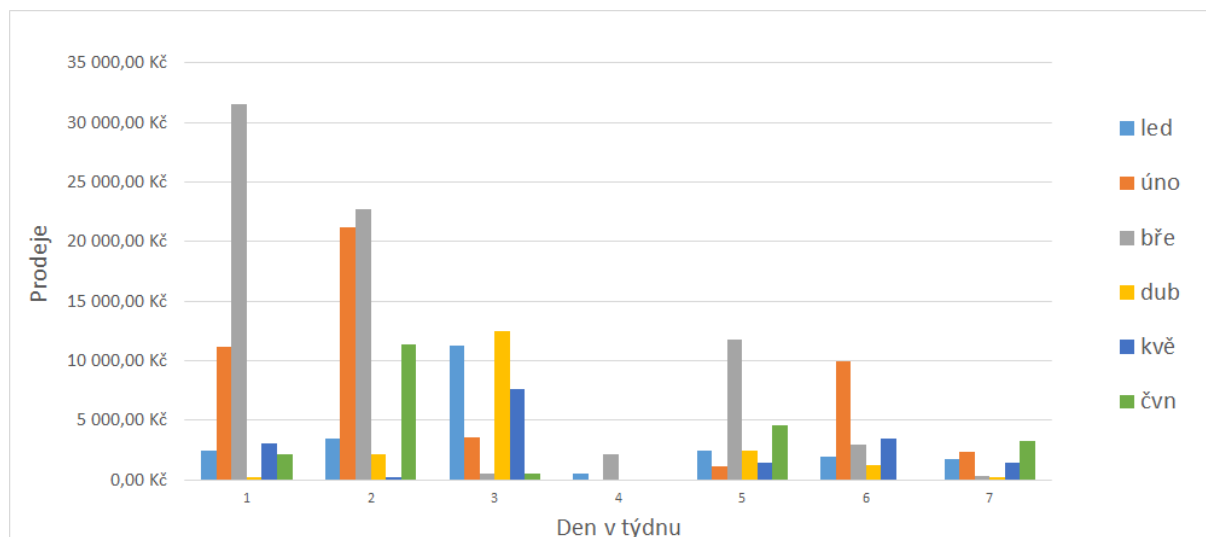
Takto zobrazená data ukazují hodnoty pro všechny měsíce a dny. Pro zobrazení konkrétních hodnot lze použít filtr hodnot. Všimněte si, že v záhlaví *Měsíce* a *Den v týdnu* se nachází rozbalovací seznam, ve kterém lze vybrat potřebné hodnoty. Po vyfiltrování hodnot se tabulka zredukuje jen na hledané hodnoty. Existuje však vícero způsobů, pomocí kterých lze data filtrovat. Jednou z nich je *Časová osa*. Pro vložení časové osy označte kont. tabulku a přejděte na kartu *Analýza kontingenční tabulky* a ve skupině *Filtr* zvolte *Vložit časovou osu*. Zobrazí se nabídka, ve které zaškrtnete pole *DatumObjednávky* a následně potvrďte tlačítkem OK. Časová osa se automaticky zobrazí do stejného sešitu jako kont. tabulka. Po vytvoření časové osy je filtr automaticky nastaven na měsíce, lze jej však změnit např. na dny nebo roky. Pro filtrování kont. tabulky pak stačí klepnout na jednu z modrých dlaždic symbolizující časové období, viz *Obrázek 47*.



Obrázek 47: Časová osa Zisk za jednotlivá období

### 8.1.2 Kontingenční graf Zisk za jednotlivá období

Poté co byla úspěšně vytvořena kontingenční tabulka s časovou osou, je vhodné připojit kontingenční graf. Pro jeho získání označte vytvořenou kont. tabulku a přejděte na kartu *Vložení*. Zde můžete vybírat ze široké škály grafů které aplikace Excel nabízí. Je vhodné se před vytvořením grafu zamyslet, který typ grafu bude pro zvolená data vhodný. Pro tento případ zvolte skupinový sloupcový graf. Po kliknutí na zvolený graf se automaticky načte do sešitu. Vytvořený graf by pak měl vypadat následovně, viz *Graf 1*. Kont. graf je provázán s kont. tabulkou a časovou osou, tudíž aplikované filtry nebo úpravy dat se propíší jak do tabulky, tak i do grafu.



Graf 1: Kontingenční graf Zisk za jednotlivá období

## 8.2 Prodeje po jednotlivých značkách

V rámci vytvořené databáze si lze všimnout, že jednotlivé položky mají různé výrobce. Kromě značky VELO, která je vlastní značkou výrobce se v seznamu položek vyskytuje i řada externích značek. Tato kapitola se blíže zaměří na vytvoření vizuálů pro jednotlivé značky výrobců. Výsledkem pak budou data o prodeji produktů jednotlivých značek za časové období. Na základě takto vytvořených vizuálů lze např. analyzovat trendy jednotlivých značek a predikovat budoucí vývoj.

### 8.2.1 Kontingenční tabulka Průměrná cena objednávky

V následující kont. tabulce budou zobrazena data obsahující informace o průměrných cenách objednávek podle jednotlivých výrobců za časové období. Vytvořte kontingenční tabulku na novém listě sešitu. Pro vložení nové kont. tabulky lze aplikovat postup z kapitoly 8.1.1, nebo přes kartu *Vložit*, rozkliknout nabídku *Kontingenční tabulka* ve skupině *Tabulky* a zvolit možnost *Z datového modelu*. Po vložení prázdné tabulky opět přejděte na lištu Pole kontingenční tabulky a z patřičných dotazů vyberte požadovaná data.

Z dotazu *Objednávky* přetáhněte hodnotu *Měsíc* do pole *Řádky* a míru *PrůCenaObj* do pole *Hodnoty*. Následně z dotazu *Položky* vyberte hodnotu *Výrobce* a přetáhněte ji do pole *Sloupce*. Tímto jsou vybrána veškerá data, která jsou potřebná a zbývá už jen upravit vzhled tabulky a názvy jednotlivých sloupců, např. viz *Obrázek 48*.

Z vytvořené kont. tabulky lze vidět jakým značkám se v uplynulých měsících dařilo a jak si vůči sobě stály. V posledním řádku jsou zobrazeny průměrné ceny za všechna období. V důsledku nízkého počtu objednávek a krátké doby prodeje, nemají tato data příliš vysokou vypovídající hodnotu. V reálných podmínkách však má tato analýza větší význam. Na základě dat o počtu objednávek a průměrné ceně košíku se dají např. predikovat výnosy a nastavit plány výroby podniku.

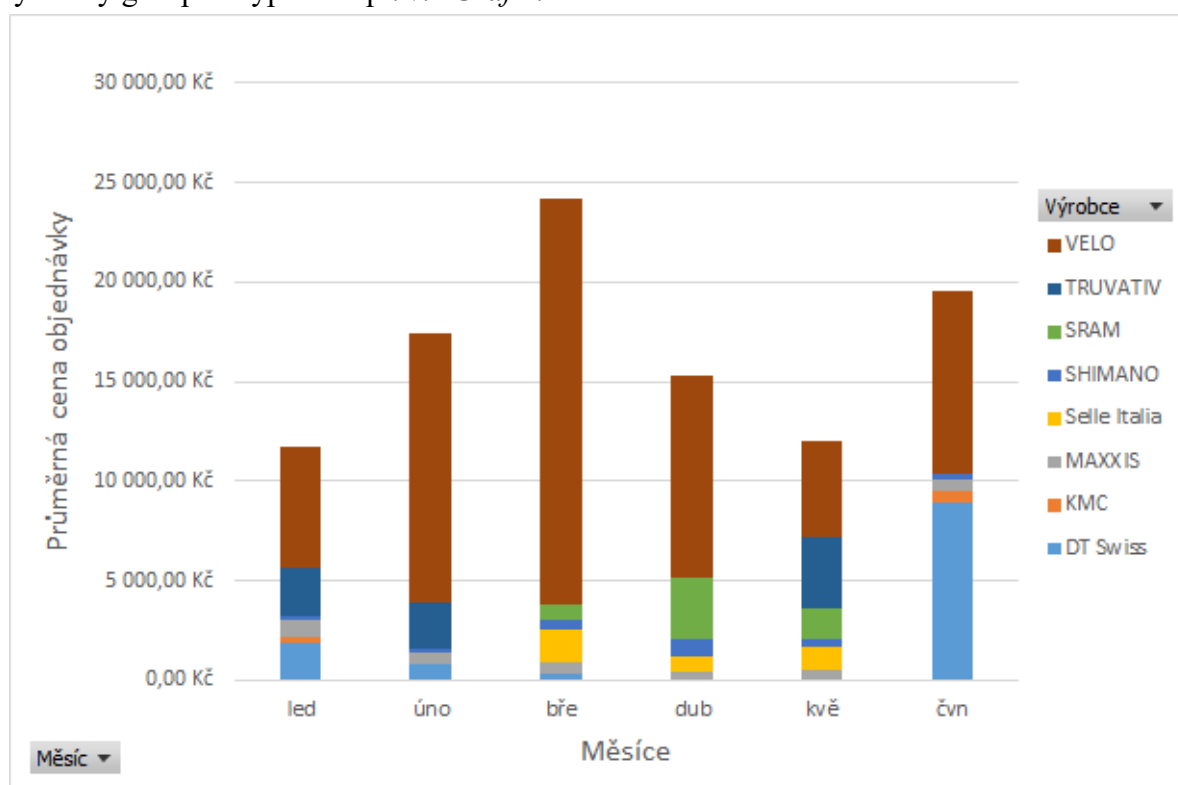
Prů Cena Obj	Výrobce							
Měsíc	DT Swiss	KMC	MAXXIS	Selle Italia	SHIMANO	SRAM	TRUVATIV	VELO
led	1 909,80 Kč	282,05 Kč	786,32 Kč		275,64 Kč		2 384,62 Kč	6 096,34 Kč
úno	763,86 Kč		576,92 Kč		205,13 Kč		2 384,62 Kč	13 513,27 Kč
bře	291,20 Kč		576,92 Kč	1 641,03 Kč	496,79 Kč	786,32 Kč		20 379,95 Kč
dub	8,97 Kč		384,62 Kč	820,51 Kč	851,28 Kč	3 076,92 Kč		10 122,81 Kč
kvě	4,11 Kč		469,23 Kč	1 230,77 Kč	324,79 Kč	1 561,54 Kč	3 576,92 Kč	4 840,18 Kč
čvn	8 889,10 Kč	564,10 Kč	630,77 Kč		333,33 Kč			9 127,58 Kč
<b>Celkový průměr</b>	<b>972,09 Kč</b>	<b>470,09 Kč</b>	<b>575,57 Kč</b>	<b>1 230,77 Kč</b>	<b>429,95 Kč</b>	<b>1 632,05 Kč</b>	<b>2 861,54 Kč</b>	<b>11 508,61 Kč</b>

Obrázek 48: Kontingenční tabulka Průměrná cena objednávky

### 8.2.2 Kontingenční graf Průměrná cena objednávky

Podobně jako u první kont. tabulky, i nyní bude následovat vytvoření kont. grafu. Pro tento případ budou data z tabulky znázorněna ve skládaném sloupcovém grafu. Pro vytvoření kont. grafu klepněte do libovolného pole kontingenční tabulky z předešlé kapitoly a na kartě *Vložit* zvolte zmíněný typ grafu. Pro výstižné zobrazení zvolených dat se však dá použít i řada jiných grafů, jako například pruhový, koláčový anebo jiný z nabídky sloupcových grafů. Vzhled grafu můžete upravit pomocí nástrojů na kartě *Návrh*.

Výsledný graf pak vypadá např. viz *Graf 2*.



Graf 2: Kontingenční graf Průměrná cena objednávky

### 8.2.3 Kontingenční tabulka Celkového prodeje a množství

Nyní je vznesen požadavek na zobrazení dat o celkovém prodeji a množství prodaných produktů pro každého výrobce. Z těchto dat lze vidět, které značky mají dominantní podíl na celkové tržbě a počtu prodaných kusů. Pro vytvoření nové kontingenční tabulky pokračujte stejným způsobem jako v předchozích kapitolách. Tabulku zobrazte v novém listu sešitu.

Následně v okně *Pole kontingenční tabulky* vyberte potřebná data pro vytvoření tabulky. Do pole *Hodnoty* vložte míry *CelkovýProdej* a *PočetKusů* z dotazu *Objednávky*. Dále z dotazu *Položky* vložte hodnotu *Výrobce* do pole *Řádky*. Do pole *Sloupce* by se pak měla automaticky přidat hodnota  $\Sigma$  *Hodnoty*. Nyní jsou definována všechna data pro kont. tabulku a zbývá jen upravit design tabulky podle potřeb uživatele. Výsledná tabulka by pak měla vypadat např. viz *Obrázek 49*.

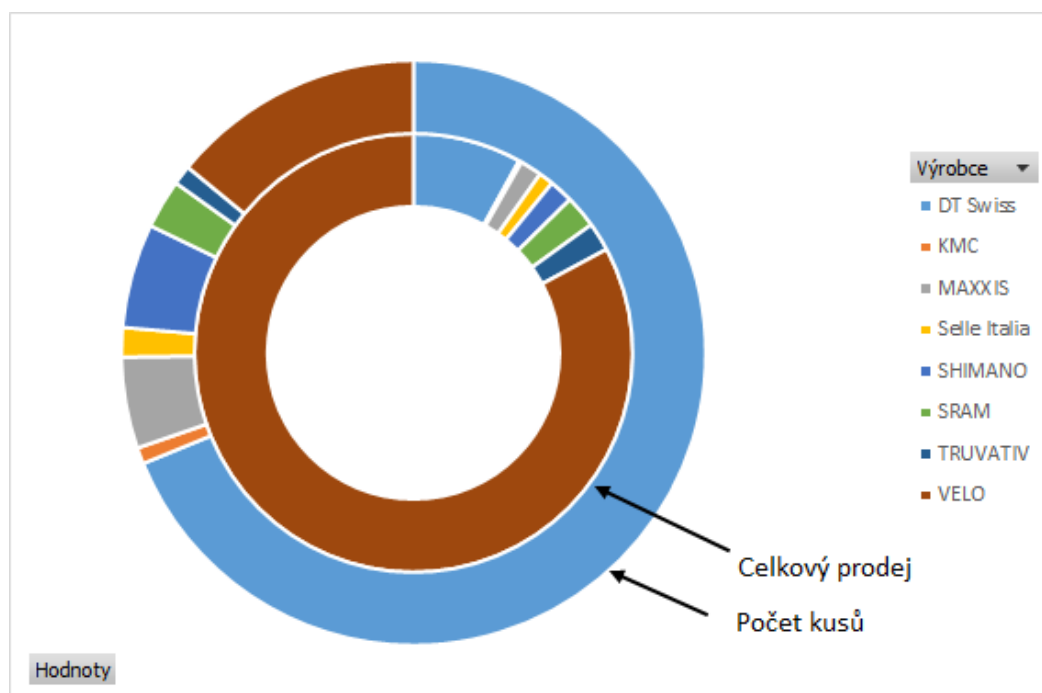
Výrobce	CelkovýProdej	Počet kusů
DT Swiss	54 437,18 Kč	382
KMC	1 410,26 Kč	5
MAXXIS	10 935,90 Kč	28
Selle Italia	7 384,62 Kč	9
SHIMANO	12 038,46 Kč	32
SRAM	16 320,51 Kč	15
TRUVATIV	14 307,69 Kč	6
VELO	563 922,08 Kč	78
<b>Celkový součet</b>	<b>680 756,69 Kč</b>	<b>555</b>

Obrázek 49: Kontingenční tabulka Celkové prodeje a množství

#### 8.2.4 Kontingenční graf Celkové prodeje a množství

K předchozí tabulce je třeba ještě vytvořit vhodný kont. graf pro úplnost demonstrace zvolených dat. Pro tento případ byl zvolen Prstencový graf, který v jednom z prstenců zobrazuje podíl tržeb každé značky v kontextu celkového prodeje a v druhém prstenci jsou zobrazeny podíly prodaného množství daných značek.

K vytvoření nového prstencového grafu opět klikněte do libovolného pole kont. Tabulky, ze které graf vychází a vložte požadovaný graf (karta *Vložení*, skupina *Grafy*). Výsledný graf s tabulkou lze případně doplnit o nástroje pro filtrování, jako je např. časová osa nebo průřez.



Graf 3: Kontingenční graf Celkové prodeje a množství



### 8.3 Přehled objednávek a tržeb za určité období

V posledním z vizuálů budou zobrazena data obsahující informace o počtu objednávek za den a kolik peněz se za ně utřžilo. K zobrazení těchto dat bude vytvořen pouze kont. graf doplněný o průřez pro jednoduší filtrování dat. Postup pro vytvoření nového samostatného kont. grafu je velmi podobný jako u vytváření kont. tabulky.

Přejděte do okna Power Pivotu a kartě *Domů* rozklikněte nabídku u pole *Kontingenční tabulka*. V nabídce je vidět celá řada možností, jako např. vložení kont. tabulky, grafu, ale i vícero grafů a tabulek najednou. Z nabídky zvolte možnost *Kontingenční graf* a následně vložte prázdný graf do nového listu sešitu. Po kliknutí do oblasti grafu se napravo zobrazí lišta *Pole kontingenčního grafu* (jedná se o obdobnou lištu jako již dříve zmíněná lišta *Pole kontingenční tabulky*). Do jednotlivých polí lišty postupně vložte potřebná data.

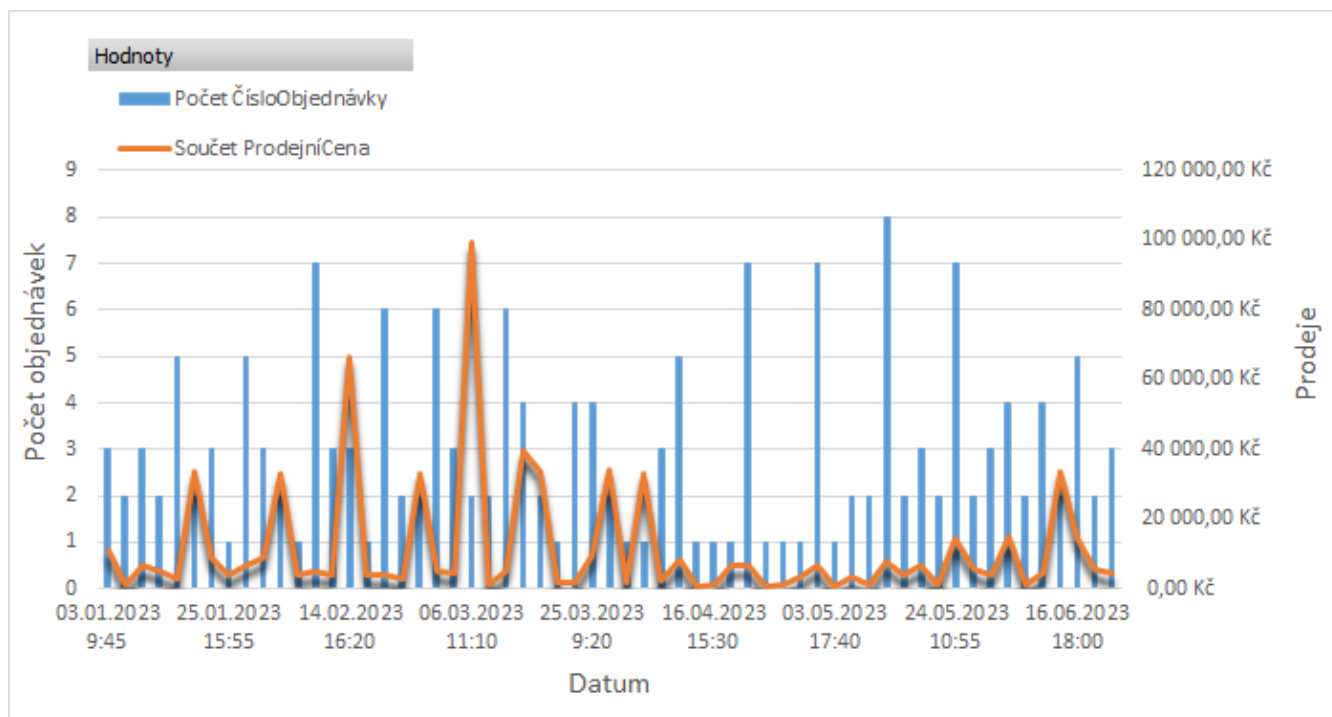
Z dotazu *Objednávky* vyberte hodnoty *ČísloObjednávky* a *ProdejníCena*, ty následně vložte do pole *Hodnoty*. Dbejte na to, aby bylo nastaveno správné zobrazení těchto hodnot. Excel většinou vkládá hodnoty automaticky jako *součet*. V případě *ČísloObjednávky* je třeba vložit hodnoty jako *Počet* (pokud by se hodnoty vložili jako součet, pak by došlo k sečtení čísel objednávky a výsledkem by byla chybná hodnota). Pro změnu typu vkládaných dat rozklikněte požadovanou hodnotu v poli a z nabídky zvolte možnost *Nastavení polí hodnot*. Zde pak už stačí vybrat typ výpočtu, který má být použit pro shrnutí dat. Hodnotu *ProdejníCena* ponechejte jako *součet*. V poslední řadě vložte hodnotu *DatumObjednávky* do pole *Osa (kategorie)*. V poli *Legenda (Řady)* se opět automaticky vloží hodnota  $\Sigma$  *Hodnoty*. Nyní jsou definována všechna potřebná data.

Typ grafu se nastavil na výchozí, pro změnu typu grafu poklepejte pravým tlačítkem do grafu a zvolte z nabídky možnost *Změnit typ grafu...* Následně zvolte typ *Vlastní Kombinovaný graf* a nastavte zobrazení grafu podle následujícího obrázku.

Název řady	Typ grafu	Vedlejší osa
 Počet ČísloObjednávky	Skupinový sloupcový	<input type="checkbox"/>
 Součet ProdejníCena	Spojnicový se značkami	<input checked="" type="checkbox"/>

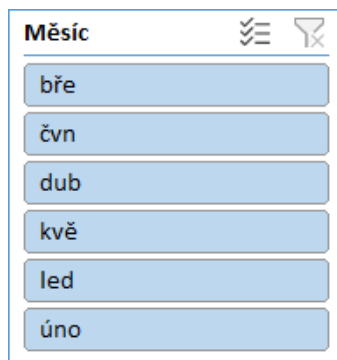
Obrázek 50: Nastavení datových řad pro kombinovaný graf

Po potvrzení výběru tlačítkem OK se zobrazí graf v požadované formě. Všimněte si, že popisky u horizontální osy jsou takřka nečitelné, poněvadž jsou zobrazeny popisky pro každý den, kdy byla zaznamenána alespoň jedna objednávka. V tomto případě je vhodné ručně nastavit interval popisků např po 7 dnech, pro lepší přehlednost. Pro úpravu popisků a os je potřeba se přepnout do lišty *Formát osy*. Toho docílíte klepnutím pravého tlačítka myši do popisků dat osy a zvolením možnosti *Formát osy...* Po zobrazení lišty se přepněte do záložky *Možnosti osy* a v oddílu *Popisky* zadejte jednotku intervalu, např na 7. Dále upravte design grafu podle vlastního uvážení, dbejte na přehlednost a kontrast barev. Výsledek je zobrazen v grafu níže (viz Graf 4). Lze si všimnout, že popisky horizontální osy uvádějí datum a čas. Pro zobrazení dat po jednotlivých dnech by bylo zapotřebí přejít zpět do okna Power Pivotu a vytvořit nový počítaný sloupec obsahující informaci pouze o dni vytvoření objednávky a tento sloupec pak následně použít v grafu namísto sloupce *ČísloObjednávky*.



**Graf 4: Kontingenční graf Přehled objednávek a tržeb za určité období**

K tomu, aby bylo možné data rychle a jednoduše filtrovat, připojte ke grafu průřez. Jedná se o další z prvků vizualizací určených k filtrování dat. Průřez, podobně jako Časová osa, se nachází ve skupině Filtry na kartě *Vložení*. Po kliknutí na ikonu průřezu se zobrazí seznam dat, podle kterých se bude tabulka filtrovat. V tomto případě zvolte jako filtr sloupec *Měsíc* z dotazu *Objednávky*. Při tvorbě průřezu je potřeba mít označenou tabulku nebo graf ke kterým se průřez přidává. Okno průřezu je pak zobrazeno na obrázku, viz *Obrázek 51*.



**Obrázek 51: Průřez Kontingenčního grafu**



## Závěr

V rámci této bakalářské práce byly představeny tři příklady, na kterých bylo demonstrováno použití nástrojů aplikace Excel pro práci s daty. Všechny příklady byly realizovány na datech z vytvořené databáze fiktivního výrobce jízdních kol. Cílem této práce bylo postupnou transformací dat získat databázi obsahující komplexní informace o jednotlivých položkách kusovníku. Data byla nejprve nahrána do prostředí Power Query Excel, ve kterém proběhlo třídění a úprava dat do vhodné formy. Následně byl využit nástroj Power Pivot, ve kterém došlo k propojení jednotlivých dotazů a vytvoření nových sloupců a měř potřebných pro další analýzu. V posledním kroku se z takto upravených dat vytvořily vizuály ve formě kontingenčních tabulek a grafů obohacené o průřezy a další prvky pro filtrování dat.

Zmíněné příklady detailně popisují postup vytvoření podnikové databáze obsahující hierarchii položek (v tomto případě kusovník jízdního kola) v prostředí Excelu. Takto vytvořená databáze představuje zjednodušený model reálného řešení. Konkrétní řešení reálných podnikových databází obsahují zpravidla větší objemy dat a jsou doplněna o další dotazy. Řešení takto rozsáhlých databází by však bylo nad rámec této práce. Pro lepší přehlednost a snazší interpretaci jednotlivých řešení, byla v této práci zvolena databáze o menším objemu dat. Při řešení rozsáhlejších databází mohou být použity výše uvedené vztahy a řešení, bez újmy na funkčnosti.

V návaznosti na vytvořené příklady by bylo možné doplnit stávající databázi o další dotazy a vizualizace. Datový model by se tak mohl rozšířit o tabulky s údaji o zaměstnancích, stavu zásob na skladě, nebo například používaných nástrojích při výrobě. Zmíněná data, ale i mnoho dalších, by následně mohla být prezentována v podobě vizuálů.

Ačkoliv pro tvorbu a úpravu podnikových databází existují na trhu aplikace přímo k tomu určené (např. Microsoft Access), tato práce potvrzuje, že tyto úlohy lze řešit i pomocí aplikace Excel. Při řešení úlohy kusovníku v rámci nástroje Power Pivot bylo zapotřebí učinit velké množství mezikroků k její realizaci. Jelikož tento nástroj k tomu není přímo určený, doporučil bych zvážit kombinaci např. právě zmiňovaného Accessu a Excelu, pro efektivnější práci. Nicméně toto zjištění mě utvrdilo v tom, že Excel je opravdu všestranný pomocník a dokáže řešit nespočet různých úloh, a proto se hodí vědět, jak s ním zacházet.

## Seznam použité literatury

- [1] DEVENS MILLER RICHARD. *Cyclopædia of Commercial and Business A. D. Appleton*. 1868.
- [2] LUHN, H P. A Business Intelligence System. *IBM Journal of Research and Development* [online]. 1958, 2(4), 314–319. Dostupné z: doi:10.1147/rd.24.0314
- [3] FOOTE D. KEITH. *A Brief History of Business Intelligence - DATAVERSITY* [online]. 2023 [vid. 2023-12-01]. Dostupné z: <https://www.dataversity.net/brief-history-business-intelligence/>
- [4] *What Is Business Intelligence | Microsoft Power BI* [online]. [vid. 2023-12-01]. Dostupné z: <https://powerbi.microsoft.com/en-au/what-is-business-intelligence/>
- [5] NOVOTNÝ, Ota, Jan POUR a David SLÁNSKÝ. *Business intelligence: jak využít bohatství ve vašich datech. Management v informační společnosti*. 2005.
- [6] STEDMAN CRAIG. *What is Business Intelligence (BI)? | Definition from TechTarget* [online]. [vid. 2023-12-01]. Dostupné z: <https://www.techtarget.com/searchbusinessanalytics/definition/business-intelligence-BI>
- [7] GÁLA LIBOR, POUR JAN a TOMAN PROKOP. *Podniková informatika*. 2006.
- [8] *What is OLAP? | IBM* [online]. [vid. 2023-12-01]. Dostupné z: <https://www.ibm.com/topics/olap>
- [9] TAN PANG-NING, STEINBACH MICHAEL a KUMAR VIPIN. *Introduction to Data Mining*. 2014.
- [10] ZUCKERMAN ARTHUR. *A Detailed Look At The History Of Business Intelligence Software | CompareCamp.com* [online]. 2015 [vid. 2023-12-01]. Dostupné z: <https://comparecamp.com/detailed-look-history-business-intelligence-software/>
- [11] *Best Business Intelligence Software in 2023 | FinancesOnline* [online]. [vid. 2023-12-01]. Dostupné z: <https://financesonline.com/c/business-intelligence-software/#history>
- [12] CLARK, Dan. *Beginning Power BI* [online]. B.m.: Apress, 2017. Dostupné z: doi:10.1007/978-1-4842-2577-6
- [13] WEBB, Chris. *Power Query for Power BI and Excel* [online]. B.m.: Apress, 2014 [vid. 2024-05-07]. Dostupné z: doi:10.1007/978-1-4302-6692-1
- [14] MASLYUK, Daniil. *Analyzing and Visualizing Data with Microsoft Power BI. Pearson Education*. 2018, 200.
- [15] *Power Pivot - Overview and Learning - Microsoft Support* [online]. [vid. 2023-12-01]. Dostupné z: <https://support.microsoft.com/en-au/office/power-pivot-overview-and-learning-f9001958-7901-4caa-ad80-028a6d2432ed>
- [16] PULS, Ken a Miguel ESCOBAR. *Master your data with Excel and Power BI : leveraging Power Query to get & transform your task flow*. nedatováno.
- [17] *DAX Functions – DAX Guide* [online]. [vid. 2024-05-18]. Dostupné z: <https://dax.guide/functions/>
- [18] DUCHOŇ, Bedřich. *Inženýrská ekonomika*. 2007.

# Přílohy

Power Query Editor: Položky – Editor Power Query

Dotazy [1]: Položky

Formule: = Table.ReplaceValue("#Přejmenované sloupce",0,null,Replacer.ReplaceValue,

123	PoložkaID	ABC	Položka	123	NadřazenáPoložka	ABC	Výrobce
1	1		Kolo			null	VELO
2	2		Přední kolo			1	VELO
3	3		Zadní kolo			1	VELO
4	4		Rám			1	VELO
5	5		Vidlice			1	DT Swiss
6	6		Sedlovka			1	VELO
7	7		Sedlo			1	Selle Italia
8	8		Řídítka			1	VELO
9	9		Představec			1	VELO
10	10		Brzda			1	SHIMANO
11	11		Přehazovačka			1	SRAM
12	12		Řazení			1	SRAM
13	13		Kazeta			1	SRAM
14	14		Převodník			1	SRAM
15	15		Řetěz			1	KMC
16	16		Kličky			1	TRUVATIV
17	17		Přední kotouč			2	SHIMANO
18	18		Zadní kotouč			3	SHIMANO
19	19		Přední třmen			2	SHIMANO

SLoupce: 8. Řádků: 30. Profílce sloupce založená na horních 1000 řádcích. NÁHLID SE STÁHL PONDĚLÍ 25. BŘEZNA 2024.

### Nastavení dotazů

**VLASTNOSTI**

Název: Položky

Všechny vlastnosti

**POUŽITÝ POSTUP**

- Zdroj
- Navigace
- Záhlaví se zvýšenou úrovní
- Nahrazená hodnota
- Změněný typ
- Nahrazená hodnota1
- Změněný typ1
- Nahrazená hodnota2
- Změněný typ2
- Rozdělit sloupec oddělovačem
- Změněný typ3
- Přejmenované sloupce
- X Nahrazená hodnota3**

Obrázek 52: Ukázka prostředí Power Query

Power Pivot pro Excel - Sešit1

Soubor Domů Návrh Rozšířené

Vložit Vložením připojit Vložením nahradit Kopírovat

Z Z datové Z jiných Existující

Aktualizovat Kontingenční

Datový typ: Formát: \$ % .00 .00

Řadit od A do Z Řadit od Z do A Zrušit řazení Zrušit všechny filtry Seřadit podle sloupců Najít Najít

AutoSum Vytvořit klíčový ukazatel výkonu

Zobrazení dat Zobrazení diagramu Zobrazit skryté Oblast výpočtu

Schránka Načíst externí data Formátování Seřadit a filtrovat Najít Výpočty Zobrazit

[File As] fx

	File As	Contact Name	ID	Company	Last Name	First Name	E-mail Address	Job Title	Business Phone	Home Phone	Mobile Phone	Fax Number	Address
1	Anderse...	Elizabeth Ander...	8	Company H	Andersen	Elizabeth		Purchasin...	(123)555-0100			(123)555-0101	123 8th St...
2	Autier ...	Catherine Autie...	18	Company R	Autier Miconi	Catherine		Purchasin...	(123)555-0100			(123)555-0101	456 18th ...
3	Axen, Th...	Thomas Axen	3	Company C	Axen	Thomas		Purchasin...	(123)555-0100			(123)555-0101	123 3rd St...
4	Bagel, Je...	Jean Philippe Ba...	17	Company Q	Bagel	Jean Philippe		Owner	(123)555-0100			(123)555-0101	456 17th ...
5	Bedecs, ...	Anna Bedecs	1	Company A	Bedecs	Anna		Owner	(123)555-0100			(123)555-0101	123 1st St...
6	Edward...	John Edwards	12	Company L	Edwards	John		Purchasin...	(123)555-0100			(123)555-0101	123 12th ...
7	Eggerer, ...	Alexander Eggerer	19	Company S	Eggerer	Alexander		Accountin...	(123)555-0100			(123)555-0101	789 19th ...
8	Entin, M...	Michael Entin	23	Company W	Entin	Michael		Purchasin...	(123)555-0100			(123)555-0101	789 23th ...
9	Goldsch...	Daniel Goldsch...	16	Company P	Goldschmidt	Daniel		Purchasin...	(123)555-0100			(123)555-0101	456 16th ...
10	Grataco...	Antonio Gratac...	2	Company B	Gratacos Sol...	Antonio		Owner	(123)555-0100			(123)555-0101	123 2nd S...
11	Grilo, Ca...	Carlos Grilo	14	Company N	Grilo	Carlos		Purchasin...	(123)555-0100			(123)555-0101	456 14th ...
12	Hasselb...	Jonas Hasselberg	24	Company X	Hasselberg	Jonas		Owner	(123)555-0100			(123)555-0101	789 24th ...
13	Krschne...	Peter Krschne	11	Company K	Krschne	Peter		Purchasin...	(123)555-0100			(123)555-0101	123 11th ...
14	Kupkova...	Helena Kupkova	15	Company O	Kupkova	Helena		Purchasin...	(123)555-0100			(123)555-0101	456 15th ...
15	Lee, Chri...	Christina Lee	4	Company D	Lee	Christina		Purchasin...	(123)555-0100			(123)555-0101	123 4th St...
16	Lee, Soo...	Soo Jung Lee	29	Company CC	Lee	Soo Jung		Purchasin...	(123)555-0100			(123)555-0101	789 29th ...
17	Li, George	George Li	20	Company T	Li	George		Purchasin...	(123)555-0100			(123)555-0101	789 20th ...
18	Liu, Run	Run Liu	26	Company Z	Liu	Run		Accountin...	(123)555-0100			(123)555-0101	789 26th ...
19	Ludick, ...	Andre Ludick	13	Company M	Ludick	Andre		Purchasin...	(123)555-0100			(123)555-0101	456 13th ...
20	Mortens...	Sven Mortensen	9	Company I	Mortensen	Sven		Purchasin...	(123)555-0100			(123)555-0101	123 9th St...

Obrázek 53: Ukázka prostředí Power Pivot