

Univerzita Jana Evangelisty Purkyně
v Ústí nad Labem

Přírodovědecká fakulta, Katedra informatiky

OLAP a ClickHouse

Laila Machová

Aplikovaná informatika

KI/ODM — OLAP a Data Mining

Letní semestr 2025

1 Volba OLAP

Pro zápočtovou práci jsem si zvolila open-source nástroj **ClickHouse**. V tomto projektu pracuji s ClickHouse pomocí Docker containeru, existuje však i cloudová služba ClickHouse Cloud.

2 Datová sada

Datovou sadu, se kterou pracuji, lze stáhnout z Kaggle ¹. Jedná se o uměle vytvořená data simulující prodeje v maloobchodu. V datové sadě jsem si upravila formáty hodnot některých sloupců (např.: Customer ID – CUST001 -> 1). Dále jsem ošetřila, aby se v datech nevyskytovaly duplicitní záznamy či chybějící hodnoty.

1. **Transaction ID:** A unique identifier for each transaction, allowing tracking and reference.
2. **Date:** The date when the transaction occurred, providing insights into sales trends over time.
3. **Customer ID:** A unique identifier for each customer, enabling customer-centric analysis.
4. **Gender:** The gender of the customer (Male/Female), offering insights into gender-based purchasing patterns.
5. **Age:** The age of the customer, facilitating segmentation and exploration of age-related influences.
6. **Product Category:** The category of the purchased product (e.g., Electronics, Clothing, Beauty), helping understand product preferences.
7. **Quantity:** The number of units of the product purchased, contributing to insights on purchase volumes.
8. **Price per Unit:** The price of one unit of the product, aiding in calculations related to total spending.
9. **Total Amount:** The total monetary value of the transaction, showcasing the financial impact of each purchase.

Obrázek 1: Informace ke sloupcům datasetu

2.1 Rozdělení do zón

Vrstvy datového skladu:

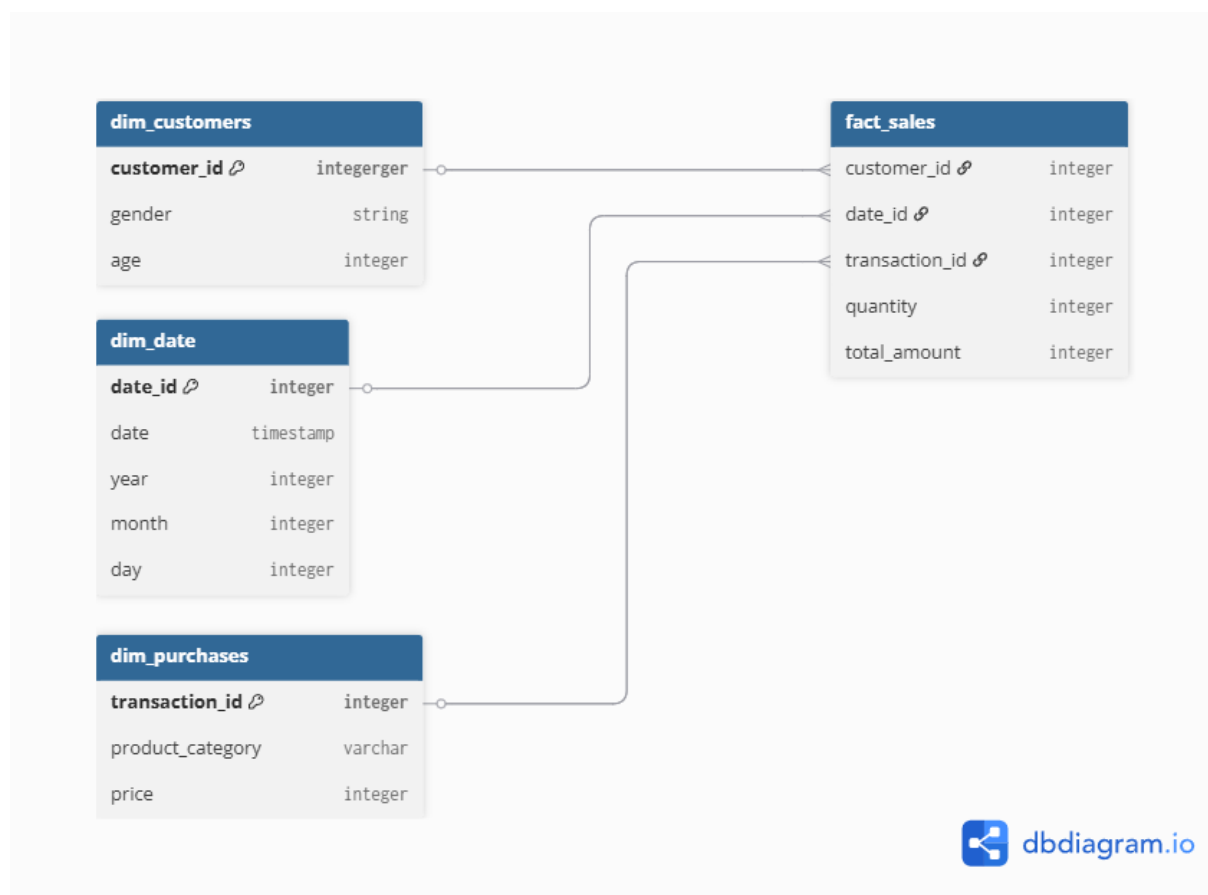
- **Raw Zone** – surová data v původní podobě, tj. stažený dataset `sales_data.csv` z Kaggle
- **Cleansed Zone** – pročištěná a rozdělená data, soubory pro import do databáze (`customers_dim.csv`, `date_dim.csv`, `purchases_dim.csv`, `fact.csv`)
- **Transformed Zone** – hvězdicová struktura (tabulka faktů + dimenze) a následné řezy datovou kostkou (analytické dotazy)

¹<https://www.kaggle.com/datasets/mohamadtalib786/retail-sales-dataset>

3 Datová kostka

Použila jsem schéma **Hvězda** se třemi dimenzemi a jednou faktovou tabulkou.

- **Dimenze Zákazníků (dim_customers)**
 - Customer ID (PK), Gender, Age
- **Dimenze Času (dim_date)**
 - Date ID (PK), Date, Year, Month, Day
- **Dimenze Objednávek (dim_purchases)**
 - Transaction ID (PK), Product Category, Price per Unit
- **Tabulka faktů (fact_sales)**
 - Customer ID (FK), Date ID (FK), Transaction ID (FK), Quantity, Total Amount



Obrázek 2: ERD diagram

3.1 Tvorba schématu Hvězda

3.1.1 Vytvoření tabulek

```
CREATE DATABASE IF NOT EXISTS dwh_sales;
USE dwh_sales;

CREATE TABLE dim_customers (
    customer_id UInt32,
    gender String,
    age UInt8,
    PRIMARY KEY (customer_id)
) ENGINE = MergeTree()
ORDER BY customer_id;

CREATE TABLE dim_date (
    date_id UInt32,
    date Date,
    day UInt8,
    month UInt8,
    year UInt16,
    PRIMARY KEY (date_id)
) ENGINE = MergeTree()
ORDER BY date_id;

CREATE TABLE dim_purchases (
    transaction_id UInt32,
    product_category String,
    price Float32,
    PRIMARY KEY (transaction_id)
) ENGINE = MergeTree()
ORDER BY transaction_id;

CREATE TABLE fact_sales (
    customer_id UInt32,
    date_id UInt32,
    transaction_id UInt32,
    quantity UInt32,
    total_amount Float32
) ENGINE = MergeTree()
ORDER BY (customer_id, date_id, transaction_id);
```

3.1.2 Import předpřipravených dat

```
#!/bin/bash

CONTAINER=clickhouse-server
DB=dwh_sales

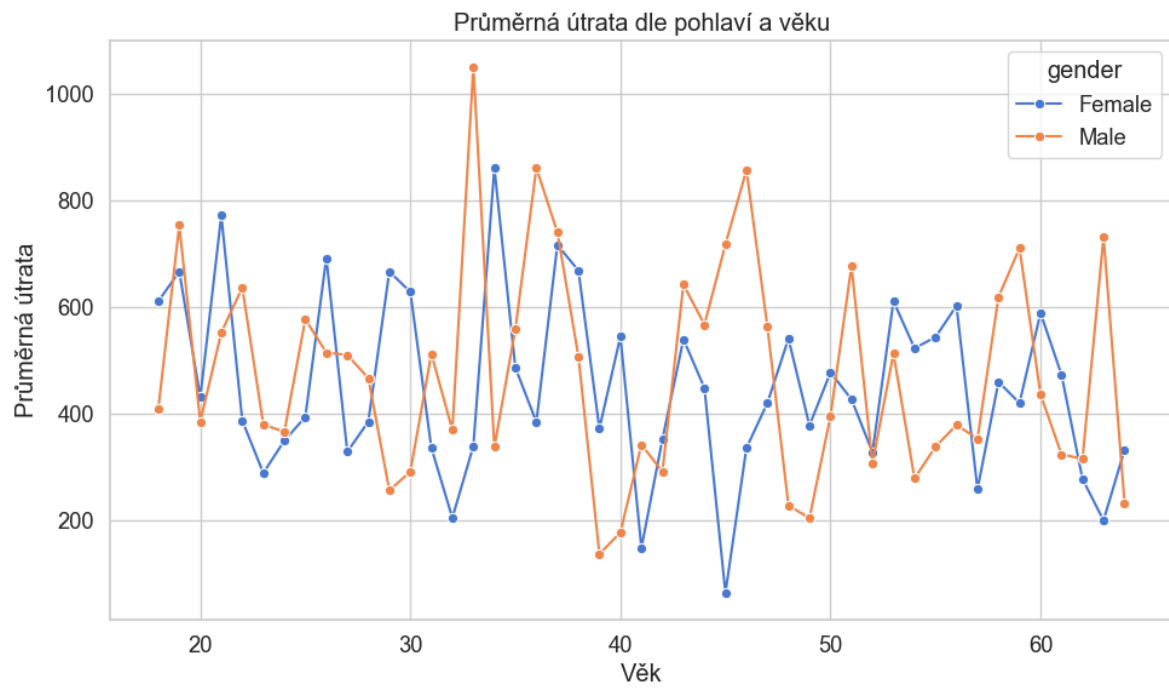
docker exec -i $CONTAINER clickhouse-client --query="INSERT INTO $DB.dim_customers
FORMAT CSVWithNames" < ./data/customers_dim.csv

docker exec -i $CONTAINER clickhouse-client --query="INSERT INTO $DB.dim_date
FORMAT CSVWithNames" < ./data/date_dim.csv

docker exec -i $CONTAINER clickhouse-client --query="INSERT INTO $DB.dim_purchases
FORMAT CSVWithNames" < ./data/purchases_dim.csv

docker exec -i $CONTAINER clickhouse-client --query="INSERT INTO $DB.fact_sales
FORMAT CSVWithNames" < ./data/fact.csv
```

[illegible]

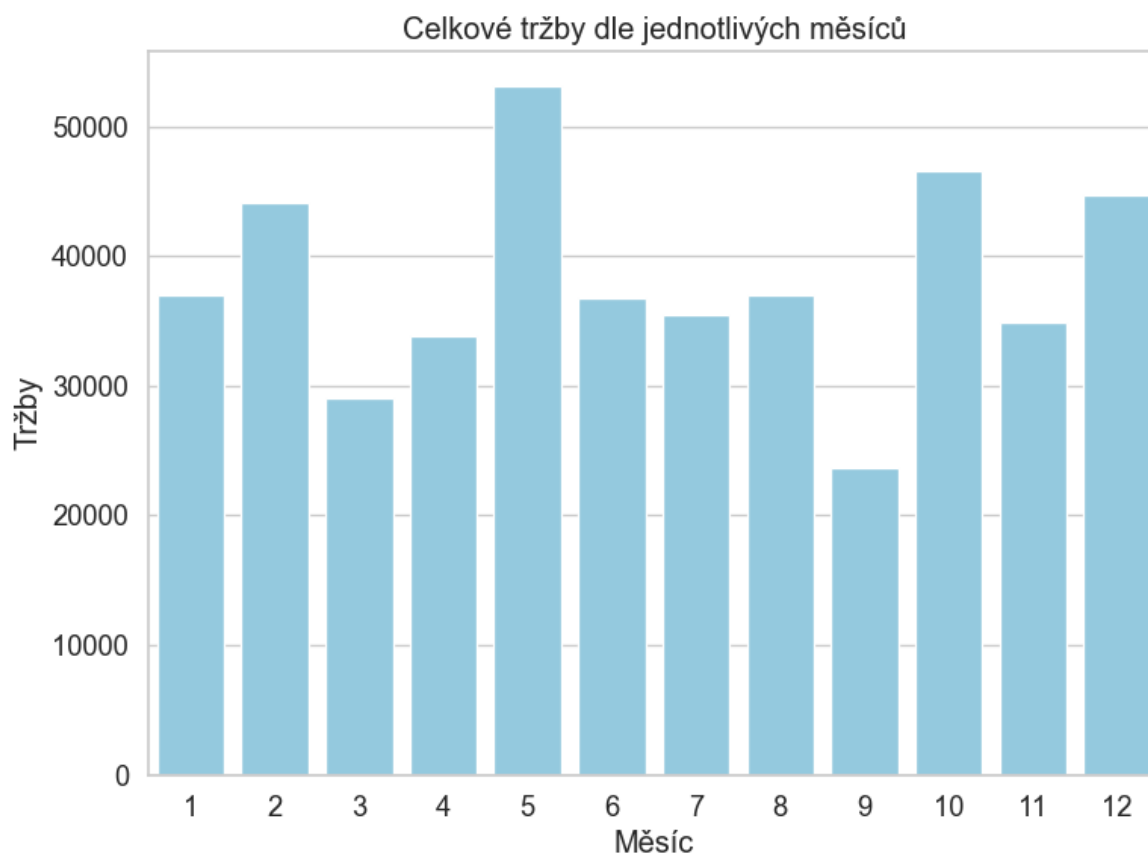


Obrázek 3: Vizualizace 1

4.2 Celkové tržby v jednotlivých měsících

```
SELECT d.month, SUM(f.total_amount) AS revenue
FROM fact_sales f
JOIN dim_date d ON f.date_id = d.date_id
GROUP BY d.month
ORDER BY month;
```

| | month | revenue |
|-----|-------|---------|
| 1. | 1 | 36980 |
| 2. | 2 | 44060 |
| 3. | 3 | 28990 |
| 4. | 4 | 33870 |
| 5. | 5 | 53150 |
| 6. | 6 | 36715 |
| 7. | 7 | 35465 |
| 8. | 8 | 36960 |
| 9. | 9 | 23620 |
| 10. | 10 | 46580 |
| 11. | 11 | 34920 |
| 12. | 12 | 44690 |

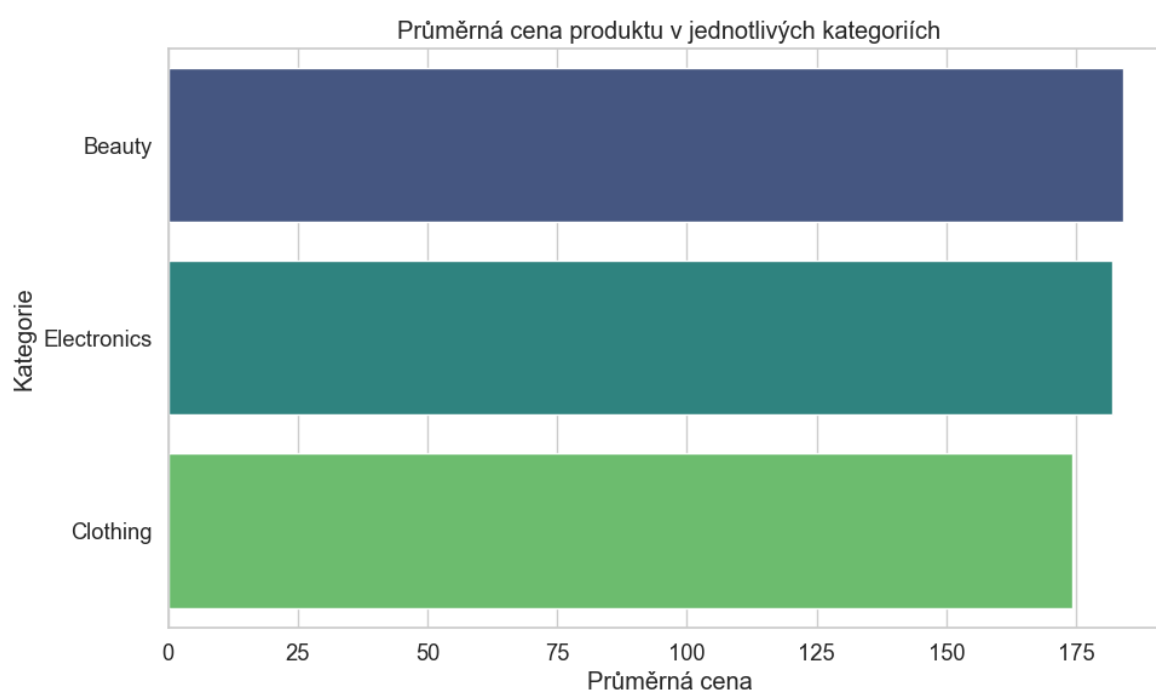


Obrázek 4: Vizualizace 2

4.3 Průměrná cena produktu v jednotlivých kategoriích

```
SELECT p.product_category, AVG(p.price) AS avg_price
FROM fact_sales f
JOIN dim_purchases p ON f.transaction_id = p.transaction_id
GROUP BY p.product_category
ORDER BY avg_price DESC;
```

| | product_category | avg_price |
|----|------------------|--------------------|
| 1. | Beauty | 184.05537459283389 |
| 2. | Electronics | 181.90058479532163 |
| 3. | Clothing | 174.28774928774928 |

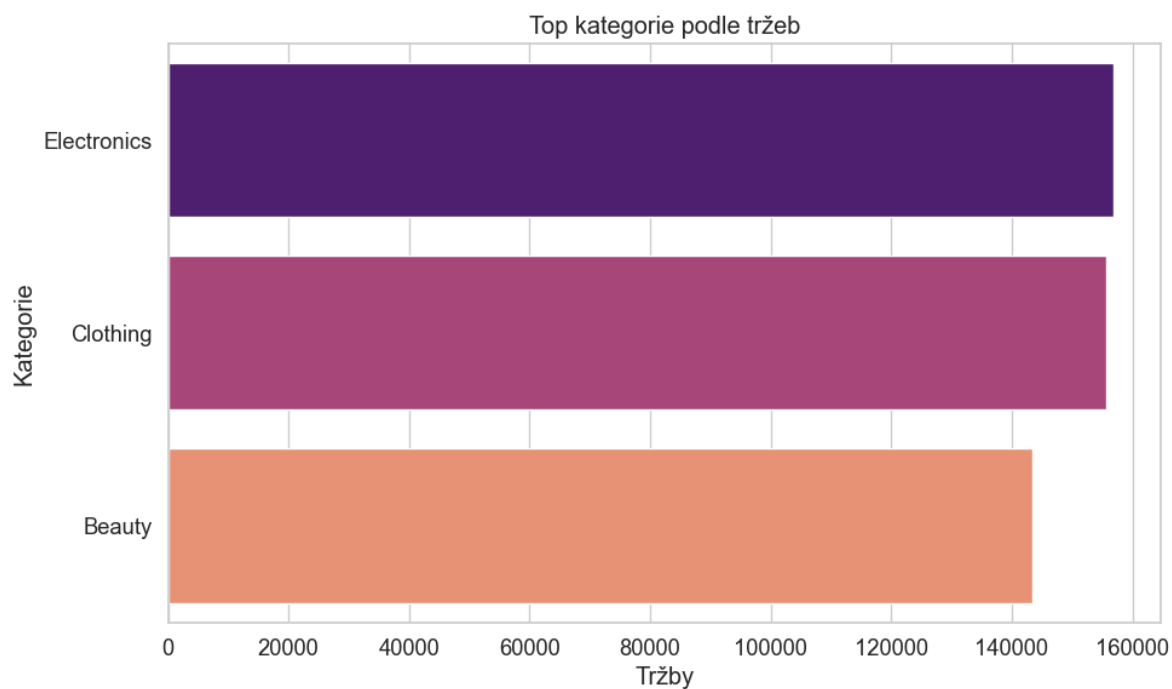


Obrázek 5: Vizualizace 3

4.4 Top kategorie dle tržeb

```
SELECT p.product_category, SUM(f.total_amount) AS revenue
FROM fact_sales f
JOIN dim_purchases p ON f.transaction_id = p.transaction_id
GROUP BY p.product_category
ORDER BY revenue DESC;
```

| | product_category | revenue |
|----|------------------|---------|
| 1. | Electronics | 156905 |
| 2. | Clothing | 155580 |
| 3. | Beauty | 143515 |

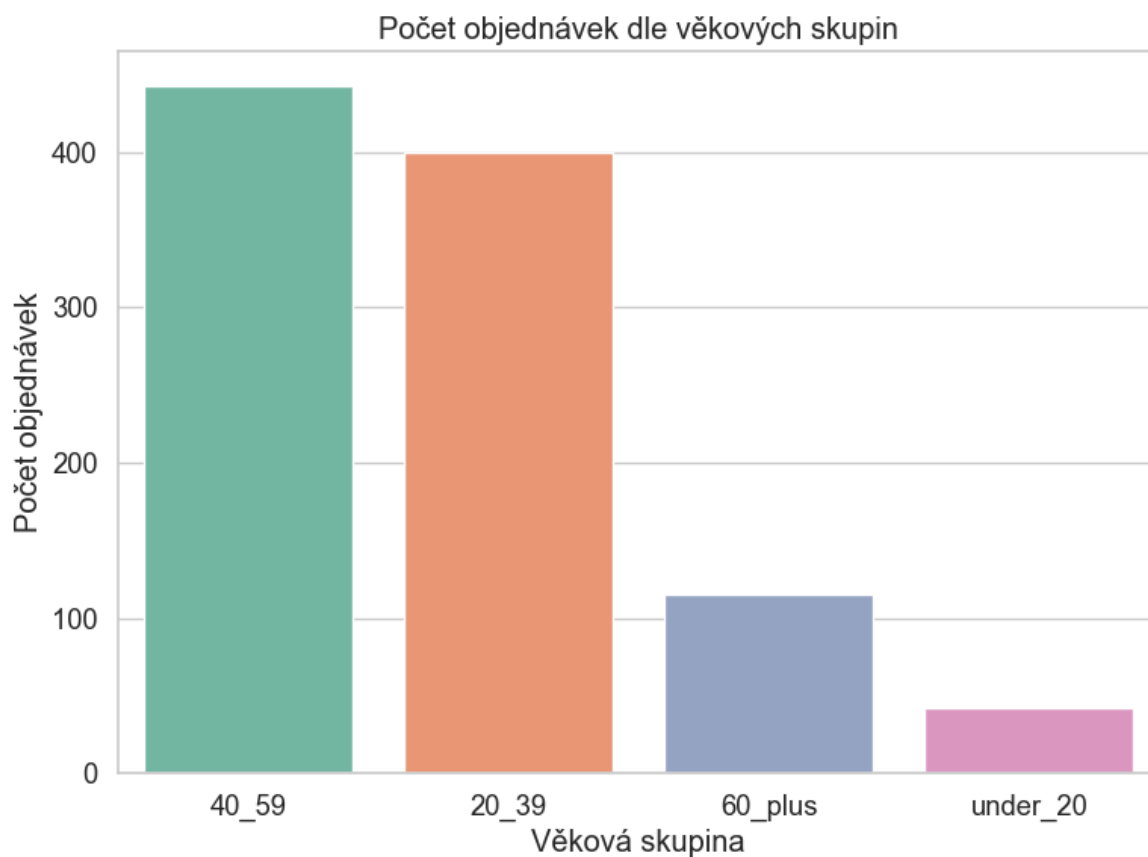


Obrázek 6: Vizualizace 4

4.5 Počet objednávek dle věkových skupin

```
SELECT
CASE
  WHEN c.age < 20 THEN 'under_20'
  WHEN c.age BETWEEN 20 AND 39 THEN '20_39'
  WHEN c.age BETWEEN 40 AND 59 THEN '40_59'
  ELSE '60_plus'
END AS age_group,
COUNT(*) AS orders
FROM fact_sales f
JOIN dim_customers c ON f.customer_id = c.customer_id
GROUP BY age_group
ORDER BY orders DESC;
```

| | age_group | orders |
|----|-----------|--------|
| 1. | 40_59 | 443 |
| 2. | 20_39 | 400 |
| 3. | 60_plus | 115 |
| 4. | under_20 | 42 |



Obrázek 7: Vizualizace 5

5 Data Mining metoda – Shlukování (Clustering)

Jelikož ClickHouse nepodporuje pokročilé metody data miningu ani strojové učení, zvolila jsem postup, kdy se výstupy SQL dotazů uloží do souborů ve formátu CSV. Tyto soubory jsou následně zpracovány v jazyce Python pomocí algoritmu K-Means z knihovny scikit-learn.

Průměrná útrata zákazníků dle věku

```
SELECT c.age, AVG(f.total_amount) AS avg_spent
FROM fact_sales f
JOIN dim_customers c ON f.customer_id = c.customer_id
GROUP BY c.age
ORDER BY age
INTO OUTFILE 'sql_dm.csv' FORMAT CSV;
```

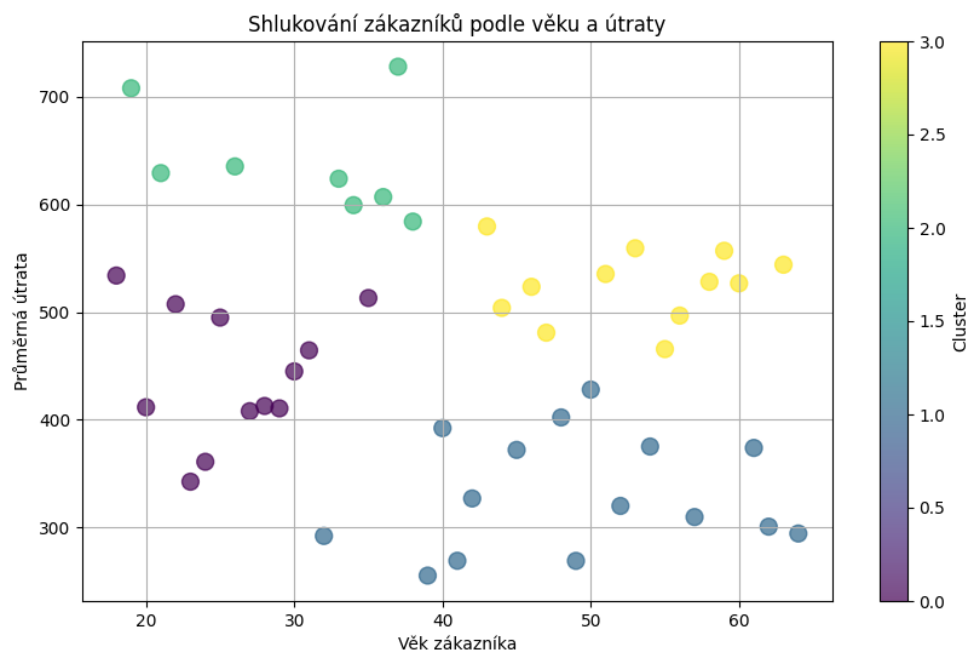
```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler

df = pd.read_csv("sql_dm.csv")

# Standardizace
scaler = StandardScaler()
X_scaled = scaler.fit_transform(df)

# Shlukování (KMeans)
kmeans = KMeans(n_clusters=4, random_state=42)
df["cluster"] = kmeans.fit_predict(X_scaled)
print(df.groupby("cluster").mean())
```

| | age | avg_spent |
|---------|-----------|------------|
| cluster | | |
| 0 | 26.000000 | 442.156216 |
| 1 | 49.066667 | 332.049800 |
| 2 | 30.500000 | 639.449949 |
| 3 | 52.916667 | 525.159671 |



Obrázek 8: Vizualizace Shlukování

Zdroje

- [1] YANDEX. *ClickHouse Documentation* [online]. 2025 [cit. 2025-06-08]. Dostupné z: <https://clickhouse.com/docs/en/>