### 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 ClikHouse 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 – CUSTOO1 -> 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:

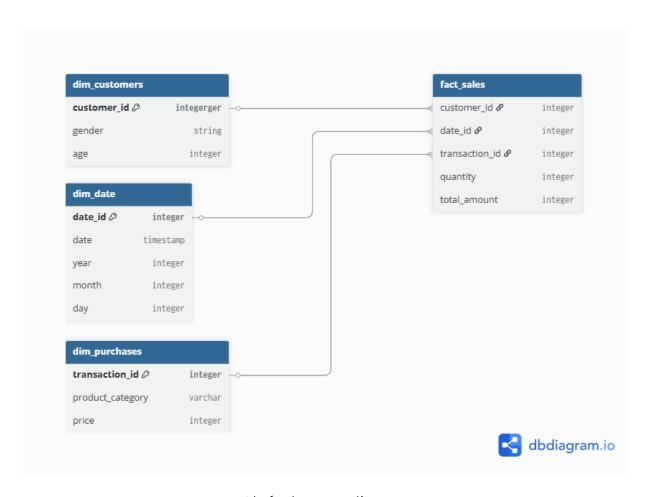
- **Raw Zone** surová data v původní podobě, tj. stažený dataset sales\_data.csv z Kaggle
- **Cleansed Zone** pročištěná a rozdělená data, soubory pro import do databáze (customers\_dim.csv, date\_dim.csv, purchases\_dim.csv, fact.csv)
- Transformed Zone hvězdicová struktura (tabulka faktů + dimenze) a následné řezy datovou kostkou (analytické dotazy)

<sup>&</sup>lt;sup>1</sup>https://www.kaggle.com/datasets/mohammadtalib786/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

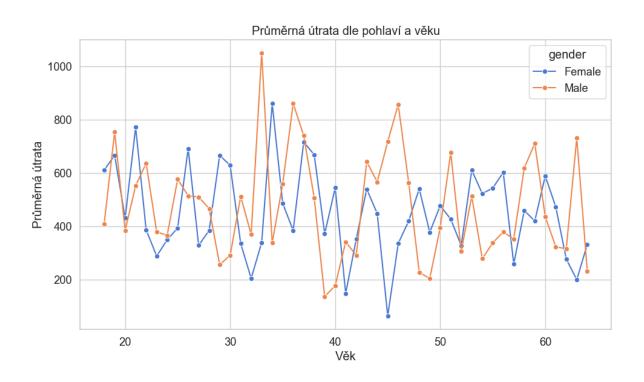
docker exec -i $CONTAINER clickhouse-client --query="INSERT_INTO_$DB.fact_sales_
FORMAT_CSVWithNames" < ./data/fact.csv
```

## **4** Řezy datovou kostkou – Dotazy nad daty

## 4.1 Průměrná útrata dle pohlaví a věku

```
SELECT c.age, c.gender, 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, c.gender
ORDER BY age DESC;
```

1.	<b>-age</b> 64	<b>gender</b> Female	avg_spent = 332.89473684210526	34.	48	Male	228.75	67.	31	Female	336,666666666667
2.	64 64	Male	233.333333333333334	35.	46 47	Female	421	68.	31	Male	512.5
3.	63	Male	731.363636363636364	36.	47	Male	562.7272727272727	69.	30	Female	628.5
4.	63	Female	200.833333333333334	30. 37.	46	Male	856.6666666666666	70.	30	Male	292.0833333333333
5.	62	Male	316.25	38.	46	Female	336.25	71.	29	Male	257
6.	62	Female	278.1818181818182	39.	45	Male	717.5	72.	29	Female	666,66666666666
7.	61	Male	324.16666666666667	40.	45	Female	65	73.	28	Female	385.7142857142857
8.	61	Female	473.33333333333333	41.	44	Female	448.75	74.	28	Male	467.14285714285717
9.	60	Male	436.6666666666667	42.	44	Male	567.1428571428571	75.	27	Male	510.5
10.	60	Female	589.2307692307693	43.	43	Female	540	76.	27	Female	329.2307692307692
11.	59	Male	710.625	44.	43	Male	642.5	77.	26	Female	691.666666666666
12.	59	Female	420.5555555555554	45.	42	Male	291.8181818181818	78.	26	Male	515
13.	58	Female	460	46.	42	Female	352.6666666666667	79.	25	Female	394.444444444444
14.	58	Male	619.166666666666	47.	41	Female	149.375	80.	25	Male	577.2727272727273
15.	57	Male	353.75	48.	41	Male	342.6923076923077	81.	24	Male	366.5
16.	57	Female	259.2857142857143	49.	40	Male	178.5	82.	24	Female	350
17.	56	Female	602.5	50.	40	Female	545	83.	<b>2</b> 3	Male	380.35714285714283
18.	56	Male	379.4444444444446	51.	39	Female	372.77777777777777	84.	23	Female	289.5
19.	55	Female	543.8461538461538	52.	39	Male	137.77777777777777	85.	22	Female	387.5
20.	55	Male	338.75	53.	38	Female	668.88888888888	86.	22	Male	636.5384615384615
21.	54	Male	279.4117647058824	54.	38	Male	508	87.	21	Female	771.4285714285714
22.	54	Female	523.1818181818181	55.	37	Female	716.25	88.	21	Male	552.6923076923077
23.	53	Female	611.25	56.	37	Male	740	89.	20	Female	431.25
24.	53	Male	513.33333333333334	57.	36	Female	385	90.	20	Male	385.555555555554
25.	52	Male	307.7777777777777	58.	36	Male	860.7142857142857	91.	19	Female	666.8181818181819
26.	52	Female	328.46153846153845	59.	35	Male	559.375	92.	19	Male	753.5
27.	51	Male	676.5384615384615	60.	35	Female	486.7857142857143	93.	18	Female	610.7692307692307
28.	51	Female	427.6470588235294	61.	34	Male	338.2142857142857	94.	18	Male	409.375
29.	50	Female	477.7777777777777	62.	34	Female	860.7142857142857		-age	gender_	avg_spent_
30.	50	Male	396.07142857142856	63.	33	Male	1050				
31.	49	Male	205	64.	33	Female	340				
32.	49	Female	378 <b>.</b> 57142857142856	65.	32	Male	370				
33.	48	Female	541	66.	32	Female	205.5555555555554				

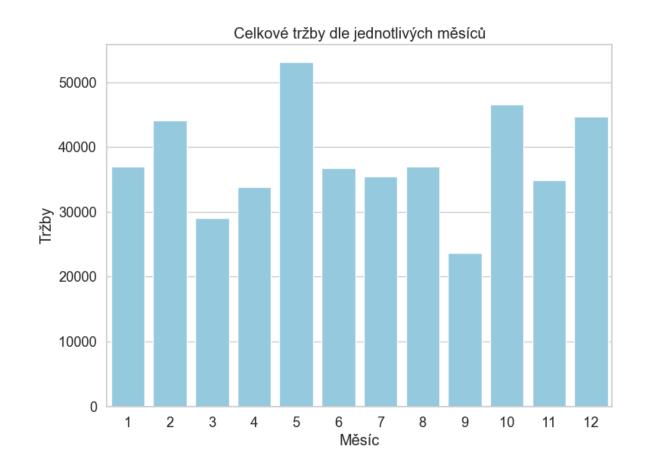


Obrázek 3: Vizualizace 1

## 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	3 <u>6</u> 980
2.	2	4 <u>4</u> 060
3.	3	2 <u>8</u> 990
4.	4	3 <u>3</u> 870
5.	5	5 <u>3</u> 150
6.	6	3 <u>6</u> 715
7.	7	3 <u>5</u> 465
8.	8	3 <u>6</u> 960
9.	9	2 <u>3</u> 620
10.	10	4 <u>6</u> 580
11.	11	3 <u>4</u> 920
12.	12	4 <u>4</u> 690



Obrázek 4: Vizualizace 2

### 4.3 Průměrná cena produktu v jednotlivých kategoriích

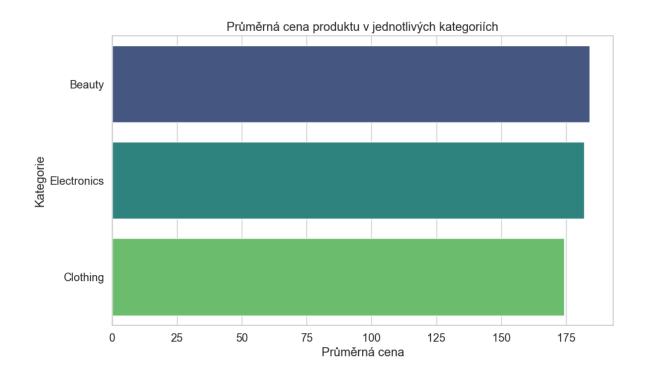
```
SELECT p.product_category, AVG(p.price) AS avg_price
FROM fact_sales f
JOIN dim_purchases p ON f.transaction_id = p.transaction_id
GROUP BY p.product_category
ORDER BY avg_price DESC;
```

```
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

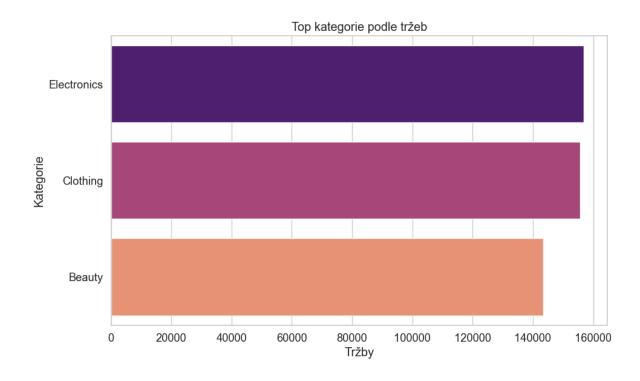
```
SELECT p.product_category, SUM(f.total_amount) AS revenue
FROM fact_sales f
JOIN dim_purchases p ON f.transaction_id = p.transaction_id
GROUP BY p.product_category
ORDER BY revenue DESC;
```

```
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,

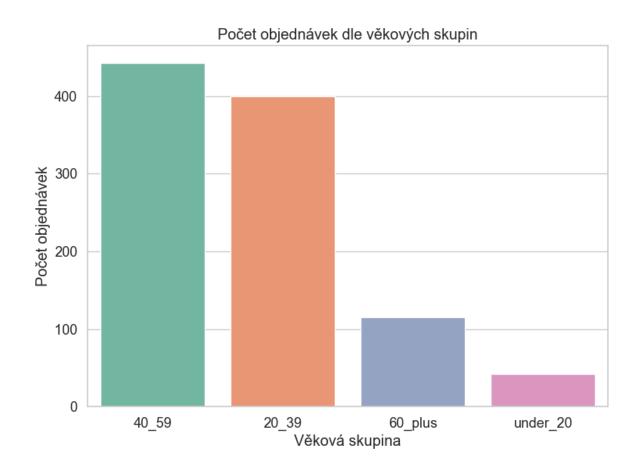
COUNT(*) AS orders

FROM fact_sales f

JOIN dim_customers c ON f.customer_id = c.customer_id

GROUP BY age_group

ORDER BY orders DESC;
```



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;
```

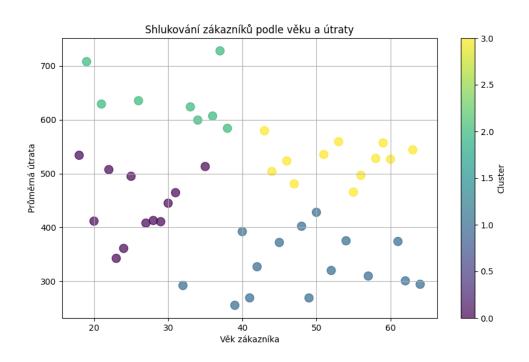
```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler

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

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

# Shlukovani (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 4442.156216
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/