Univerzitet u Sarajevu
Elektrotehnički fakultet u Sarajevu
Stručni studij Razvoj softvera
Skladišta podataka, akademska 2024/2025
Predmetni profesor: R. prof. dr Samir Omanović dipl. ing. el.

Projekat skladišta podataka

Radili: Mahmutović Eldin 211-ST, Petković Marko 232-ST,
Pap Anand 238-ST, Skopljak Malik 181-ST

Uvod

Za izradu našeg projekta koristili smo:

1. SQLite DBMS (Database management system = sistem za upravljanje bazom podataka)
2. Već pripremljenu produkcijsku bazu podataka napunjenu početnim podacima, a sa kojom smo se bili upoznali kroz rad na predmetu UBP (Uvod u baze podataka, 1. semestar RS studija)
3. DB Browser aplikaciju za inspekciju stanja baze podataka i izvršavanje upita

Produkcijska baza se nalazi u trećoj normalnoj formi (3NF), te će biti potrebna denormalizacija prilikom prenosa podataka u DWH (Data warehouse = skladište podataka). U svrhu izvještavanja i analize, podaci se denormalizuju kako bi se unaprijedile performanse upita. Denormalizaciju ćemo obaviti spajanjima (join-ovima) odgovarajućih tabela.

Naš DWH smo osmislili tako da se sastoji od sljedećih dimenzionih modela zvjezdane strukture:

1. Model narudžbe i odgovarajuća mjera
2. Model inventure i odgovarajuća mjera

Za mjeru narudžbe vežemo četiri dimenzije:

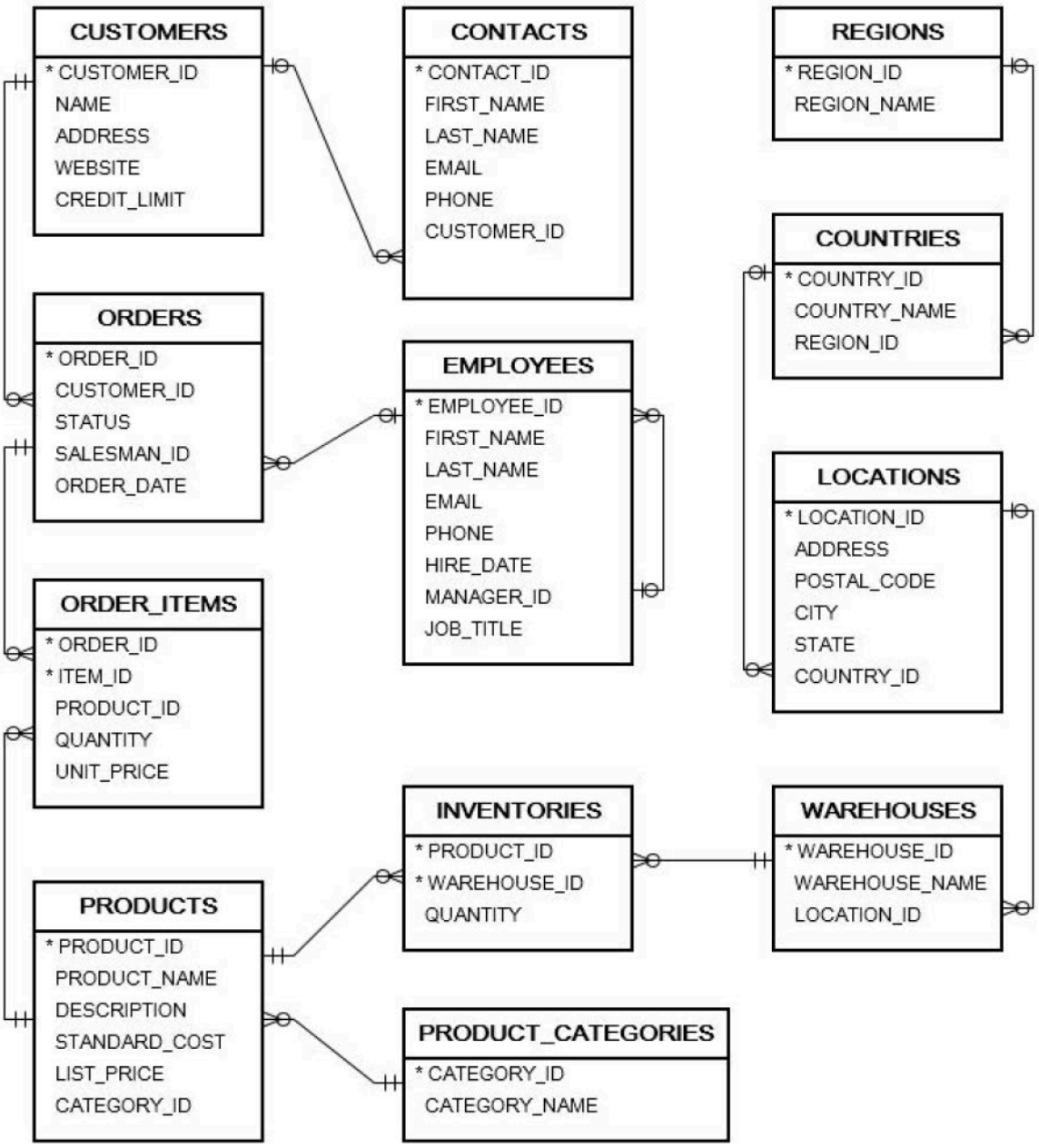
1. Uposlenik
2. Kupac
3. Proizvod
4. Vrijeme

A za mjeru inventura vežemo sljedeće dvije dimenzije:

1. Proizvod
2. Skladište

Dijagrami strukture produkcijske baze i DWH

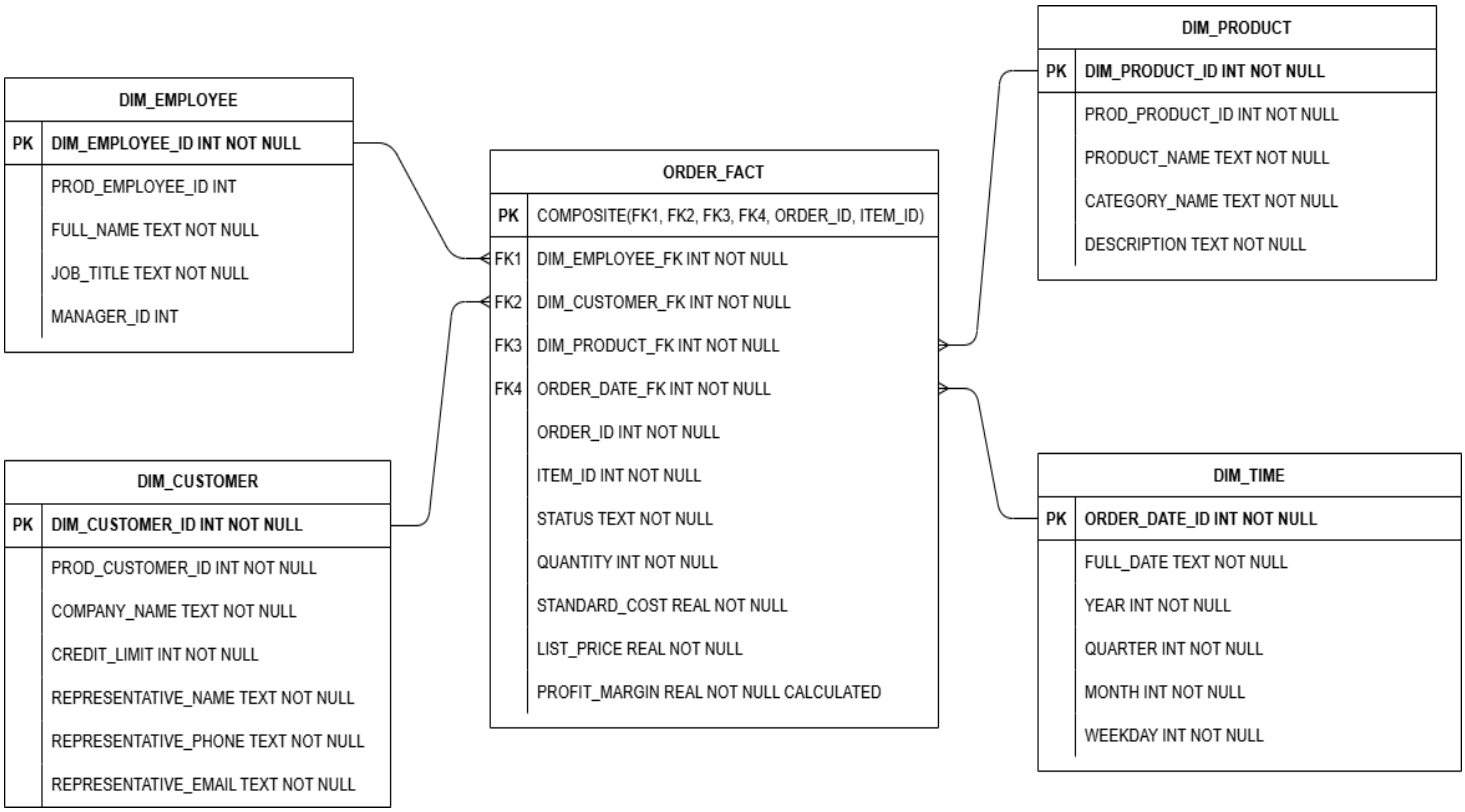
ERD (Entity-relationship dijagram) produkcijske baze



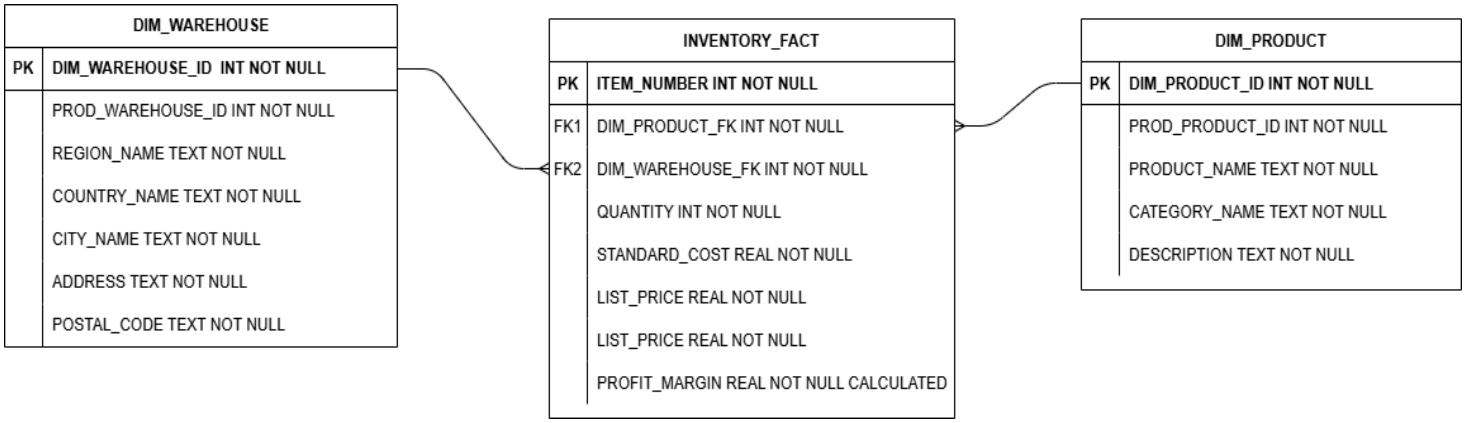
Matrica arhitekture DWH

Poslovni proces/događaj	Vrijeme	Klijent/kupac	Proizvod	Zaposlenik prodavač	Skladište
Narudžba	X	X	X	X	
Inventura			X		X

ERD ORDER_FACT tabele



ERD INVENTORY_FACT tabele



Kreiranje DWH

Na početku ćemo spojiti produkcijsku bazu na naš novoformirani DWH, te postaviti obavezu poštovanja ograničenja stranih ključeva.

```
ATTACH DATABASE 'production.db' AS prod;

PRAGMA foreign_keys = ON;
```

Narednim komandama kreiramo redom dimenzije koje ćemo kasnije vezati na naše fact tabele.

```

CREATE TABLE DIM_EMPLOYEE (
    DIM_EMPLOYEE_ID INTEGER PRIMARY KEY AUTOINCREMENT,
    PROD_EMPLOYEE_ID INTEGER,
    FULL_NAME TEXT NOT NULL,
    JOB_TITLE TEXT NOT NULL,
    MANAGER_ID INTEGER
);

CREATE TABLE DIM_CUSTOMER (
    DIM_CUSTOMER_ID INTEGER PRIMARY KEY AUTOINCREMENT,
    PROD_CUSTOMER_ID INTEGER NOT NULL,
    COMPANY_NAME TEXT NOT NULL,
    CREDIT_LIMIT INTEGER NOT NULL,
    REPRESENTATIVE_NAME TEXT NOT NULL,
    REPRESENTATIVE_PHONE TEXT NOT NULL,
    REPRESENTATIVE_EMAIL TEXT NOT NULL
);

CREATE TABLE DIM_PRODUCT (
    DIM_PRODUCT_ID INTEGER PRIMARY KEY AUTOINCREMENT,
    PROD_PRODUCT_ID INTEGER NOT NULL,
    PRODUCT_NAME TEXT NOT NULL,
    CATEGORY_NAME TEXT NOT NULL,
    DESCRIPTION TEXT NOT NULL
);

CREATE TABLE DIM_WAREHOUSE (
    DIM_WAREHOUSE_ID INTEGER PRIMARY KEY AUTOINCREMENT,
    PROD_WAREHOUSE_ID INTEGER NOT NULL,
    REGION_NAME TEXT NOT NULL,
    COUNTRY_NAME TEXT NOT NULL,
    CITY_NAME TEXT NOT NULL,
    ADDRESS TEXT NOT NULL,
    POSTAL_CODE TEXT NOT NULL
);

CREATE TABLE DIM_TIME (
    ORDER_DATE_ID INTEGER PRIMARY KEY,
    FULL_DATE TEXT NOT NULL,
    YEAR INTEGER NOT NULL,
    QUARTER INTEGER NOT NULL,
    MONTH INTEGER NOT NULL,
    WEEKDAY INTEGER NOT NULL
);
```

Punjenje tabela dimenzija

```sql

INSERT INTO DIM_EMPLOYEE (PROD_EMPLOYEE_ID, FULL_NAME, JOB_TITLE, MANAGER_ID)
SELECT EMPLOYEE_ID, FIRST_NAME || ' ' || LAST_NAME, JOB_TITLE, MANAGER_ID FROM prod.EMPLOYEES;

INSERT INTO DIM_EMPLOYEE (DIM_EMPLOYEE_ID, PROD_EMPLOYEE_ID, FULL_NAME, JOB_TITLE, MANAGER_ID)
VALUES (0, NULL, 'Unknown', 'Unknown', NULL);

INSERT INTO DIM_CUSTOMER (PROD_CUSTOMER_ID, COMPANY_NAME, CREDIT_LIMIT, REPRESENTATIVE_NAME, REPRESENTATIVE_PHONE, REPRESENTATIVE_EMAIL)
SELECT c.CUSTOMER_ID, c.NAME, c.CREDIT_LIMIT, con.FIRST_NAME || ' ' || con.LAST_NAME, con.PHONE, con.EMAIL
FROM prod.CUSTOMERS c
JOIN prod.CONTACTS con ON c.CUSTOMER_ID = con.CUSTOMER_ID;

INSERT INTO DIM_PRODUCT (PROD_PRODUCT_ID, PRODUCT_NAME, CATEGORY_NAME, DESCRIPTION)
SELECT p.PRODUCT_ID, p.PRODUCT_NAME, pc.CATEGORY_NAME, p.DESCRIPTION
FROM prod.PRODUCTS p
JOIN prod.PRODUCT_CATEGORIES pc ON p.CATEGORY_ID = pc.CATEGORY_ID;

INSERT INTO DIM_WAREHOUSE (PROD_WAREHOUSE_ID, REGION_NAME, COUNTRY_NAME, CITY_NAME, ADDRESS, POSTAL_CODE)
SELECT DISTINCT
    w.WAREHOUSE_ID,
    r.REGION_NAME,
    c.COUNTRY_NAME,
    w.WAREHOUSE_NAME,
    l.ADDRESS,
    l.POSTAL_CODE
FROM prod.WAREHOUSES w
JOIN prod.LOCATIONS l ON w.LOCATION_ID = l.LOCATION_ID
JOIN prod.COUNTRIES c ON l.COUNTRY_ID = c.COUNTRY_ID
JOIN prod.REGIONS r ON c.REGION_ID = r.REGION_ID;

INSERT INTO DIM_TIME (ORDER_DATE_ID, FULL_DATE, YEAR, QUARTER, MONTH, WEEKDAY)
SELECT DISTINCT CAST(strftime('%Y%m%d', ORDER_DATE) AS INTEGER) AS ORDER_DATE_ID, ORDER_DATE, strftime('%Y', ORDER_DATE),
       (strftime('%m', ORDER_DATE) - 1) / 3 + 1, strftime('%m', ORDER_DATE),
       (CASE
            WHEN strftime('%w', ORDER_DATE) = '0' THEN 7
            ELSE strftime('%w', ORDER_DATE)
        END) AS WEEKDAY
FROM prod.ORDERS;
```

Obratite pažnju da smo u **DIM_EMPLOYEE** dimenziju unijeli zaseban red koji označava uposlenika koji nije zabilježen tokom dogovaranja narudžbe (njegov ID je vrijednosti NULL). Da nismo tako postupili, prilikom unosa podataka u **ORDER_FACT** tabelu, narudžbe koje su dogovorene od strane prodavača koji nisu zabilježeni bi bile jednostavno odstranjene i ne bi bile uzete u razmatranje za analize koje slijede nakon kreiranja DWH, a time ne bi dobili pravu sliku stanja.

Dodatna napomena: kupac je kompanija, a ne pojedinac i sa njom naša firma komunicira preko kontakt osobe koja je predstavnik te kompanije sa kojom poslujemo.

Kreiranje i punjenje ORDER_FACT tabele mjere

```
87
88  CREATE TABLE ORDER_FACT (
89      DIM_EMPLOYEE_FK INTEGER NOT NULL,
90      DIM_CUSTOMER_FK INTEGER NOT NULL,
91      DIM_PRODUCT_FK INTEGER NOT NULL,
92      ORDER_DATE_FK INTEGER NOT NULL,
93      ORDER_ID INTEGER NOT NULL,
94      ITEM_ID INTEGER NOT NULL,
95      STATUS TEXT NOT NULL,
96      QUANTITY INTEGER NOT NULL,
97      STANDARD_COST REAL NOT NULL,
98      LIST_PRICE REAL NOT NULL,
99      PROFIT_MARGIN REAL NOT NULL GENERATED ALWAYS AS (ROUND(LIST_PRICE - STANDARD_COST, 2)) STORED,
100     PRIMARY KEY (DIM_EMPLOYEE_FK, DIM_CUSTOMER_FK, DIM_PRODUCT_FK, ORDER_DATE_FK, ORDER_ID, ITEM_ID),
101     FOREIGN KEY (DIM_EMPLOYEE_FK) REFERENCES DIM_EMPLOYEE(DIM_EMPLOYEE_ID),
102     FOREIGN KEY (DIM_CUSTOMER_FK) REFERENCES DIM_CUSTOMER(DIM_CUSTOMER_ID),
103     FOREIGN KEY (DIM_PRODUCT_FK) REFERENCES DIM_PRODUCT(DIM_PRODUCT_ID),
104     FOREIGN KEY (ORDER_DATE_FK) REFERENCES DIM_TIME(ORDER_DATE_ID)
105 );
106
107 INSERT INTO ORDER_FACT (DIM_EMPLOYEE_FK, DIM_CUSTOMER_FK, DIM_PRODUCT_FK, ORDER_DATE_FK, ORDER_ID,
108 ITEM_ID, STATUS, QUANTITY, STANDARD_COST, LIST_PRICE)
109 SELECT COALESCE(de.DIM_EMPLOYEE_ID, 0), dc.DIM_CUSTOMER_ID, dp.DIM_PRODUCT_ID, dt.ORDER_DATE_ID,
110 o.ORDER_ID, oi.ITEM_ID, o.STATUS, oi.QUANTITY, p.STANDARD_COST, p.LIST_PRICE
111 FROM prod.ORDERS o
112 JOIN prod.ORDER_ITEMS oi ON o.ORDER_ID = oi.ORDER_ID
113 JOIN prod.PRODUCTS p ON oi.PRODUCT_ID = p.PRODUCT_ID
114 LEFT JOIN DIM_EMPLOYEE de ON o.SALESMAN_ID = de.PROD_EMPLOYEE_ID
115 JOIN DIM_CUSTOMER dc ON o.CUSTOMER_ID = dc.PROD_CUSTOMER_ID
116 JOIN DIM_PRODUCT dp ON oi.PRODUCT_ID = dp.PROD_PRODUCT_ID
117 JOIN DIM_TIME dt ON o.ORDER_DATE = dt.FULL_DATE;
```

Činjenice naše **ORDER_FACT** tabele su: **QUANTITY** (količina naručenog proizvoda), **STANDARD_COST** (trošak proizvodnje datog proizvoda iz stavke), **LIST_COST** (cijena koju potražujemo prilikom narudžbe) i **PROFIT_MARGIN** (marža ili očekivana zarada) izračunata na osnovu razlike cijene i troška proizvodnje.

Ovdje je bitno obratiti pažnju da **ORDER_ID** i **STATUS** kolone predstavljaju degenerisane dimenzije. To su podaci koji liče na dimenzije samo što se ne referenciraju preko stranog ključa (FK) već su direktno upisani u tabelu mjere. Nisu prave dimenzije, a po svome smislu, one to jesu.
ORDER_ID je redni broj narudžbe, a **STATUS** uzima vrijednosti “Canceled”, “Pending” ili “Shipped”, ovisno o stanju date narudžbe.

Problem narudžbe čiji prodavač nije poznat je riješen referenciranjem na specijalni red iz **DIM_EMPLOYEE** dimenzije čiji je **DIM_EMPLOYEE_ID** jednak **0**. Time je sačuvan zapis o narudžbi, a uposlenik koji je nju dogovorio je označen kao “Unknown” (nepoznat).

Kreiranje i punjenje INVENTORY_FACT tabele mjere

```
119  CREATE TABLE INVENTORY_FACT (
120      ITEM_NUMBER INTEGER PRIMARY KEY AUTOINCREMENT,
121      DIM_PRODUCT_FK INTEGER,
122      DIM_WAREHOUSE_FK INTEGER,
123      QUANTITY INTEGER,
124      STANDARD_COST REAL NOT NULL,
125      LIST_PRICE REAL NOT NULL,
126      PROFIT_MARGIN REAL NOT NULL GENERATED ALWAYS AS (ROUND(LIST_PRICE - STANDARD_COST, 2)) STORED,
127      FOREIGN KEY (DIM_PRODUCT_FK) REFERENCES DIM_PRODUCT (DIM_PRODUCT_ID),
128      FOREIGN KEY (DIM_WAREHOUSE_FK) REFERENCES DIM_WAREHOUSE (DIM_WAREHOUSE_ID)
129  );
130
131  INSERT INTO INVENTORY_FACT (DIM_PRODUCT_FK, DIM_WAREHOUSE_FK, QUANTITY, STANDARD_COST, LIST_PRICE)
132  SELECT dp.DIM_PRODUCT_ID, dw.DIM_WAREHOUSE_ID, i.QUANTITY, p.STANDARD_COST, p.LIST_PRICE
133  FROM prod.INVENTORIES i
134  JOIN prod.PRODUCTS p ON i.PRODUCT_ID = p.PRODUCT_ID
135  JOIN DIM_PRODUCT dp ON i.PRODUCT_ID = dp.PROD_PRODUCT_ID
136  JOIN DIM_WAREHOUSE dw ON i.WAREHOUSE_ID = dw.PROD_WAREHOUSE_ID;
137
```

Činjenice **INVENTORY_FACT** tabele su: **QUANTITY** (količina proizvoda na stanju u datom skladištu), **STANDARD_COST** (trošak proizvodnje), **LIST_PRICE** (prodajna cijena) i **PROFIT_MARGIN** (marža), slično kao za **ORDER_FACT** tabelu mjere.

Primjeri upita nad ORDER_FACT tabelom iz DWH

1. Prihodi po kvartalima

```
SELECT dt.YEAR AS Godina,
       dt.QUARTER AS Kvartal,
       CAST(SUM(ofc.QUANTITY * ofc.PROFIT_MARGIN) AS INTEGER) AS "Ukupni prihodi"
FROM ORDER_FACT ofc
JOIN DIM_TIME dt ON ofc.ORDER_DATE_FK = dt.ORDER_DATE_ID
GROUP BY dt.YEAR, dt.QUARTER
ORDER BY dt.YEAR, dt.QUARTER;
```

	Godina	Kvartal	Ukupni prihodi
1	2013	2	119147
2	2015	2	922253
3	2015	4	678221
4	2016	1	193937
5	2016	2	837584
6	2016	3	1521790
7	2016	4	1537516
8	2017	1	2345403
9	2017	2	587421
10	2017	3	891141
11	2017	4	500349

2. Ostvareni profit po kategoriji proizvoda

```
SELECT dp.CATEGORY_NAME AS "Ime kategorije",
       SUM(ofc.QUANTITY) AS "Ukupno narudžbi",
       CAST(SUM(ofc.QUANTITY * ofc.PROFIT_MARGIN) AS INTEGER) AS "Ukupni profit",
       CAST(SUM(ofc.QUANTITY * ofc.PROFIT_MARGIN)/NULLIF(SUM(ofc.QUANTITY), 0) AS INTEGER)
       AS "Zarada po jedinici proizvoda"
FROM ORDER_FACT ofc
JOIN DIM_PRODUCT dp ON ofc.DIM_PRODUCT_FK = dp.DIM_PRODUCT_ID
WHERE ofc.STATUS != 'Pending'
GROUP BY dp.CATEGORY_NAME
ORDER BY "Ukupni profit" DESC;
```

	Ime kategorije	Ukupno narudžbi	Ukupni profit	Zarada po jedinici proizvoda
1	CPU	12176	3042690	249
2	Storage	19707	2411529	122
3	Video Card	7844	2079552	265
4	Mother Board	9646	827828	85

3. Top 5 kompanija kupaca po ukupnoj potrošnji

```
SELECT dc.COMPANY_NAME AS "Ime kompanije kupca",
       CAST(SUM(ofc.QUANTITY * ofc.LIST_PRICE) AS INTEGER) AS "Ukupna potrošnja kompanije kupca",
       CAST(SUM(ofc.QUANTITY * ofc.PROFIT_MARGIN) AS INTEGER) AS "Naš ukupni profit poslovanja s njima"
FROM ORDER_FACT ofc
JOIN DIM_CUSTOMER dc ON ofc.DIM_CUSTOMER_FK = dc.DIM_CUSTOMER_ID
GROUP BY dc.COMPANY_NAME
ORDER BY "Ukupna potrošnja kompanije kupca" DESC
LIMIT 5;
```

	Ime kompanije kupca	Ukupna potrošnja kompanije kupca	Naš ukupni profit poslovanja s njima
1	General Mills	3725138	767328
2	Jabil Circuit	3334221	688370
3	Emerson Electric	2893564	475754
4	Raytheon	2778083	503361
5	International Paper	2642238	502036

4. Uspjeh uposlenika prodavača rangirano opadajuće

```
SELECT de.FULL_NAME AS "Ime uposlenika",
       COUNT(DISTINCT ofc.ORDER_ID) AS "Broj narudžbi",
       CAST(SUM(ofc.QUANTITY * ofc.LIST_PRICE) AS INTEGER)
       AS "Ukupna vrijednost dogovorenih narudžbi"
FROM ORDER_FACT ofc
JOIN DIM_EMPLOYEE de ON ofc.DIM_EMPLOYEE_FK = de.DIM_EMPLOYEE_ID
GROUP BY de.FULL_NAME
ORDER BY "Ukupna vrijednost dogovorenih narudžbi" DESC;
```

	Ime uposlenika	Broj narudžbi	Ukupna vrijednost dogovorenih narudžbi
1	Unknown	35	18245463
2	Freya Gomez	13	8081332
3	Florence Freeman	12	4341842
4	Chloe Cruz	7	3900172
5	Grace Ellis	10	3525462
6	Scarlett Gibson	5	3522704
7	Daisy Ortiz	6	3252131
8	Isabelle Marshall	7	3233737
9	Evie Harrison	5	2754951
10	Lily Fisher	5	1884295

5. Mjesečni trendovi po statusu narudžbi

```
1  SELECT dt.YEAR AS Godina,
2         dt.MONTH AS Mjesec,
3         ofc.STATUS AS Status,
4         COUNT(ofc.ORDER_ID) AS "Ukupno narudžbi",
5         CAST(SUM(ofc.QUANTITY * ofc.PROFIT_MARGIN) AS INTEGER) AS Profit
6  FROM ORDER_FACT ofc
7  JOIN DIM_TIME dt ON ofc.ORDER_DATE_FK = dt.ORDER_DATE_ID
8  GROUP BY dt.YEAR, dt.MONTH, ofc.STATUS
9  ORDER BY dt.YEAR, dt.MONTH, Profit;
```

	Godina	Mjesec	Status	Ukupno narudžbi	Profit
1	2013	6	Shipped	6	119147
2	2015	4	Shipped	24	332016
3	2015	5	Shipped	8	155919
4	2015	5	Canceled	14	245020
5	2015	6	Shipped	10	189297
6	2015	10	Pending	8	119512
7	2015	10	Shipped	15	304681
8	2015	12	Shipped	6	73189
9	2015	12	Pending	6	180838
10	2016	2	Pending	2	18323
11	2016	2	Shipped	8	175614
12	2016	5	Canceled	13	101975

Primjeri upita nad INVENTORY_FACT tabelom iz DWH

1. Top 3 proizvoda na stanju po različitim skladištima

```sql
WITH RangiraniProizvodi AS (
    SELECT dw.CITY_NAME AS "Lokacija skladišta",
           dp.CATEGORY_NAME AS "Kategorija proizvoda",
           dp.PRODUCT_NAME AS "Ime proizvoda",
           SUM(ifc.QUANTITY) AS "Trenutno stanje",
           ROW_NUMBER() OVER (
               PARTITION BY dw.CITY_NAME
               ORDER BY SUM(ifc.QUANTITY) DESC
           ) AS br_reda
    FROM INVENTORY_FACT ifc
    JOIN DIM_WAREHOUSE dw ON ifc.DIM_WAREHOUSE_FK = dw.DIM_WAREHOUSE_ID
    JOIN DIM_PRODUCT dp ON ifc.DIM_PRODUCT_FK = dp.DIM_PRODUCT_ID
    GROUP BY dw.CITY_NAME, dp.PRODUCT_NAME
)
SELECT "Ime proizvoda",
       "Kategorija proizvoda",
       "Lokacija skladišta",
       "Trenutno stanje"
FROM RangiraniProizvodi
WHERE br_reda <= 3
ORDER BY "Lokacija skladišta" ASC, "Trenutno stanje" DESC;
```

	Ime proizvoda	Kategorija proizvoda	Lokacija skladišta	Trenutno stanje
1	G.Skill Ripjaws V Series	Storage	Beijing	636
2	G.Skill Trident Z	Storage	Beijing	530
3	Corsair Vengeance LPX	Storage	Beijing	524
4	G.Skill Ripjaws V Series	Storage	Bombay	488
5	G.Skill Trident Z	Storage	Bombay	426
6	Corsair Vengeance LPX	Storage	Bombay	321
7	Kingston SA400S37/120G	Storage	Mexico City	294
8	Zotac ZT-P10810D-10P	Video Card	Mexico City	196
9	Gigabyte GV-N1070WF2OC-8GD	Video Card	Mexico City	196
10	Zotac ZT-P10810D-10P	Video Card	New Jersey	304

2. Skladišta sa najvećom vrijednošću

```
SELECT dw.CITY_NAME AS "Lokacija skladišta",
       CAST(SUM(ifc.QUANTITY * ifc.STANDARD_COST) AS INTEGER) AS "Ukupna vrijednost",
       CAST(SUM(ifc.QUANTITY * ifc.PROFIT_MARGIN) AS INTEGER) AS "Potencijalni profit"
FROM INVENTORY_FACT ifc
JOIN DIM_WAREHOUSE dw ON ifc.DIM_WAREHOUSE_FK = dw.DIM_WAREHOUSE_ID
GROUP BY dw.CITY_NAME
ORDER BY "Ukupna vrijednost" DESC;
```

	Lokacija skladišta	Ukupna vrijednost	Potencijalni profit
1	San Francisco	18743692	4646804
2	Sydney	13065912	3224296
3	Seattle, Washington	11338582	2847706
4	Toronto	8873312	2196216
5	New Jersey	8306634	2044929
6	Beijing	8223708	2041737
7	Southlake, Texas	5841509	1507486
8	Bombay	5741230	1430773
9	Mexico City	5692248	1427501

3. Top 10 najzastupljenijih proizvoda u skladištima

```
SELECT dp.PRODUCT_NAME AS "Ime proizvoda",
       dp.CATEGORY_NAME AS "Kategorija proizvoda",
       SUM(ifc.QUANTITY) AS "Dostupna količina",
       dw.CITY_NAME AS "Lokacija skladišta"
FROM INVENTORY_FACT ifc
JOIN DIM_WAREHOUSE dw ON ifc.DIM_WAREHOUSE_FK = dw.DIM_WAREHOUSE_ID
JOIN DIM_PRODUCT dp ON ifc.DIM_PRODUCT_FK = dp.DIM_PRODUCT_ID
GROUP BY dp.PRODUCT_NAME, dw.CITY_NAME
ORDER BY "Dostupna količina" DESC
LIMIT 10;
```

	Ime proizvoda	Kategorija proizvoda	Dostupna količina	Lokacija skladišta
1	G.Skill Ripjaws V Series	Storage	1894	San Francisco
2	G.Skill Trident Z	Storage	1499	San Francisco
3	G.Skill Ripjaws V Series	Storage	1279	Seattle, Washington
4	Corsair Vengeance LPX	Storage	1203	San Francisco
5	G.Skill Trident Z	Storage	1029	Seattle, Washington
6	G.Skill Ripjaws V Series	Storage	1021	Sydney
7	G.Skill Trident Z	Storage	881	Sydney
8	Corsair Vengeance LPX	Storage	848	Seattle, Washington
9	Corsair Dominator Platinum	Storage	786	San Francisco
10	Corsair Vengeance LPX	Storage	720	Sydney