

Pedagogická fakulta v Ružomberku, Katedra informatiky

Semestrálna práca z predmetu Databázové systémy  
(elektronický obchod športových potrieb)

---

## ----- 2. Zadanie -----

**29.** Vytvorte dátový model pre systém elektronického obchodu predaja športových potrieb. Pri každom tovare je potrebné evidovať presný názov, krajinu pôvodu, druh tovaru, cenu a stav spracovania objednávky. Systém bude evidovať aj osobné údaje zákazníkov portálu. Pomocou tohto systému bude možné objednať si napríklad korčule a prilbu pre ľadový hokej, ale aj outdoorové či trekkingové dámske športové oblečenie. Systém bude poskytovať aj sumárne výkazy predaja jednotlivých tovarov za zvolené obdobie alebo sumárny mesačný či týždenný predaj tovarov.

---

## ----- 3. Vytvorenie databázy a tabuliek -----

```
CREATE DATABASE sportovy_obchod;
```

```
CREATE TABLE mesto
```

```
(  
    id      SERIAL,  
    nazov   VARCHAR (50),  
    PSC     CHAR (5),  
    PRIMARY KEY (id)  
)
```

```
CREATE TABLE zakaznici
```

```
(  
    id          SERIAL,  
    meno        VARCHAR (20),  
    priezvisko VARCHAR (50),  
    tel_cislo   CHAR (10),  
    email       VARCHAR (50),  
    id_mesto    INTEGER,  
    ulica       VARCHAR (20),  
    cislo        VARCHAR (10),  
    PRIMARY KEY (id),  
    FOREIGN KEY (id_mesto) REFERENCES mesto (id) ON UPDATE CASCADE ON DELETE  
    CASCADE  
)
```

```
CREATE TABLE tovar
```

```
(  
    id          SERIAL,  
    druh_tovaru VARCHAR (50),  
    nazov       VARCHAR (50),  
    krajina_povodu VARCHAR (20),  
    cena        DECIMAL (4,2),  
    PRIMARY KEY (id)  
)
```

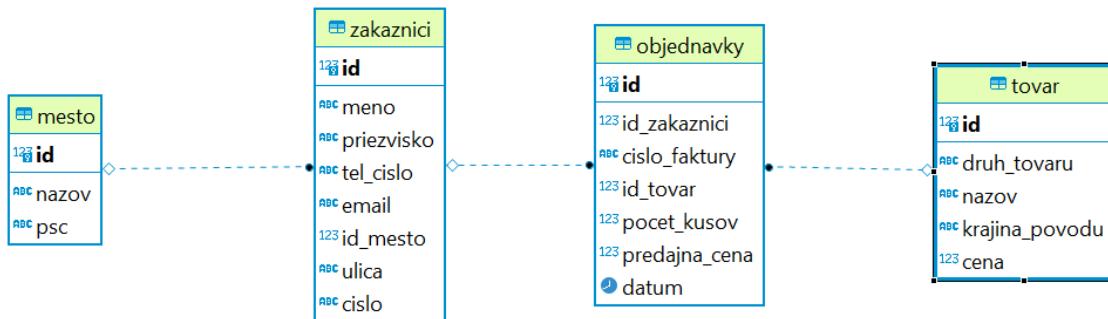
```
CREATE TABLE objednavky
```

```
(  
    id          SERIAL,  
    id_zakaznici INTEGER,  
    cislo_faktury VARCHAR (50),  
    id_tovar    INTEGER,  
    pocet_kusov INTEGER,
```

predajna\_cena DECIMAL (4,2),  
 datum DATE,  
 PRIMARY KEY (id),  
 FOREIGN KEY (id\_zakaznici) REFERENCES zakaznici (id) ON UPDATE CASCADE ON  
 DELETE CASCADE,  
 FOREIGN KEY (id\_tovar) REFERENCES tovar (id) ON UPDATE CASCADE ON DELETE  
 CASCADE  
 );

#### ----- 4. Grafický návrh tabuliek -----

Na vytvorenie grafického návrhu tabuliek som použila aplikáciu Dbeaver.



#### ----- 5. Vkladanie údajov -----

```

INSERT INTO mesto (id, nazov, PSC)
VALUES
(nextval('mesto_id_seq'), 'Zilina 1', '01001'),
(nextval('mesto_id_seq'), 'Zilina 4', '01004'),
(nextval('mesto_id_seq'), 'Cadca', '02201'),
(nextval('mesto_id_seq'), 'Ruzomberok', '04401'),
(nextval('mesto_id_seq'), 'Kosice', '04018'),
(nextval('mesto_id_seq'), 'Banska Bystrica', '97401'),
(nextval('mesto_id_seq'), 'Nemestovo', '02901'),
(nextval('mesto_id_seq'), 'Poprad 1', '05801'),
(nextval('mesto_id_seq'), 'Trnava', '91701'),
(nextval('mesto_id_seq'), 'Liptovsky Mikulas', '03101'),
(nextval('mesto_id_seq'), 'Presov', '08001')
;
  
```

```

INSERT INTO zakaznici (id, meno, priezvisko, tel_cislo, email, id_mesto, ulica, cislo)
VALUES
(nextval('zakaznici_id_seq'), 'Milan', 'Dobry', '0903007112', 'milan.dobry@gmail.com', '1', 'Okruzna', '25'),
(nextval('zakaznici_id_seq'), 'Alena', 'Vesela', '0949007112', 'alena.vesela@gmail.com', '3', 'Bukov', '736'),
(nextval('zakaznici_id_seq'), 'Juraj', 'Nizky', '0904558963', 'juraj.nizky@gmail.com', '4', 'Bottova', '12'),
(nextval('zakaznici_id_seq'), 'Anna', 'Kratka', '0903131902', 'anna.kratka@gmail.com', '2', 'Benku', '99'),
(nextval('zakaznici_id_seq'), 'Kata', 'Drevena', '0904006512', 'kata.drevena@gmail.com', '6', '29 augusta', '135'),
(nextval('zakaznici_id_seq'), 'Peter', 'Rovny', '0908445779', 'peter.rovny@gmail.com', '5', 'Adamova', '5'),
(nextval('zakaznici_id_seq'), 'Jana', 'Dobra', '0904187563', 'jana.dobra@gmail.com', '5', 'Benkova', '557'),
(nextval('zakaznici_id_seq'), 'Jan', 'Mily', '0907787563', 'jan.mily@gmail.com', '7', 'Slnecnna', '57'),
(nextval('zakaznici_id_seq'), 'Olina', 'Kratka', '0949187563', 'olina.kratka@gmail.com', '9', 'Atomova', '55'),
  
```

```

(nextval('zakaznici_id_seq'), 'Aneta', 'Drza', '0904188900', 'aneta.drza@gmail.com', '8', 'Bajkalska', '93'),
(nextval('zakaznici_id_seq'), 'Alena', 'Nahnevana', '0904187563', 'alena.nahnevana@gmail.com', '10', 'Bottova',
  '11'),
(nextval('zakaznici_id_seq'), 'Zuzana', 'Nudna', '0904765563', 'zuzana.nudna@gmail.com', '11', 'Hlavna', '992'),
(nextval('zakaznici_id_seq'), 'Tibor', 'Nahly', '0908907563', 'tibor.nahly@gmail.com', '9', 'Bellova', '20'),
(nextval('zakaznici_id_seq'), 'Natalia', 'Vesela', '0904185478', 'natalia.vesela@gmail.com', '6', 'Sasova', '9'),
(nextval('zakaznici_id_seq'), 'Juraj', 'Smutny', '0904109587', 'juraj.smutny@gmail.com', '2', 'Vlcince', '332'),
(nextval('zakaznici_id_seq'), 'Jana', 'Mudra', '0909189733', 'jana.mudra@gmail.com', '7', 'Veterna', '971'),
(nextval('zakaznici_id_seq'), 'Lubomir', 'Mudry', '0904274203', 'lubo.mudry@gmail.com', '5', 'Benkova', '12')
;

```

INSERT INTO tovar (id, druh\_tovaru, nazov, krajina\_povodu, cena)  
VALUES

```

(nextval('tovar_id_seq'), 'zimne sporty', 'hokejka', 'Cina', 125),
(nextval('tovar_id_seq'), 'zimne sporty', 'korcule', 'Nemecko', 45),
(nextval('tovar_id_seq'), 'zimne sporty', 'prilba na ladovy hokej', 'Turecko', 77),
(nextval('tovar_id_seq'), 'zimne sporty', 'lyze', 'Nemecko', 355),
(nextval('tovar_id_seq'), 'obuv', 'tenisky', 'Cina', '53'),
(nextval('tovar_id_seq'), 'obuv', 'turisticke topinky', 'Cina', 99),
(nextval('tovar_id_seq'), 'obuv', 'kopacky', 'Nemecko', 67),
(nextval('tovar_id_seq'), 'obuv', 'tretry', 'Polsko', 117),
(nextval('tovar_id_seq'), 'oblecenie', 'tricko', 'Slovensko', 33),
(nextval('tovar_id_seq'), 'oblecenie', 'kratasy', 'Posko', 14),
(nextval('tovar_id_seq'), 'oblecenie', 'plavky', 'Cina', 17),
(nextval('tovar_id_seq'), 'oblecenie', 'teplaky', 'Rakusko', 22),
(nextval('tovar_id_seq'), 'oblecenie', 'ciapka', 'Cina', 5),
(nextval('tovar_id_seq'), 'oblecenie', 'kupacia ciapka', 'Cina', 3),
(nextval('tovar_id_seq'), 'cyklistika', 'cyklisticke kratasy', 'Polsko', 30),
(nextval('tovar_id_seq'), 'cyklistika', 'cyklisticka prilba', 'Cina', 77),
(nextval('tovar_id_seq'), 'cyklistika', 'cyklisticka vetrovka', 'Turecko', 125),
(nextval('tovar_id_seq'), 'cyklistika', 'kolobezka', 'Francuzsko', 125),
(nextval('tovar_id_seq'), 'cyklistika', 'horsky bicykel', 'Taliansko', 553),
(nextval('tovar_id_seq'), 'cyklistika', 'elektrobickykel', 'Spanielsko', 1199),
(nextval('tovar_id_seq'), 'cyklistika', 'BMX', 'Nemecko', 250),
(nextval('tovar_id_seq'), 'cyklistika', 'mestsky bicykel', 'Turecko', 280),
(nextval('tovar_id_seq'), 'letne sporty', 'tenisova raketa', 'Cina', 213),
(nextval('tovar_id_seq'), 'letne sporty', 'kolieskove korcule', 'Slovensko', 65),
(nextval('tovar_id_seq'), 'letne sporty', 'futbalova lopta', 'Ceska Republika', 33),
(nextval('tovar_id_seq'), 'letne sporty', 'trekingove palice', 'Madarsko', 20),
(nextval('tovar_id_seq'), 'letne sporty', 'tenisova lopta', 'Cina', 7),
(nextval('tovar_id_seq'), 'letne sporty', 'skateboard', 'Cina', 73),
(nextval('tovar_id_seq'), 'letne sporty', 'pennyboard', 'Anglicko', 32),
(nextval('tovar_id_seq'), 'letne sporty', 'Batoh', 'Turecko', 42),
(nextval('tovar_id_seq'), 'letne sporty', 'badmintonovy kosik', 'Cina', 10),
(nextval('tovar_id_seq'), 'letne sporty', 'badmintonova sada', 'Ceska Republika', 35),
(nextval('tovar_id_seq'), 'letne sporty', 'badmintonova siet', 'Slovensko', 27)
;
```

INSERT INTO objednavky (id, id\_zakaznici, cislo\_faktury, id\_tovar, pocet\_kusov, predajna\_cena, datum)

VALUES

```

(nextval('objednavky_id_seq'), '1', '1001', '30', '1,50', '01.01.2020'),
(nextval('objednavky_id_seq'), '1', '1001', '23', '2,220', '01.01.2020'),
(nextval('objednavky_id_seq'), '1', '1001', '27', '1,8', '01.01.2020'),
(nextval('objednavky_id_seq'), '2', '1002', '9', '1,35', '02.01.2020'),

```

```

(nextval('objednavky_id_seq'), '2', '1002', '13', 1, 6, '02.01.2020'),
(nextval('objednavky_id_seq'), '2', '1002', '11', 1, 25, '02.01.2020'),
(nextval('objednavky_id_seq'), '3', '1003', '2', 1, 49, '02.01.2020'),
(nextval('objednavky_id_seq'), '4', '1004', '15', 1, 35, '04.01.2020'),
(nextval('objednavky_id_seq'), '4', '1004', '16', 1, 80, '04.01.2020'),
(nextval('objednavky_id_seq'), '5', '1005', '28', 1, 78, '13.01.2020'),
(nextval('objednavky_id_seq'), '6', '1006', '32', 1, 39, '05.02.2020'),
(nextval('objednavky_id_seq'), '6', '1006', '33', 1, 30, '05.02.2020'),
(nextval('objednavky_id_seq'), '6', '1006', '31', 2, 11, '05.02.2020'),
(nextval('objednavky_id_seq'), '7', '1007', '3', 1, 80, '17.02.2020'),
(nextval('objednavky_id_seq'), '8', '1008', '4', 1, 360, '23.02.2020'),
(nextval('objednavky_id_seq'), '9', '1009', '5', 1, 58, '11.03.2020'),
(nextval('objednavky_id_seq'), '10', '1010', '7', 1, 75, '23.02.2020'),
(nextval('objednavky_id_seq'), '10', '1010', '25', 1, 35, '23.02.2020'),
(nextval('objednavky_id_seq'), '11', '1011', '5', 1, 55, '23.02.2020'),
(nextval('objednavky_id_seq'), '5', '1012', '8', 1, 120, '23.02.2020'),
(nextval('objednavky_id_seq'), '7', '1013', '28', 1, 75, '23.02.2020'),
(nextval('objednavky_id_seq'), '12', '1014', '6', 1, 101, '14.03.2020'),
(nextval('objednavky_id_seq'), '13', '1015', '10', 2, 16, '23.03.2020'),
(nextval('objednavky_id_seq'), '13', '1015', '14', 1, 5, '23.03.2020'),
(nextval('objednavky_id_seq'), '13', '1015', '11', 1, 25, '23.03.2020'),
(nextval('objednavky_id_seq'), '14', '1016', '12', 2, 25, '30.03.2020'),
(nextval('objednavky_id_seq'), '14', '1016', '18', 1, 127, '30.03.2020'),
(nextval('objednavky_id_seq'), '15', '1017', '17', 1, 127, '31.03.2020'),
(nextval('objednavky_id_seq'), '16', '1018', '19', 1, 560, '14.04.2020'),
(nextval('objednavky_id_seq'), '17', '1019', '20', 1, 1300, '15.04.2020'),
(nextval('objednavky_id_seq'), '16', '1020', '21', 1, 271, '16.04.2020'),
(nextval('objednavky_id_seq'), '4', '1021', '22', 1, 300, '17.04.2020'),
(nextval('objednavky_id_seq'), '4', '1021', '16', 1, 80, '17.04.2020'),
(nextval('objednavky_id_seq'), '10', '1022', '24', 1, 70, '20.04.2020'),
(nextval('objednavky_id_seq'), '10', '1022', '16', 1, 80, '20.04.2020'),
(nextval('objednavky_id_seq'), '17', '1023', '26', 4, 22, '24.04.2020'),
(nextval('objednavky_id_seq'), '17', '1023', '29', 1, 34, '24.04.2020'),
(nextval('objednavky_id_seq'), '17', '1023', '26', 4, 22, '24.04.2020'),
(nextval('objednavky_id_seq'), '13', '1024', '30', 2, 50, '17.05.2020'),
(nextval('objednavky_id_seq'), '2', '1025', '25', 3, 35, '23.05.2020'),
(nextval('objednavky_id_seq'), '2', '1025', '7', 2, 75, '23.05.2020')
;

```

## ----- 6. Vytvorenie skupín, používateľov a práv -----

CREATE GROUP spravca;  
CREATE GROUP zamestnanci;

CREATE USER riaditel IN GROUP spravca;  
CREATE USER zastupca\_riaditela in group spravca;  
CREATE USER administrator IN GROUP zamestnanci;  
CREATE USER uctovnik IN GROUP zamestnanci;

GRANT ALL ON mesto, zakaznici, tovar, objednavky TO GROUP spravca;  
GRANT SELECT, INSERT ON zakaznici, tovar TO GROUP zamestnanci;  
GRANT ALL ON mesto, zakaznici, tovar, objednavky TO GROUP zamestnanci;  
REVOKE DELETE ON mesto FROM GROUP zamestnanci;

## ----- 7. Výber údajov -----

```
select * from objednavky where id=33;
```

sportovy_obchod=# select * from objednavky where id=33;						
id	id_zakaznici	cislo_faktury	id_tovar	pocet_kusov	predajna_cena	datum
33	4	1021		16	1	80.00   2020-04-17
(1 row)						

```
select * from objednavky where predajna_cena>100;
```

sportovy_obchod=# select * from objednavky where predajna_cena>100;						
id	id_zakaznici	cislo_faktury	id_tovar	pocet_kusov	predajna_cena	datum
2	1	1001		23	2	220.00   2020-01-01
15	8	1008		4	1	360.00   2020-02-23
20	5	1012		8	1	120.00   2020-02-23
22	12	1014		6	1	101.00   2020-03-14
27	14	1016		18	1	127.00   2020-03-30
28	15	1017		17	1	127.00   2020-03-31
29	16	1018		19	1	560.00   2020-04-14
30	17	1019		20	1	1300.00   2020-04-15
31	16	1020		21	1	271.00   2020-04-16
32	4	1021		22	1	300.00   2020-04-17
(10 rows)						

```
select * from objednavky limit 7 offset 10;
```

sportovy_obchod=# select * from objednavky limit 7 offset 10;						
id	id_zakaznici	cislo_faktury	id_tovar	pocet_kusov	predajna_cena	datum
11	6	1006		32	1	39.00   2020-02-05
12	6	1006		33	1	30.00   2020-02-05
13	6	1006		31	2	11.00   2020-02-05
14	7	1007		3	1	80.00   2020-02-17
15	8	1008		4	1	360.00   2020-02-23
16	9	1009		5	1	58.00   2020-03-11
17	10	1010		7	1	75.00   2020-02-23
(7 rows)						

```
select meno, priezvisko, tel_cislo, email, id_mesto from zakaznici;
```

sportovy_obchod=# select meno, priezvisko, tel_cislo, email, id_mesto from zakaznici;				
meno	priezvisko	tel_cislo	email	id_mesto
Milan	Dobry	0903007112	milan.dobry@gmail.com	1
Alena	Vesela	0949007112	alena.vesela@gmail.com	3
Juraj	Nizky	0904558963	juraj.nizky@gmail.com	4
Anna	Kratka	0903131902	anna.kratka@gmail.com	2
Kata	Drevena	0904006512	kata.drevena@gmail.com	6
Peter	Rovny	0908445779	peter.rovny@gmail.com	5
Jana	Dobra	0904187563	jana.dobra@gmail.com	5
Jan	Mily	0907787563	jan.mily@gmail.com	7
Olina	Kratka	0949187563	olina.kratka@gmail.com	9
Aneta	Drza	0904188900	aneta.drza@gmail.com	8
Alena	Nahnevana	0904187563	alena.nahnevana@gmail.com	10
Zuzana	Nudna	0904765563	zuzana.nudna@gmail.com	11
Tibor	Nahly	0908907563	tibor.nahly@gmail.com	9
Natalia	Vesela	0904185478	natalia.vesela@gmail.com	6
Juraj	Smutny	0904109587	juraj.smutny@gmail.com	2
Jana	Mudra	0909189733	jana.mudra@gmail.com	7
Lubomir	Mudry	0904274203	lubo.mudry@gmail.com	5
(17 rows)				

```
select trim(meno)|| ' '||trim(priezvisko) as meno from zakaznici;
```

```
sportovy_obchod=# select trim(meno)|| ' '||trim(priezvisko) as meno from zakaznici;
               meno
-----
 Milan Dobry
 Alena Vesela
 Juraj Nizky
 Anna Kratka
 Kata Drevna
 Peter Rovny
 Jana Dobra
 Jan Mily
 Olina Kratka
 Aneta Drza
 Alena Nahnevana
 Zuzana Nudna
 Tibor Nahly
 Natalia Vesela
 Juraj Smutny
 Jana Mudra
 Lubomir Mudry
(17 rows)
```

```
select priezvisko ||' '| meno as zakaznik, id_mesto, ulica, cislo from zakaznici;
```

```
sportovy_obchod=# select priezvisko ||' '| meno as zakaznik, id_mesto, ulica, cislo from zakaznici;
   zakaznik | id_mesto | ulica | cislo
-----
 Dobry Milan | 1 | Okruzna | 25
 Vesela Alena | 3 | Bukov | 736
 Nizky Juraj | 4 | Bottova | 12
 Kratka Anna | 2 | Benku | 99
 Drevna Kata | 6 | 29 augusta | 135
 Rovny Peter | 5 | Adamova | 5
 Dobra Jana | 5 | Benkova | 557
 Mily Jan | 7 | Slnecna | 57
 Kratka Olina | 9 | Atomova | 55
 Drza Aneta | 8 | Bajkalska | 93
 Nahnevana Alena | 10 | Bottova | 11
 Nudna Zuzana | 11 | Hlavna | 992
 Nahly Tibor | 9 | Bellova | 20
 Vesela Natalia | 6 | Sasova | 9
 Smutny Juraj | 2 | Vlcince | 332
 Mudra Jana | 7 | Veterna | 971
 Mudry Lubomir | 5 | Benkova | 12
(17 rows)
```

```
select druh_tovaru, nazov, count (*), sum (cena) from tovar left outer join objednavky on
(tovar.id=objednavky.id_tovar) group by druh_tovaru, tovar.nazov;
```

```
sportovy_obchod=# select druh_tovaru, nazov, count (*), sum (cena) from tovar left outer join objednavky on (tovar.id=objednavky.id_tovar) group by druh_tovaru, tovar.nazov
;+-----+-----+-----+-----+
| druh_tovaru | nazov | count | sum |
+-----+-----+-----+-----+
| letne sporty | tenisova raketa | 1 | 213.00
| letne sporty | koleskove korculle | 1 | 65.00
| oblecenie | kupacia ciapka | 1 | 3.00
| letne sporty | futbalova lopta | 2 | 66.00
| obuv | kopacky | 2 | 134.00
| oblecenie | kratasy | 1 | 14.00
| zimne sporty | lyze | 1 | 355.00
| letne sporty | badmintonovy kosik | 1 | 10.00
| oblecenie | tricko | 1 | 33.00
| obuv | tenisky | 2 | 106.00
| cyklistika | cyklisticka vetrovka | 1 | 125.00
| letne sporty | trekkingove palice | 2 | 48.00
| obuv | tretrey | 1 | 117.00
| letne sporty | badmintonova siet | 1 | 27.00
| cyklistika | kolobezka | 1 | 125.00
| letne sporty | skateboard | 2 | 146.00
| oblecenie | plavky | 2 | 34.00
| obuv | turisticke topanky | 1 | 99.00
| cyklistika | elektrobicykel | 1 | 1199.00
| oblecenie | ciapka | 1 | 5.00
| zimne sporty | hokejka | 1 | 125.00
| letne sporty | pennyboard | 1 | 32.00
| cyklistika | cyklisticke kratasy | 1 | 30.00
| cyklistika | cyklisticka prilba | 3 | 231.00
| zimne sporty | korculle | 1 | 45.00
| cyklistika | mestsky bicykel | 1 | 280.00
| letne sporty | badmintonova sada | 1 | 35.00
| cyklistika | BMX | 1 | 250.00
| letne sporty | tenisova lopta | 1 | 7.00
| letne sporty | Batoh | 2 | 84.00
| oblecenie | teplaky | 1 | 22.00
| cyklistika | horsky bicykel | 1 | 553.00
| zimne sporty | prilba na ladowy hokej | 1 | 77.00
(33 rows)
```

```
select nazov, count (*) from zakaznici left outer join mesto on (mesto.id=zakaznici.id_mesto) group by nazov, mesto.nazov having count (*)>1;
```

nazov	count
Trnava	2
Banska Bystrica	2
Namestovo	2
Kosice	3
Zilina 4	2
(5 rows)	

## ----- 8. Pohľady a agregácie -----

```
create view prijem_za_rok as select extract (month from datum) as mesiac, extract (year from datum) as rok, sum (predajna_cena) from objednavky where predajna_cena>0 group by 1,2 union select extract (month from datum) as mesiac, extract (year from datum) as rok, sum (predajna_cena) from objednavky where predajna_cena<0 group by 1,2 order by 1,2,3 desc;
```

List of relations			
Schema	Name	Type	Owner
public	mesto	table	postgres
public	mesto_id_seq	sequence	postgres
public	objednavky	table	postgres
public	objednavky_id_seq	sequence	postgres
public	prijem_za_rok	view	postgres
public	tovar	table	postgres
public	tovar_id_seq	sequence	postgres
public	zakaznici	table	postgres
public	zakaznici_id_seq	sequence	postgres
(9 rows)			

```
select * from prijem_za_rok;
```

mesiac	rok	sum
1	2020	586.00
2	2020	880.00
3	2020	484.00
4	2020	2739.00
5	2020	160.00
(5 rows)		

```
select druh_tovaru, nazov, pocet_kusov, datum from tovar left outer join objednavky on (objednavky.id_tovar=tovar.id) where extract (year from datum) = 2020 and extract (month from datum) = 1 and extract (week from datum) = 1;
```

druh_tovaru	nazov	pocet_kusov	datum
letne sporty	Batch	1	2020-01-01
letne sporty	tenisova raketa	2	2020-01-01
letne sporty	tenisova lopta	1	2020-01-01
oblecenie	tricko	1	2020-01-02
oblecenie	ciapka	1	2020-01-02
oblecenie	plavky	1	2020-01-02
zimne sporty	korucle	1	2020-01-02
cyklistika	cyklisticke kratasy	1	2020-01-04
cyklistika	cyklisticka prilba	1	2020-01-04
(9 rows)			

```
select priezvisko, meno, datum, predajna_cena from objednavky left outer join zakaznici on (zakaznici.id=objednavky.id) where datum in ('01.01.2020','04.01.2020','1.3.2020');
```

```
sportovy_obchod=# select priezvisko, meno, datum, predajna_cena from objednavky left outer join zakaznici on (zakaznici.id=objednavky.id) where datum in ('01.01.2020','04.01.2020','1.3.2020');
priezvisko | meno | datum | predajna_cena
-----+-----+-----+
Dobry | Milan | 2020-01-01 |      50.00
Vesela | Alena | 2020-01-01 |     220.00
Nizky | Juraj | 2020-01-01 |      8.00
Mily | Jan | 2020-01-04 |     35.00
Kratka | Olina | 2020-01-04 |     80.00
(5 rows)
```

```
select druh_tovaru, nazov, pocet_kusov, datum from tovar left outer join objednavky on (objednavky.id_tovar=tovar.id) where extract (year from datum) = 2020 and extract (month from datum) =2;
```

```
sportovy_obchod=# select druh_tovaru, nazov, pocet_kusov, datum from tovar left outer join objednavky on (objednavky.id_tovar=tovar.id) where extract (year from datum) = 2020 and extract (month from datum) =2;
druh_tovaru | nazov | pocet_kusov | datum
-----+-----+-----+
letne sporty | badmintonova sada | 1 | 2020-02-05
letne sporty | badmintonova siet | 1 | 2020-02-05
letne sporty | badmintonovy kosik | 2 | 2020-02-05
zimne sporty | prilba na ladowy hokej | 1 | 2020-02-17
zimne sporty | lyze | 1 | 2020-02-23
obuv | kopacky | 1 | 2020-02-23
obuv | futbalova lopta | 1 | 2020-02-23
obuv | tenisky | 1 | 2020-02-23
obuv | tretry | 1 | 2020-02-23
letne sporty | skateboard | 1 | 2020-02-23
(10 rows)
```

```
select druh_tovaru, nazov, pocet_kusov, datum from tovar left outer join objednavky on (objednavky.id_tovar=tovar.id) where extract (year from datum) = 2020 and extract (month from datum) = 5 and extract (day from datum) between 1 and 30;
```

```
sportovy_obchod=# select druh_tovaru, nazov, pocet_kusov, datum from tovar left outer join objednavky on (objednavky.id_tovar=tovar.id) where extract (year from datum) = 2020 and extract (month from datum) = 5 and extract (day from datum) between 1 and 30;
druh_tovaru | nazov | pocet_kusov | datum
-----+-----+-----+
letne sporty | Batoh | 2 | 2020-05-17
letne sporty | futbalova lopta | 3 | 2020-05-23
obuv | kopacky | 2 | 2020-05-23
(3 rows)
```

```
create view platby as select meno, priezvisko, sum(cena) from tovar left outer join zakaznici on (zakaznici.id=tovar.id) group by meno, priezvisko order by meno;
```

```
sportovy_obchod=# create view platby as select meno, priezvisko, sum(cena) from tovar left outer join zakaznici on (zakaznici.id=tovar.id) group by meno, priezvisko order by meno;
CREATE VIEW
sportovy_obchod=# \d
   List of relations
 Schema |        Name         | Type  | Owner
-----+-----+-----+
 public |      mesto       | table | postgres
 public | mesto_id_seq    | sequence | postgres
 public | objednavky     | table | postgres
 public | objednavky_id_seq | sequence | postgres
 public | platby          | view  | postgres
 public | prijem_za_rok   | view  | postgres
 public | tovar           | table | postgres
 public | tovar_id_seq    | sequence | postgres
 public | zakaznici       | table | postgres
 public | zakaznici_id_seq | sequence | postgres
(10 rows)
```

```
select * from platby;
```

sportovy_obchod=#	select * from platby;	
meno	priezvisko	sum
Alena	Vesela	45.00
Alena	Nahnevana	17.00
Aneta	Drza	14.00
Anna	Kratka	355.00
Jan	Mily	117.00
Jana	Mudra	77.00
Jana	Dobra	67.00
Juraj	Smutny	30.00
Juraj	Nizky	77.00
Kata	Drevena	53.00
Lubomir	Mudry	125.00
Milan	Dobry	125.00
Natalia	Vesela	3.00
Olina	Kratka	33.00
Peter	Rovny	99.00
Tibor	Nahly	5.00
Zuzana	Nudna	22.00
		2964.00

(18 rows)

## ----- 9. Integritné obmedzenia -----

```
alter table objednavky alter column pocet_kusov set not null;
alter table mesto alter column PSC set not null;
alter table zakaznici add constraint priezvisko CHECK (length(priezvisko)>3);
alter table zakaznici alter column tel_cislo set not null;
```

## ----- 10. Indexy, pravidlá, funkcie, triggere -----

### ----- 10. a Indexy -----

```
create index zakaznici_idx on zakaznici (priezvisko, meno);
```

sportovy_obchod=# create index zakaznici_idx on zakaznici (priezvisko, meno);					
CREATE INDEX					
sportovy_obchod=# \d zakaznici;					
Table "public.zakaznici"					
Column   Type   Collation   Nullable   Default					
id	integer		not null	nextval('zakaznici_id_seq'::regclass)	
meno	character varying(20)				
priezvisko	character varying(50)				
tel_cislo	character(10)		not null	false	
email	character varying(50)				
id_mesto	integer				
ulica	character varying(20)				
cislo	character varying(10)				
Indexes:					
"zakaznici_pkey" PRIMARY KEY, btree (id)					
"zakaznici_idx" btree (priezvisko, meno)					
Check constraints:					
"priezvisko" CHECK (length(priezvisko::text) > 3)					
Foreign-key constraints:					
"zakaznici_id_mesto_fkey" FOREIGN KEY (id_mesto) REFERENCES mesto(id) ON UPDATE CASCADE ON DELETE CASCADE					
Referenced by:					
TABLE "objednavky" CONSTRAINT "objednavky_id_zakaznici_fkey" FOREIGN KEY (id_zakaznici) REFERENCES zakaznici(id) ON UPDATE CASCADE ON DELETE CASCADE					

```
create unique index zakaznici_inxemail on zakaznici (email, priezvisko);
```

sportovy_obchod=# create unique index zakaznici_inxemail on zakaznici (email, priezvisko);	
CREATE INDEX	
sportovy_obchod=#	

```
create index objednavky_idx on objednavky (datum, predajna_cena);
```

```
sportovy_obchod=# \d objednavky
                                         Table "public.objednavky"
   Column    |      Type       | Collation | Nullable | Default
---+-----+-----+-----+-----+
 id | integer |          | not null | nextval('objednavky_id_seq'::regclass)
 id_zakaznici | integer |          |          |
 cislo_faktury | character varying(50) |          |          |
 id_tovar | integer |          |          |
 pocet_kusov | integer |          |          |
 predajna_cena | numeric(10,2) |          | not null |
 datum | date |          |          |
Indexes:
 "objednavky_pkey" PRIMARY KEY, btree (id)
 "objednavky_idx" btree (datum, predajna_cena)
Foreign-key constraints:
 "objednavky_id_tovar_fkey" FOREIGN KEY (id_tovar) REFERENCES tovar(id) ON UPDATE CASCADE ON DELETE CASCADE
 "objednavky_id_zakaznici_fkey" FOREIGN KEY (id_zakaznici) REFERENCES zakaznici(id) ON UPDATE CASCADE ON DELETE CASCADE
DE
```

## ----- 10. b Rules -----

Príkaz vytvorí pravidlo, ktoré pri mazaní z tabuľky mesto nevykoná nič.

```
create rule mesto_rule as on delete to mesto do instead nothing;
```

Pred vytvorením pravidla, vytvorím tabuľku pre zálohu zákazníkov:

```
CREATE TABLE zakaznici_zaloha
```

```
(  
pc           SERIAL,  
id           SERIAL,  
meno         VARCHAR (20),  
priezvisko  VARCHAR (50),  
tel_cislo    CHAR (10),  
email        VARCHAR (50),  
id_mesto    INTEGER,  
ulica        VARCHAR (20),  
cislo         VARCHAR (10),  
zmena        TIMESTAMP  
);
```

```
create rule zakaznici_rule as on delete to zakaznici do insert into zakaznici_zaloha
```

```
select nextval ('zakaznici_zaloha_pc_seq'), id, meno, priezvisko, tel_cislo, email, id_mesto, ulica,  
cislo, now() from zakaznici where id=old.id;
```

```
delete from zakaznici where id=5;
```

```
select * from zakaznici_zaloha;
```

```
sportovy_obchod=# select * from zakaznici_zaloha;
 pc | id | meno | priezvisko | tel_cislo | email | id_mesto | ulica | cislo | zmena
---+---+---+---+---+---+---+---+---+---+
 1 | 5 | Kata | Drevena | 0904006512 | kata.drevena@gmail.com | 6 | 29 augusta | 135 | 2021-05-15 23:30:31.419482
(1 row)
```

## ----- 10. c Funkcie -----

Funkcia mesačný sumár predajov s 2 parametrami – mesiac a rok.

```
sportovy_obchod=# create or replace function fnc_mesacny_sumar(integer, integer) returns varchar as
sportovy_obchod=# $$
sportovy_obchod$$# declare rk_suma decimal (10,2);
sportovy_obchod$$#           hlaska varchar;
sportovy_obchod$$# begin
sportovy_obchod$$#   hlaska='';
sportovy_obchod$$#   select sum(predajna_cena) into rk_suma from objednavky
sportovy_obchod$$#   where extract (year from datum)= $2 and extract(month from datum)=$1;
sportovy_obchod$$#   IF rk_suma ISNULL
sportovy_obchod$$#   THEN
sportovy_obchod$$#     hlaska= 'V mesiaci'||$1||' roku'||$2||' nie je ziadna suma.';
sportovy_obchod$$#   ELSE
sportovy_obchod$$#     hlaska= 'V mesiaci'||$1||' roku'||$2||' je suma penazi: '||rk_suma||';
sportovy_obchod$$#   END IF;
sportovy_obchod$$#   RETURN hlaska;
sportovy_obchod$$# end;
sportovy_obchod$$#
sportovy_obchod# language plpgsql;
CREATE FUNCTION
```

Funkcia sa zavolá cez select:

```
sportovy_obchod=# select fnc_mesacny_sumar(01,2020);
fnc_mesacny_sumar
-----
V mesiaci 1 roku 2020 je suma penazi: 586.00,
(1 row)

sportovy_obchod=# select fnc_mesacny_sumar(02,2020);
fnc_mesacny_sumar
-----
V mesiaci 2 roku 2020 je suma penazi: 880.00,
(1 row)

sportovy_obchod=# select fnc_mesacny_sumar(03,2020),
fnc_mesacny_sumar
-----
V mesiaci 3 roku 2020 je suma penazi: 484.00,
(1 row)

sportovy_obchod=# select fnc_mesacny_sumar(04,2020);
fnc_mesacny_sumar
-----
V mesiaci 4 roku 2020 je suma penazi: 2739.00,
(1 row)

sportovy_obchod=# select fnc_mesacny_sumar(05,2020);
fnc_mesacny_sumar
-----
V mesiaci 5 roku 2020 je suma penazi: 160.00,
(1 row)
```

Funkcia mesačný sumár s 2 parametrami – mesiac a rok, na zistenie prímov.

```
sportovy_obchod=# create or replace function fnc_sumar(integer, integer) returns varchar as
sportovy_obchod# $$
sportovy_obchod$# declare mesiac alias for $1;
sportovy_obchod$#           rok alias for $2;
sportovy_obchod$# suma_plus decimal (10,2);
sportovy_obchod$# suma_minus decimal (10,2);
sportovy_obchod$#           hlaska varchar;
sportovy_obchod$# begin
sportovy_obchod$$#   hlaska='';
sportovy_obchod$$#   select sum(predajna_cena) into suma_plus from objednavky
sportovy_obchod$$#   inner join zakaznici on (zakaznici.id=objednavky.id_zakaznici)
sportovy_obchod$$#   where extract (month from datum)= $1 and extract (year from datum)= $2 and predajna_cena>0;
sportovy_obchod$$#   IF suma_plus ISNULL then
sportovy_obchod$$#     suma_plus:=0;
sportovy_obchod$$#   END IF;
sportovy_obchod$$#   select sum(predajna_cena) into suma_minus from objednavky
sportovy_obchod$$#   inner join zakaznici on (zakaznici.id=objednavky.id_zakaznici)
sportovy_obchod$$#   where extract (month from datum)= $1 and extract (year from datum)= $2 and predajna_cena<0;
sportovy_obchod$$#   IF suma_minus ISNULL THEN
sportovy_obchod$$#     suma_minus:=0;
sportovy_obchod$$#   END IF;
sportovy_obchod$$#   hlaska= 'Mesacny sumar za'||$1||'/'||$2||' je obchod v pluse '||suma_plus||' v minuse '||suma_minus|| '.';
sportovy_obchod$$#   RETURN hlaska;
sportovy_obchod$$# end;
sportovy_obchod$$#
sportovy_obchod# language plpgsql;
CREATE FUNCTION
```

Funkcia sa zavolá cez select:

```
sportovy_obchod=# select * from fnc_sumar(01,2020);
fnc_sumar
-----
Mesacny sumar za1/2020 je obchod v pluse 586.00 v minuse 0.00.
(1 row)

sportovy_obchod=# select * from fnc_sumar(03,2020);
fnc_sumar
-----
Mesacny sumar za3/2020 je obchod v pluse 484.00 v minuse 0.00.
(1 row)

sportovy_obchod=# select * from fnc_sumar(07,2020);
fnc_sumar
-----
Mesacny sumar za7/2020 je obchod v pluse 0.00 v minuse 0.00.
(1 row)
```

## ----- 10. d Triggle -----

```
sportovy_obchod=# drop function fnc_zakaznici;
DROP FUNCTION
sportovy_obchod=# create or replace function fnc_zakaznici()
sportovy_obchod# returns trigger as '
sportovy_obchod# declare
sportovy_obchod'#   pocet integer;
sportovy_obchod#   zaznam record;
sportovy_obchod'# begin
sportovy_obchod'# if new.email is null then
sportovy_obchod'#   raise exception ''Email musi byt zadany.'';
sportovy_obchod'# end if;
sportovy_obchod'# select into pocet count(*) from zakaznici where email=new.email;
sportovy_obchod'# if (pocet>0) then
sportovy_obchod#   raise exception ''Lutujeme, zakaznik s touto e-mailovou adresou uz existuje'';
sportovy_obchod'# end if;
sportovy_obchod# RETURN new;
sportovy_obchod'# end;
'sportovy_obchod# language plpgsql;
CREATE FUNCTION
sportovy_obchod=# create trigger trigger_zakaznici before insert or update on zakaznici for each row execute procedure fnc_zakaznici();
CREATE TRIGGER
sportovy_obchod=# insert into zakaznici values ('19', 'Alena', 'Dokonala', '0908765555',null,'1','1 Maja','139');
ERROR: Email musi byt zadany.
CONTEXT: PL/pgSQL function fnc_zakaznici() line 8 at RAISE
sportovy_obchod=# insert into zakaznici values ('19', 'Alena', 'Dokonala', '0908765555', 'alena.nahnevana@gmail.com','1','1 Maja','139');
ERROR: Lutujeme, zakaznik s touto e-mailovou adresou uz existuje
CONTEXT: PL/pgSQL function fnc_zakaznici() line 12 at RAISE
sportovy_obchod=#

```

## ----- 11. Otázky a odpovede k projektu -----

1. Čo je to primárny a unikátny klúč v tabuľkách, uveďte po 2 príklady z vašej databázy.

**Primárny klúč** je v relačne jedinečný atribút alebo skupina atribútov, ktorý jednoznačne identifikuje každý záznam v tabuľke. Žiadne dva záznamy v konzistentne navrhnutej tabuľke nemajú zhodný primárny klúč. V tabuľke môže byť viacero atribútov, ktoré jednoznačne identifikujú záznam, pričom jeden z nich (alebo aj iná kombinácia atribútov) sa môže použiť ako primárny klúč.

**Unikátny klúč** je taký klúč, ktorého hodnoty nadobúdajú len jedinečné hodnoty

Primárny klúč: id (vo všetkých tabuľkách)

Unikátny klúč: alter table zakaznici add constraint email\_unique unique (email);

2. Vymenujte aspoň 5 príkazov jazyka SQL, ktoré ste použili vo vašom projekte.

```

CREATE DATABASE sportovy_obchod
CREATE TABLE mesto
INSERT INTO mesto (id, nazov, PSC) VALUES (nextval('mesto_id_seq'), 'Zilina 1', '01001');
SELECT * from platby;
ALTER TABLE mesto add column kraj varchar (50);
CREATE GROUP spravca;
REVOKE DELETE ON mesto FROM GROUP zamestnanci;

```

3. Uveďte po 2 príklady na množinové operácie restrict, projection a join s tabuľkami vo vašom projekte.

#### RESTRICT

- 1) select \* from objednavky limit 7 offset 10;
- 2) select \* from zakaznici where meno like 'J%';

#### PROJECTION

- 1) select druh\_tovaru, nazov, pocet\_kusov, datum from tovar left outer join objednavky on (objednavky.id\_tovar=tovar.id) where extract (year from datum) = 2020 and extract (month from datum) =2;

```

sportovy_obchod=# select druh_tovaru, nazov, pocet_kusov, datum from tovar left outer join objednavky on (objednavky.id_tovar=tovar.id) where extract (year from datum) = 2020 and extract (month from datum) =2;
druh_tovaru | nazov | pocet_kusov | datum
-----+-----+-----+-----
letne sporty | badmintonova sada | 1 | 2020-02-05
letne sporty | badmintonova siet | 1 | 2020-02-05
letne sporty | badmintonovy kosik | 2 | 2020-02-05
zimne sporty | prilba na ladowy hokej | 1 | 2020-02-17
zimne sporty | lyze | 1 | 2020-02-23
obuv | kopacky | 1 | 2020-02-23
letne sporty | futbalova lopta | 1 | 2020-02-23
obuv | tenisky | 1 | 2020-02-23
obuv | trettry | 1 | 2020-02-23
letne sporty | skateboard | 1 | 2020-02-23
(10 rows)

```

- 2) select druh\_tovaru, nazov, pocet\_kusov, datum from tovar left outer join objednavky on (objednavky.id\_tovar=tovar.id) where extract (year from datum) = 2020 and extract (month from datum) = 2 and extract (day from datum) between 12 and 28;

```

sportovy_obchod=# select druh_tovaru, nazov, pocet_kusov, datum from tovar left outer join objednavky on (objednavky.id_tovar=tovar.id) where extract (year from datum) = 2020 and extract (month from datum) = 2 and extract (day from datum) between 12 and 28;
druh_tovaru | nazov | pocet_kusov | datum
-----+-----+-----+-----
zimne sporty | prilba na ladowy hokej | 1 | 2020-02-17
zimne sporty | lyze | 1 | 2020-02-23
obuv | kopacky | 1 | 2020-02-23
letne sporty | futbalova lopta | 1 | 2020-02-23
obuv | tenisky | 1 | 2020-02-23
obuv | trettry | 1 | 2020-02-23
letne sporty | skateboard | 1 | 2020-02-23
(7 rows)

```

#### JOIN

- 1) select druh\_tovaru, nazov, count (\*), sum (cena) from tovar left outer join objednavky on (tovar.id=objednavky.id\_tovar) group by druh\_tovaru, tovar.nazov;
- 2) select datum, priezvisko from objednavky left outer join zakaznici on (zakaznici.id=objednavky.id\_zakaznici) order by datum, priezvisko;

4. Práva na aké operácie sa pridelujú a odoberajú užívateľom v jazyku SQL, uvedťte 3 príklady z vášho projektu.

Práva sa udeľujú pomocou príkazov: select, insert, delete, update.

Príklady:

GRANT ALL ON mesto, zakaznici, tovar, objednavky TO GROUP spravca;

GRANT SELECT, INSERT ON zakaznici, tovar TO GROUP zamestnanci;

GRANT ALL ON mesto, zakaznici, tovar, objednavky TO GROUP zamestnanci;

REVOKE DELETE ON mesto FROM GROUP zamestnanci;

5. Čo je to modifikácia štruktúry tabuľky, uvedťte 2 príklady z vášho projektu.

Modifikácia štruktúry tabuľky znamená zmena tabuľky, napríklad: premenovanie tabuľky, stĺpca; pridanie alebo odobranie stĺpcov; zmena typu stĺpca; zmena údajov v stĺpci, pridanie integritných obmedzení.

Príklady:

alter table mesto add column kraj varchar (50);

alter table mesto alter column PSC set not null;

6. Aký je rozdiel medzi pohľadom view a temporárnom tabuľkom? Uveďte príklad 2 pohľadov z vášho projektu.

**Pohľad:** je dynamické okno do databázy. Má svoje pomenovanie, navonok sa javí ako tabuľka.

Príklad:

create view platby as select meno, priezvisko, sum(cena) from tovar left outer join zakaznici on (zakaznici.id=tovar.id) group by meno, priezvisko order by meno;

```
sportovy_obchod=# create view platby as select meno, priezvisko, sum(cena) from tovar left outer join zakaznici on (zakaznici.id=tovar.id) group by meno, priezvisko order by meno;
CREATE VIEW
sportovy_obchod=# select meno, priezvisko, sum(cena) from tovar left outer join zakaznici on (zakaznici.id=tovar.id) group by meno, priezvisko order by meno;
+-----+-----+-----+
| meno | priezvisko | sum   |
+-----+-----+-----+
| Alena | Vesela    | 45.00
| Alena | Krasna    | 125.00
| Alena | Nahnevana | 17.00
| Aneta | Drza      | 14.00
| Anna  | Kratka    | 355.00
| Jan   | Mily       | 117.00
| Jana  | Dobra     | 67.00
| Jana  | Mudra     | 77.00
| Juraj | Smutny    | 30.00
| Juraj | Nizky     | 77.00
| Lubomir | Mudry    | 125.00
| Milan | Dobry     | 125.00
| Natalia | Vesela   | 3.00
| Olina | Kratka    | 33.00
| Peter | Rovny     | 99.00
| Tibor | Nahly    | 5.00
| Zuzana | Nudna    | 22.00
                                         | 2892.00
(18 rows)
```

create view prijem\_za\_rok as select extract (month from datum) as mesiac, extract (year from datum) as rok, sum (predajna\_cena) from objednavky where predajna\_cena>0 group by 1,2 union select extract (month from datum) as mesiac, extract (year from datum) as rok, sum (predajna\_cena) from objednavky where predajna\_cena<0 group by 1,2 order by 1,2,3 desc;

```
sportovy_obchod=# create view prijem_za_rok as select extract (month from datum) as mesiac, extract (year from datum) as rok, sum (predajna_cena) from objednavky where predajna_cena>0 group by 1,2 union select extract (month from datum) as mesiac, extract (year from datum) as rok, sum (predajna_cena) from objednavky where predajna_cena<0 group by 1,2 order by 1,2,3 desc;
CREATE VIEW
sportovy_obchod=# select extract (month from datum) as mesiac, extract (year from datum) as rok, sum (predajna_cena) from objednavky where predajna_cena>0 group by 1,2 union select extract (month from datum) as mesiac, extract (year from datum) as rok, sum (predajna_cena) from objednavky where predajna_cena<0 group by 1,2 order by 1,2,3 desc;
+-----+-----+-----+
| mesiac | rok | sum   |
+-----+-----+-----+
| 1     | 2020 | 508.00
| 2     | 2020 | 760.00
| 3     | 2020 | 484.00
| 4     | 2020 | 2739.00
| 5     | 2020 | 160.00
(5 rows)
```

**Temporárna tabuľka:** tvorí sa pomocou príkazu select. Okzrkadluje iba stav po svojom vytvorení, pohľad doň zahrňa aj všetky zmeny od vytvorenia.

Priklad:

```
select meno, priezvisko, sum(cena) into temp tmp_zakaznici from tovar left outer join zakaznici  
on (zakaznici.id=tovar.id) group by meno, priezvisko order by meno;
```

7. Aké agregačné funkcie pozná jazyk SQL? Napíšte 3 príklady agregačnej funkcie z vášho projektu.

Agregačné funkcie poznáme: count, avg, max, min, sum.

Priklady:

```
select sum(pocet_kusov) from objednavky;
```

```
sportovy_obchod=# select sum(pocet_kusov) from objednavky;  
sum  
-----  
53  
(1 row)
```

```
select max(predajna_cena) from objednavky;
```

```
sportovy_obchod=# select max(predajna_cena) from objednavky;  
max  
-----  
1300.00  
(1 row)
```

```
select avg(pocet_kusov*predajna_cena)::numeric(10,2) as priemerna_cena from objednavky  
where extract (year from datum)=2020 and extract (month from datum) between 1 and 5;
```

```
sportovy_obchod=# select avg(pocet_kusov*predajna_cena)::numeric(10,2) as priemerna_cena from objednavky where extract (year from datum)=2020 and extract (month from datum)between 1 and 5;  
priemerna_cena  
-----  
134.62  
(1 row)
```

8. Čo je to rules a index a na čo sa používajú? Uveďte po 1 príklade z vášho projektu.

**Rules:** umožňujú vykonávať akcie pri prístupe ku tabuľke, takto možno modifikovať účinky dotazov select, update, delete, existujú 2 typy pravidiel: DO – vykonáva príkazy SQL aj predložený dotaz, DO INSTEAD – nahradí dotaz užívateľa akciou pravidla, nothing je kľúčové slovo, ktoré nerobí nič.

Priklad:

```
create rule mesto_rule as on delete to mesto do instead nothing;
```

**Index:** je samostatný súbor, ktorý sa zoradí podľa jedného alebo viacerých stĺpcov. Obsahuje ukazovatele na súbor, kde je uložená tabuľka a tejto umožní rýchly prístup k špecifickým hodnotám.

Priklad:

```
create index zakaznici_idx on zakaznici (priezvisko, meno);
```

9. Aké sú etapy budovania dátového skladu? Opíšte pomocou 5 viet.

**Metóda ‘veľkého tresku‘** (vytvorenie dátového skladu implementáciu jedného projektu)

Fázy:

- analýza požiadaviek zo strany podniku
- vytvorenie dátového skladu
- vytvorenie prístupu buď priamo, alebo cez dátové trhy

**Prírastková metóda** (vytvorenie dátového skladu po etapách, vytvorenie dátových trhov, postupné dopĺňanie projektu, interaktívny proces)

Fázy:

- stratégia
- definícia
- analýza
- návrh
- zostavenie
- produkcia