

Produkce animované epizody

Dokumentace semestrální práce pro předmět 4IT218 Databáze

Kevin Nguyen

ZS 2023/2024

Uživatelské jméno: nguk08

Obsah

1	Popis zvolené výseče světa – zadání	3
2	Konceptuální schéma reality	4
3	Konceptuální datový model	5
4	Dokumentace databáze.....	6
4.1	Fyzický datový model	6
4.2	Definice relačních tabulek a souvisejících objektů	7
4.3	Integritní omezení	11
4.4	Definice přístupových práv.....	15
4.5	Definice dalších databázových objektů	16
5	Obsah databáze	16
5.1	SQL příkazy pro naplnění databáze daty	17
5.2	Opis vložených dat.....	18

1 Popis zvolené výšeče světa – zadání

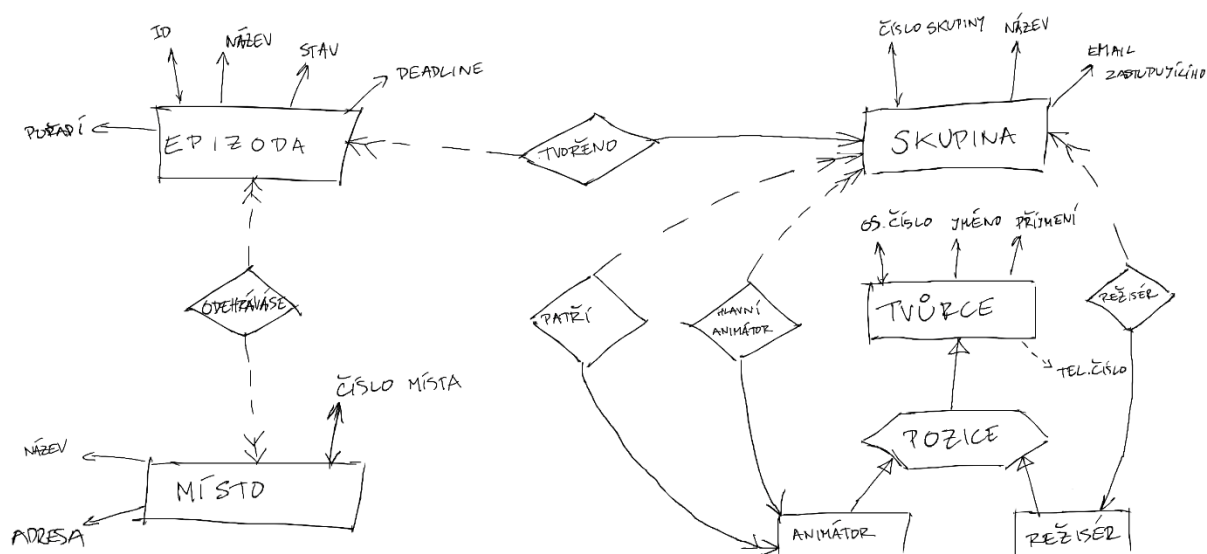
Společnost se rozhodla vyprodukovat cca 12 epizod, každá epizoda musí být tvořena právě jednou skupinou. Ke každé epizodě je nutno evidovat ID (k přiřazení dané skupině), pořadí, název epizody, stav, zdali je epizoda dokončena či nikoliv a datum dokončení neboli deadline. Epizody musí být dokončeny nejpozději do konce roku 2023, a proto deadline, tak nesmí překročit 31.12.2023.

Každá epizoda se může odehrávat ve více místech a na místě se může odehrávat více epizod. K jednotlivým míst, kde se epizody odehrávají, je nutno zapsat číslo (k rozlišení jednotlivých míst), název a adresa místa, odkud se daná epizoda odehrává.

Tvůrci pracující ve společnosti jsou zaevidovány osobní číslem (pro přiřazení do určité skupiny), jménem a příjmením. K tvůrci musí být možnost zapsat volitelně jedno telefonní číslo. Každý tvůrce je rozdělen do dvou pozic a to, buď jako animátor anebo jako režisér. Animátor může být ve více skupinách a v jedny skupině se může být hlavním animátorem a v další skupině či dalších skupinách jako běžný animátor.

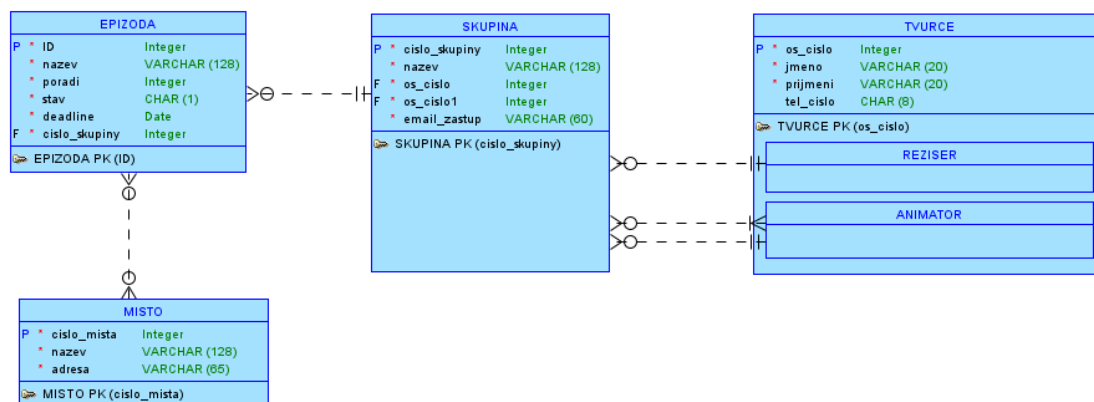
Skupina může tvořit více epizod anebo nemusí tvořit žádnou. Každá skupina je nutná být zaevidována číslem (k přiřazení epizody či epizod), názvem a emailem zastupujícího dané skupiny ke komunikaci s danou skupinou. V každé skupině musí být právě jeden hlavní animátor a režisér, animátorů musí být alespoň 1.

2 Konceptuální schéma reality



Obrázek 1: Produkce animované epizody – konceptuální schéma reality, zdroj: Kevin Nguyen (autor)

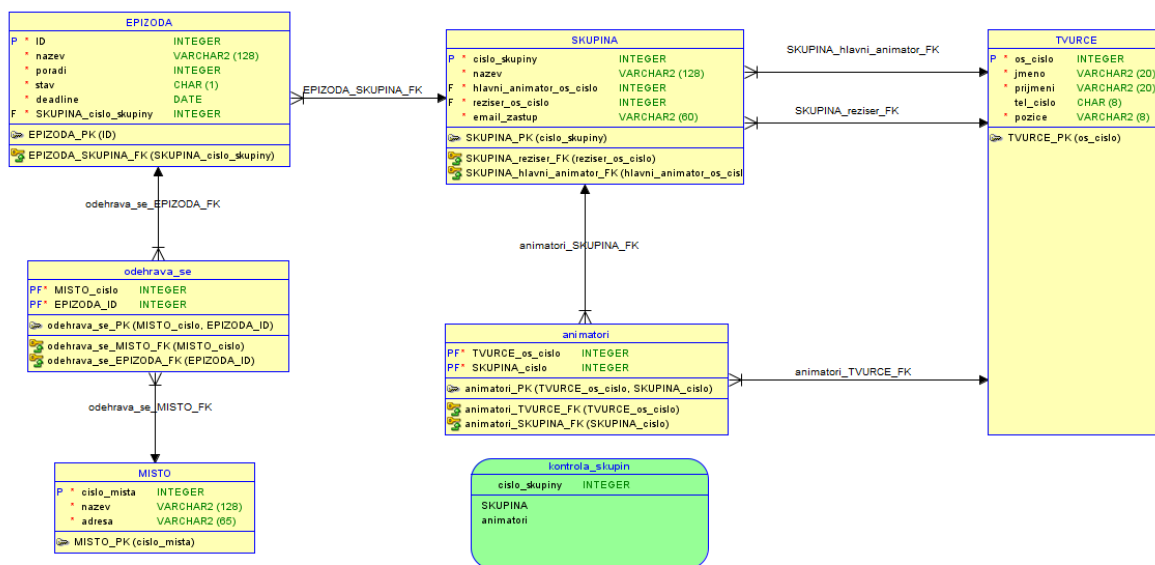
3 Konceptuální datový model



Obrázek 2: Produkce animované epizody – konceptuální datový model, zdroj: Kevin Nguyen (autor)

4 Dokumentace databáze

4.1 Fyzický datový model



Obrázek 3: Produkce animované epizody – fyzický datový model, zdroj: Kevin Nguyen (autor)

4.2 Definice relačních tabulek a souvisejících objektů

```
/*=====*/  
/*      Table: epizoda                                */  
/*=====*/
```

```
CREATE TABLE epizoda (  
    id            INTEGER NOT NULL,  
    nazev         VARCHAR2(128) NOT NULL,  
    poradi        INTEGER NOT NULL,  
    stav          CHAR(1) DEFAULT '0' NOT NULL,  
    deadline      DATE DEFAULT SYSDATE NOT NULL,  
    skupina_cislo_skupiny INTEGER NOT NULL  
);
```

```
ALTER TABLE epizoda  
    ADD CHECK ( stav IN ( '0', '1' ) );
```

```
ALTER TABLE epizoda ADD CONSTRAINT datum_deadline CHECK ( deadline <= TO_DATE ( '31.12.2023', 'DD.MM.YYYY' ) );
```

```
ALTER TABLE epizoda ADD CONSTRAINT epizoda_pk PRIMARY KEY ( id );
```

```
GRANT select ON epizoda TO STUDENT;
```

```
GRANT delete, insert, select, update ON epizoda TO IT218;
```

```
/*=====*/  
/*      Table: misto                                */  
/*=====*/
```

```
CREATE TABLE misto (  
    cislo_mista INTEGER NOT NULL,  
    nazev       VARCHAR2(128) NOT NULL,  
    adresa      VARCHAR2(65) NOT NULL  
);
```

```
ALTER TABLE misto ADD CONSTRAINT misto_pk PRIMARY KEY ( cislo_mista );
```

```
GRANT select ON misto TO STUDENT;
```

```
GRANT delete, insert, select, update ON misto TO IT218;
```

```
/*=====*/  
/*      Table: animatori      */  
/*=====*/
```

```
CREATE TABLE animatori (  
    tvurce_os_cislo INTEGER NOT NULL,  
    skupina_cislo  INTEGER NOT NULL  
);
```

```
ALTER TABLE animatori ADD CONSTRAINT animatori_pk PRIMARY KEY ( tvurce_os_cislo,  
skupina_cislo );
```

```
GRANT select ON animatori TO STUDENT;
```

```
GRANT delete, insert, select, update ON animatori TO IT218;
```

```
/*=====*/  
/*      Table: odehrava_se      */  
/*=====*/
```

```
CREATE TABLE odehrava_se (  
    misto_cislo INTEGER NOT NULL,  
    epizoda_id  INTEGER NOT NULL  
);
```

```
ALTER TABLE odehrava_se ADD CONSTRAINT odehrava_se_pk PRIMARY KEY ( misto_cislo,  
epizoda_id );
```

```
GRANT select ON odehrava_se TO STUDENT;
```

```
GRANT delete, insert, select, update ON odehrava_se TO IT218;
```



```
/*=====*/  
/*      Table: skupina                                     */  
/*=====*/
```

```
CREATE TABLE skupina (  
    cislo_skupiny    INTEGER NOT NULL,  
    nazev            VARCHAR2(128) NOT NULL,  
    reziser_os_cislo INTEGER NOT NULL,  
    hlavni_animator_os_cislo INTEGER NOT NULL,  
    email_zastup     VARCHAR2(60) NOT NULL  
);
```

```
ALTER TABLE skupina ADD CONSTRAINT skupina_pk PRIMARY KEY ( cislo_skupiny );
```

```
GRANT select ON skupina TO STUDENT;
```

```
GRANT delete, insert, select, update ON skupina TO IT218;
```

```
/*=====*/  
/*      Table: tvurce                                     */  
/*=====*/
```

```
CREATE TABLE tvurce (  
    os_cislo INTEGER NOT NULL,  
    jmeno    VARCHAR2(20) NOT NULL,  
    prijmeni VARCHAR2(20) NOT NULL,  
    tel_cislo CHAR(9),  
    pozice   VARCHAR2(8) NOT NULL  
);
```

```
ALTER TABLE tvurce  
    ADD CONSTRAINT ch_inh_tvurce CHECK ( pozice IN ( 'ANIMATOR', 'REZISER' ) );
```

```
ALTER TABLE tvurce ADD CONSTRAINT tvurce_pk PRIMARY KEY ( os_cislo );
```

```
GRANT select ON tvurce TO STUDENT;
```

```
GRANT delete, insert, select, update ON tvurce TO IT218;
```

```
/*=====*/
/*      View: kontrola_skupin                                */
/*=====*/
CREATE OR REPLACE VIEW kontrola_skupin ( cislo_skupiny ) AS
SELECT
    skupina.cislo_skupiny
FROM
    skupina
    LEFT JOIN animatori ON skupina.cislo_skupiny = animatori.skupina_cislo
WHERE
    animatori.tvurce_os_cislo IS NULL
;

GRANT select ON kontrola_skupin TO STUDENT;

GRANT select ON kontrola_skupin TO IT218;

ALTER TABLE animatori
    ADD CONSTRAINT animatori_skupina_fk FOREIGN KEY ( skupina_cislo )
        REFERENCES skupina ( cislo_skupiny );

ALTER TABLE animatori
    ADD CONSTRAINT animatori_tvurce_fk FOREIGN KEY ( tvurce_os_cislo )
        REFERENCES tvurce ( os_cislo );

ALTER TABLE epizoda
    ADD CONSTRAINT epizoda_skupina_fk FOREIGN KEY ( skupina_cislo_skupiny )
        REFERENCES skupina ( cislo_skupiny );

ALTER TABLE odehrava_se
    ADD CONSTRAINT odehrava_se_epizoda_fk FOREIGN KEY ( epizoda_id )
        REFERENCES epizoda ( id );

ALTER TABLE odehrava_se
    ADD CONSTRAINT odehrava_se_misto_fk FOREIGN KEY ( misto_cislo )
        REFERENCES misto ( cislo_mista );

ALTER TABLE skupina
```

```
ADD CONSTRAINT skupina_hlavni_animator_fk FOREIGN KEY ( hlavni_animator_os_cislo )  
REFERENCES tvurce ( os_cislo );
```

```
ALTER TABLE skupina
```

```
ADD CONSTRAINT skupina_reziser_fk FOREIGN KEY ( reziser_os_cislo )  
REFERENCES tvurce ( os_cislo );
```

4.3 Integritní omezení

Tabulka EPIZODA

Entitní integrita

Atributy tvořící primární klíč: ID

SQL kód pro definici primárního klíče:

```
ALTER TABLE epizoda ADD CONSTRAINT epizoda_pk PRIMARY KEY ( id );
```

Pozn.: Out-of-line definice omezení v rámci příkazu pro vytvoření tabulky EPIZODA

Doménová integrita

Sloupec STAV může nabývat pouze 0 (neniHotovo) a 1 (hotovo)

Popis omezení: Každá epizoda je buď dokončená (1 = hotovo) anebo je stále v produkci (0 = neniHotovo)

SQL kód příslušného omezení:

```
ALTER TABLE epizoda
```

```
ADD CHECK ( stav IN ( '0', '1' ) );
```

Pozn.: Inline specifikace omezení v rámci definice sloupce STAV, která je součástí příkazu pro vytvoření tabulky EPIZODA.

Sloupec DEADLINE může nabývat pouze data 31.12.2023 a dřívějšího

Popis omezení: Produkce animovaných epizod musí být dokončena nejpozději do konce roku 2023. Datum deadlinu nemůže tak nabývat datum s rokem 2024;

SQL kód příslušného omezení:

```
ALTER TABLE epizoda ADD CONSTRAINT datum_deadline CHECK ( deadline <= TO_DATE ( '31.12.2023', 'DD.MM.YYYY' ) );
```

Pozn.: Inline specifikace omezení v rámci definice sloupce DEADLINE, která je součástí příkazu pro vytvoření tabulky EPIZODA

Referenční integrita

Sloupec SKUPINA_cislo_skupiny v tabulce EPIZODA představuje cizí klíč

Popis omezení: Každá epizoda má svoji skupinu. Daná skupina je evidována v tabulce SKUPINA. Sloupec SKUPINA_cislo_skupiny tedy obsahuje číslo dané skupiny.

Druh použitého řešení referenční integrity pro operaci DELETE: RESTRICT.

SQL kód příslušného omezení:

```
ALTER TABLE epizoda
```

```
ADD CONSTRAINT epizoda_skupina_fk FOREIGN KEY ( skupina_cislo_skupiny )  
REFERENCES skupina ( cislo_skupiny );
```

Tabulka SKUPINA

Entitní integrita

Atributy tvořící primární klíč: cislo_skupiny

SQL kód pro definici primárního klíče:

```
ALTER TABLE skupina ADD CONSTRAINT skupina_pk PRIMARY KEY ( cislo_skupiny );
```

Pozn.: Out-of-line definice omezení v rámci příkazu pro vytvoření tabulky SKUPINA

Doménová integrita

V rámci tabulky SKUPINA nejsou definována žádná omezení, která by sloužila k zajištění doménové integrity.

Referenční integrita

Sloupec reziser_os_cislo v tabulce SKUPINA představuje cizí klíč

Popis omezení: V každé skupině evidujeme právě jednoho režiséra. Režisér je jedním z tvůrců evidovaných v tabulce TVURCE. Sloupec reziser_os_cislo tedy obsahuje os_cislo příslušného režiséra.

Druh použitého řešení referenční integrity pro operaci DELETE: RESTRICT.

SQL kód příslušného omezení:

```
ALTER TABLE skupina
```

```
ADD CONSTRAINT skupina_reziser_fk FOREIGN KEY ( reziser_os_cislo )  
REFERENCES tvurce ( os_cislo );
```

Sloupec hlavni_animator_os_cislo v tabulce SKUPINA představuje cizí klíč

Popis omezení: V každé skupině evidujeme právě jednoho hlavního animátora. Hlavní animátor je jedním z tvůrců evidovaných v tabulce TVURCE. Sloupec hlavni_animator_os_cislo tedy obsahuje os_cislo příslušného hlavního animátora.

Druh použitého řešení referenční integrity pro operaci DELETE: RESTRICT

SQL kód příslušného omezení:

```
ALTER TABLE skupina
```

```
ADD CONSTRAINT skupina_hlavni_animator_fk FOREIGN KEY ( hlavni_animator_os_cislo )  
REFERENCES tvurce ( os_cislo );
```

Tabulka TVURCE

Entitní integrita

Atributy tvořící primární klíč: os_cislo

SQL kód pro definici primárního klíče:

```
ALTER TABLE tvurce ADD CONSTRAINT tvurce_pk PRIMARY KEY ( os_cislo );
```

Pozn.: Out-of-line definice primárního klíče v rámci příkazu pro vytvoření tabulky TVURCE

Doménová integrita

Tvůrce je v pozici buď animátor nebo režisér

Popis omezení: Každý tvůrce má svoji pozici. Pozice je rozdělena na dvě, a to buď jako animátor nebo jako režisér.

SQL kód příslušného omezení:

```
ALTER TABLE tvurce
```

```
ADD CONSTRAINT ch_inh_tvurce CHECK ( pozice IN ( 'ANIMATOR', 'REZISER' ) );
```

Referenční integrita

V tabulce TVURCE žádný ze sloupců nepředstavuje cizí klíč.

Tabulka MISTO

Entitní integrita

Atributy tvořící primární klíč: cislo_mista

SQL kód pro definici primárního klíče:

```
ALTER TABLE misto ADD CONSTRAINT misto_pk PRIMARY KEY ( cislo_mista );
```

Pozn.: Out-of-line definice omezení v rámci příkazu pro vytvoření tabulky MISTO

Doménová integrita

V rámci tabulky MISTO nejsou definována žádná omezení, která by sloužila k zajištění doménové integrity

Referenční integrita

V tabulce MISTO žádný ze sloupců nepředstavuje cizí klíč.

Tabulka ODEHRAVA_SE

Entitní integrita

Atributy tvořící primární klíč: MISTO_cislo, EPIZODA_ID

SQL kód pro definici primárního klíče:

```
ALTER TABLE odehrava_se ADD CONSTRAINT odehrava_se_pk PRIMARY KEY ( misto_cislo, epizoda_id );
```

Pozn.: Out-of-line definice omezení v rámci příkazu pro vytvoření tabulky

Doménová integrita

V rámci tabulky ODEHRAVA_SE nejsou definována žádná omezení, která by sloužila k zajištění doménové integrity

Referenční integrita

Sloupec MISTO_cislo v tabulce ODEHRAVA_SE představuje cizí klíč

Popis omezení: Tabulka ODEHRAVA_SE je tabulkou zajišťující realizaci vztahu M:N mezi entitními množinami Epizoda a Místo. Záznamy v tabulce jsou tak tvořeny pouze dvojicemi cizích klíčů identifikujících související záznamy v tabulkách EPIZODA a MISTO. Sloupec MISTO_cislo v tabulce ODEHRAVA_SE tedy obsahuje číslo místa, ve které se epizoda odehrává.

Druh použitého řešení referenční integrity pro operaci DELETE: RESTRICT

SQL kód příslušného omezení:

```
ALTER TABLE odehrava_se  
  
ADD CONSTRAINT odehrava_se_misto_fk FOREIGN KEY ( misto_cislo )  
  
REFERENCES misto ( cislo_mista );
```

Sloupec EPIZODA_ID v tabulce ODEHRAVA_SE představuje cizí klíč

Popis omezení: Tabulka ODEHRAVA_SE je tabulkou zajišťující realizaci vztahu M:N mezi entitními množinami Epizoda a Místo. Záznamy v tabulce jsou tak tvořeny pouze dvojicemi cizích klíčů identifikujících související záznamy v tabulkách EPIZODA a MÍSTO. Sloupec EPIZODA_ID v tabulce ODEHRAVA_SE tedy obsahuje id epizody, která se odehrává na určeném místě.

Druh použitého řešení referenční integrity pro operaci DELETE: RESTRICT

SQL kód příslušného omezení:

```
ALTER TABLE odehrava_se  
  
ADD CONSTRAINT odehrava_se_epizoda_fk FOREIGN KEY ( epizoda_id )  
  
REFERENCES epizoda ( id );
```

Tabulka ANIMATORI

Entitní integrita

Atributy tvořící primární klíč: TVURCE_os_cislo, SKUPINA_cislo

SQL kód pro definici primárního klíče:

```
ALTER TABLE animatori ADD CONSTRAINT animatori_pk PRIMARY KEY ( tvurce_os_cislo, skupina_cislo );
```

Pozn.: Out-of-line definice omezení v rámci příkazu pro vytvoření tabulky

Doménová integrita

V rámci tabulky ANIMATORI nejsou definována žádná omezení, která by sloužila k zajištění doménové integrity

Referenční integrita

Sloupec TVURCE_os_cislo v tabulce ANIMATORI představuje cizí klíč

Popis omezení: Tabulka ANIMATORI je tabulkou zajišťující realizaci vztahu M:N mezi entitními množinami Skupina a Tvůrce. Záznamy v tabulce jsou tak tvořeny pouze dvojicemi cizích klíčů identifikujících související záznamy v tabulkách SKUPINA a TVURCE. Sloupec TVURCE_os_cislo v tabulce ANIMATOR tedy obsahuje osobní číslo tvůrce, který pracuje v určené skupině.

Druh použitého řešení referenční integrity pro operaci DELETE: RESTRICT

SQL kód příslušného omezení:

```
ALTER TABLE animatori
```

```
ADD CONSTRAINT animatori_tvurce_fk FOREIGN KEY ( tvurce_os_cislo )
```

```
REFERENCES tvurce ( os_cislo );
```

Sloupec SKUPINA_cislo v tabulce ANIMATORI představuje cizí klíč

Popis omezení: Tabulka ANIMATORI je tabulkou zajišťující realizaci vztahu M:N mezi entitními množinami Skupina a Tvůrce. Záznamy v tabulce jsou tak tvořeny pouze dvojicemi cizích klíčů identifikujících související záznamy v tabulkách SKUPINA a TVURCE. Sloupec SKUPINA_cislo v tabulce ANIMATORI tedy obsahuje číslo dané skupiny, ve kterém pracuje daný tvůrce.

Druh použitého řešení referenční integrity pro operaci DELETE: RESTRICT

SQL kód příslušného omezení:

```
ALTER TABLE animatori
```

```
ADD CONSTRAINT animatori_skupina_fk FOREIGN KEY ( skupina_cislo )
```

```
REFERENCES skupina ( cislo_skupiny );
```

4.4 Definice přístupových práv

Definice přístupových práv pro uživatele STUDENT

```
GRANT select ON epizoda TO STUDENT;
```

```
GRANT select ON misto TO STUDENT;
```

```
GRANT select ON animatori TO STUDENT;
```

```
GRANT select ON odehrava_se TO STUDENT;
```

```
GRANT select ON skupina TO STUDENT;
```

```
GRANT select ON tvurce TO STUDENT;
```

```
GRANT select ON kontrola_skupin TO STUDENT;
```

Definice přístupových práv pro uživatele IT218

```
GRANT delete, insert, select, update ON epizoda TO IT218;
GRANT delete, insert, select, update ON misto TO IT218;
GRANT delete, insert, select, update ON animatori TO IT218;
GRANT delete, insert, select, update ON odehrava_se TO IT218;
GRANT delete, insert, select, update ON skupina TO IT218;
GRANT delete, insert, select, update ON tvurce TO IT218;
GRANT select ON kontrola_skupin TO IT218;
```

4.5 Definice dalších databázových objektů

VIEW pro výpis skupin, které neobsahují žádného hlavního animátora anebo žádného režiséra

Účel databázového objektu: V každé skupině pracuje právě jeden hlavní animátor, režisér a musí být alespoň jeden animátor. VIEW kontrola_skupin umožňuje vypsát skupiny, v rámci, kterých nepracuje žádný animátor a dochází tak k porušení pravidla pro vytváření skupiny.

SQL kód příslušného objektu:

```
CREATE OR REPLACE VIEW kontrola_skupin ( cislo_skupiny ) AS
SELECT
    skupina.cislo_skupiny
FROM
    skupina
    LEFT JOIN animatori ON skupina.cislo_skupiny = animatori.skupina_cislo
WHERE
    animatori.tvurce_os_cislo IS NULL
;
```


5 Obsah databáze

5.1 SQL příkazy pro naplnění databáze daty

```
INSERT into TVURCE (os_cislo, jmeno, prijmeni, tel_cislo, pozice) VALUES (1, 'Louis', 'Gogh', 782668909, 'REZISER');
```

```
INSERT into TVURCE (os_cislo, jmeno, prijmeni, tel_cislo, pozice) VALUES (2, 'Marta', 'Silva', 751268925, 'REZISER');
```

```
INSERT into TVURCE (os_cislo, jmeno, prijmeni, tel_cislo, pozice) VALUES (3, 'Long', 'Michael', 721548968, 'ANIMATOR');
```

```
INSERT into TVURCE (os_cislo, jmeno, prijmeni, tel_cislo, pozice) VALUES (4, 'Fernando', 'Torres', 685789123, 'ANIMATOR');
```

```
INSERT into TVURCE (os_cislo, jmeno, prijmeni, pozice) VALUES (5, 'Filip', 'Inzaghi', 'ANIMATOR');
```

```
INSERT into TVURCE (os_cislo, jmeno, prijmeni, tel_cislo, pozice) VALUES (6, 'Didier', 'Drogba', 789558752, 'REZISER');
```

```
INSERT into SKUPINA (cislo_skupiny, nazev, reziser_os_cislo, hlavni_animator_os_cislo, email_zastup) VALUES (1, 'Chelsea', 1, 3, 'louis@gmail.com');
```

```
INSERT into SKUPINA (cislo_skupiny, nazev, reziser_os_cislo, hlavni_animator_os_cislo, email_zastup) VALUES (2, 'Milan', 2, 4, 'fernando@gmail.com');
```

```
INSERT into SKUPINA (cislo_skupiny, nazev, reziser_os_cislo, hlavni_animator_os_cislo, email_zastup) VALUES (3, 'Berlin', 6, 5, 'inzaghi@gmail.com');
```

```
INSERT into SKUPINA (cislo_skupiny, nazev, reziser_os_cislo, hlavni_animator_os_cislo, email_zastup) VALUES (4, 'Madrid', 2, 3, 'long@gmail.com');
```

```
INSERT into EPIZODA (id, nazev, poradi, stav, deadline, skupina_cislo_skupiny) VALUES (1, 'The Rickshank Rickdemption', 1, 0, TO_DATE('21.01.2023', 'DD.MM.YYYY'), 1);
```

```
INSERT into EPIZODA (id, nazev, poradi, stav, deadline, skupina_cislo_skupiny) VALUES (2, 'Rickmancing the Stone', 2, 1, TO_DATE('05.03.2023', 'DD.MM.YYYY'), 2);
```

```
INSERT into EPIZODA (id, nazev, poradi, stav, deadline, skupina_cislo_skupiny) VALUES (3, 'Pickle Rick', 3, 0, TO_DATE('14.05.2023', 'DD.MM.YYYY'), 3);
```

```
INSERT into MISTO (cislo_mista, nazev, adresa) VALUES (1, 'Nickelodeon', '1515 Broadway, New York, NY 10036, Spojené státy americké');
```

```
INSERT into MISTO (cislo_mista, nazev, adresa) VALUES (2, 'Prima COOL', 'Vinohradská 3217/167, Praha, 100 00 Strašnice, Česká republika');
```

```
INSERT into MISTO (cislo_mista, nazev, adresa) VALUES (3, 'Barrandov TV', 'Kříženeckého náměstí 322, Praha, 152 00, Česká republika');
```

```
INSERT into ODEHRAVA_SE (MISTO_cislo, EPIZODA_id) VALUES (1,3);
```

```
INSERT into ODEHRAVA_SE (MISTO_cislo, EPIZODA_id) VALUES (1,1);
```

```
INSERT into ODEHRAVA_SE (MISTO_cislo, EPIZODA_id) VALUES (2,1);
```

```
INSERT into ODEHRAVA_SE (MISTO_cislo, EPIZODA_id) VALUES (3,3);
```

```

INSERT into ODEHRAVA_SE (MISTO_cislo, EPIZODA_id) VALUES (1,2);
INSERT into ANIMATORI (TVURCE_OS_CISLO, SKUPINA_CISLO) VALUES (4,1);
INSERT into ANIMATORI (TVURCE_OS_CISLO, SKUPINA_CISLO) VALUES (5,1);
INSERT into ANIMATORI (TVURCE_OS_CISLO, SKUPINA_CISLO) VALUES (3,2);
INSERT into ANIMATORI (TVURCE_OS_CISLO, SKUPINA_CISLO) VALUES (5,2);
INSERT into ANIMATORI (TVURCE_OS_CISLO, SKUPINA_CISLO) VALUES (3,3);
INSERT into ANIMATORI (TVURCE_OS_CISLO, SKUPINA_CISLO) VALUES (4,3);

```

5.2 Opis vložených dat

V této části uveďte opis dat uložených ve Vámi vytvořené databázi. Je třeba, aby zde byl uveden opis dat všech relačních tabulek ve Vaší databázi. Pro každou relační tabulku uveďte samostatnou podkapitolu. V této podkapitole pak uveďte tabulku, která bude svojí strukturou a obsahem odpovídat příslušné relační tabulce.

Tabulka EPIZODA

ID	NAZEV	PORADI	STAV	DEADLINE	SKUPINA_CISLO_SKUPINY
1	The Rickshank Rickdemption	1	0	21.01.23	1
2	Rickmancing the Stone	2	1	05.03.23	2
3	Pickle Rick	3	0	14.05.23	3

Tabulka MISTO

CISLO_MISTA	NAZEV	ADRESA
1	Nickelodeon	1515 Broadway, New York, NY 10036, Spojené státy americké
2	Prima COOL	Vinohradská 3217/167, Praha, 100 00 Strašnice, Česká republika
3	Barrandov TV	Kříženeckého náměstí 322, Praha, 152 00, Česká republika

Tabulka SKUPINA

CISLO_SKUPINY	NAZEV	REZISER_OS_CISLO	HLAVNI_ANIMATOR_OS_CISLO	EMAIL_ZASTUP
1	Chelsea	1	3	louis@gmail.com
2	Milan	2	4	fernando@gmail.com
3	Berlin	6	5	inzaghi@gmail.com
4	Madrid	2	3	long@gmail.com

Tabulka TVURCE

OS_CISLO	JMENO	PRIJMENI	TEL_CISLO	POZICE
1	Louis	Gogh	782668909	REZISER
2	Marta	Silva	751268925	REZISER
3	Long	Michael	721548968	ANIMATOR
4	Fernando	Torres	685789123	ANIMATOR
5	Filip	Inzaghi		ANIMATOR
6	Didier	Drogba	789558752	REZISER

Tabulka ODEHRAVA_SE

MISTO_CISLO	EPIZODA_ID
1	3
1	1
2	1
3	3
1	2

Tabulka ANIMATORI

TVURCE_OS_CISLO	SKUPINA_CISLO
4	1
5	1
3	2
5	2
3	3
4	3