

Databáze Domu dětí a mládeže

KIV/DB1 – Semestrální práce

Student:	Dominik Zappe
Osobní číslo:	A20B0279P
Email:	zapped99@students.zcu.cz
Datum:	14. 12. 2021

Obsah

1.	Formulář	1
2.	Schéma	2
3.	Zadání	3
4.	Tabulky	3
4.1.	Dítě	3
4.2.	Vedoucí	3
4.3.	Kroužek	4
4.4.	Místnost	4
4.5.	Rozvrhová akce	4
4.6.	Přihláška	5
5.	Pohledy	5
5.1.	Přihlášené děti na konkrétní rozvrhové akce	5
5.1.1.	SQL	5
5.1.2.	Tabulka výsledků	6
5.2.	Které kroužky vede konkrétní vedoucí	6
5.2.1.	SQL	6
5.2.2.	Tabulka výsledků	7
5.3.	Počet dětí z konkrétního města	7
5.3.1.	SQL	7
5.3.2.	Tabulka výsledků	7
6.	Scénáře	7
6.1.	Scénář 1	8
6.2.	Scénář 2	8
6.3.	Scénář 3	9
7.	Závěr	10

1. Formulář



Databázové systémy I

**zimní semestr
2021/2022**

Dominik Zappe

Orion login: zapped99

Osobní číslo: A20B0279P

Studuji	ZČU
Fakulta	FAV
Studijní program	Informatika a výpočetní technika
Studijní obor	Informatika
Rok studia	2.

Název práce: Databáze Domu dětí a mládeže

Základem každého DDM jsou kroužky a děti.

Na jeden kroužek je podáno určitě více než jedna přihláška. Každá přihláška je právě na jeden kroužek.

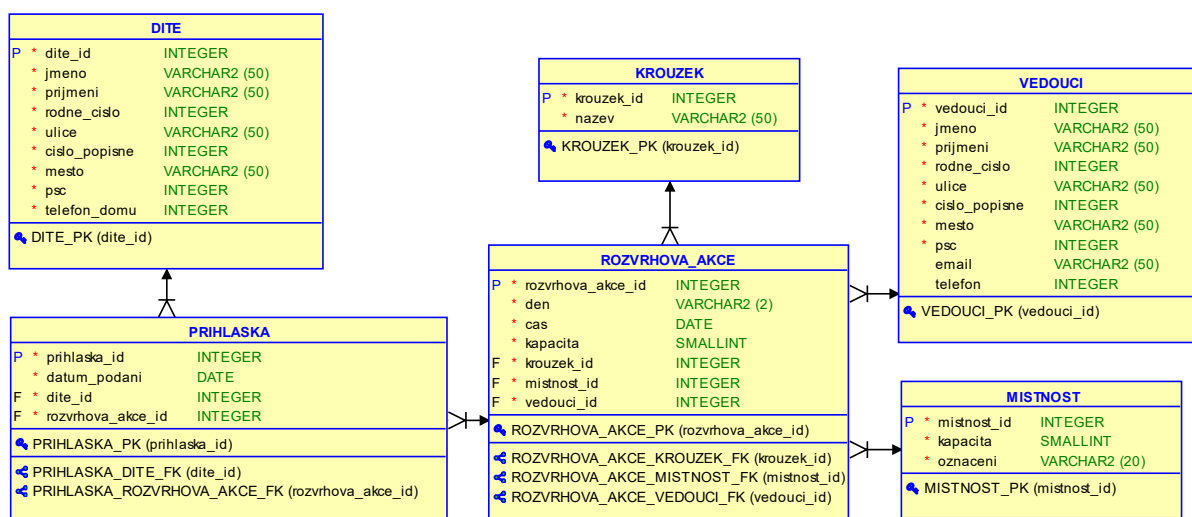
Jedno dítě může podat více přihlášek na různé kroužky. Jedna daná přihláška patří právě jednomu dítěti.

Každý kroužek je v právě jedné místnosti, ale v jedné místnosti se může za celý týden vystřídat i více kroužků.

Každý kroužek má jednoho vedoucího, ale jeden vedoucí může vést i více kroužků.

Zadání společné s předmětem KIV/WEB: Ne

2. Schéma



3. Zadání

Zadaná semestrální práce z předmětu KIV/DB1 má 3 části.

V první části jde o slovní návrh databáze. Jedná se o byznysový návrh, pouze sděluje, jaké entity budou později v databázi zavedeny a jak spolu souvisí – tedy popisuje relace mezi entitami.

Druhou částí je relační schéma. Tedy zobrazení entit s atributy a relacemi mezi nimi. Relační schéma je vymodelováno na základě byznysového popisu z prvního kroku.

Poslední třetí částí je samotná implementace databáze. Vychází se z předchozích bodů – tedy z modelu relačního schématu. Datový model musí obsahovat alespoň 5 tabulek. Dále musí splňovat 3. normální formu. V každé tabulce musí být alespoň 1 záznam, pokud tabulka obsahuje cizí klíč, tak alespoň k jednomu cizímu klíči musí existovat 2 různé záznamy. Databáze musí obsahovat alespoň 2 databázové pohledy, v každém musí být uplatněna operace projekce a selekce. Alespoň jeden z pohledů musí spojoval dvě či více tabulek. Dále budou navrženy alespoň dva scénáře. Scénář je SQL kód ověřující funkčnost navržených databázových pohledů. Scénáře mimo jiné také testují tzv. konzistenci databáze.

4. Tabulky

4.1. Dítě

Tabulka reprezentuje konkrétní děti, které jsou přihlášeny na nějaký kroužek. Dítě je popsáno následujícími atributy:

- **dite_id** – unikátní celé číslo, primární klíč
- **jmeno** – řetězec až padesáti znaků, nemůže být prázdný řetězec ani null; reprezentuje křestní jméno dítěte
- **prijmeni** – řetězec až padesáti znaků, nemůže být prázdný řetězec ani null; reprezentuje příjmení dítěte
- **rodne_cislo** – celočíselný typ, nemůže být null; číslo reprezentující rodné číslo dítěte (formát bez lomítka)
- **ulice** – řetězec až padesáti znaků, nemůže být null; reprezentuje název ulice trvalého bydliště dítěte
- **cislo_popisne** – celočíselný typ, nemůže být null; číslo reprezentující číslo popisné domu trvalého bydliště dítěte
- **mesto** – řetězec až padesáti znaků, nemůže být null; reprezentuje název obce, v které dítě bydlí
- **psc** – celočíselný typ, nemůže být null; číslo reprezentující poštovní směrovací číslo bydliště dítěte
- **telefon_domu** – celočíselný typ, nemůže být null, musí být devítimístné; číslo reprezentující telefonní číslo na jednoho ze zákonných zástupců

4.2. Vedoucí

Tabulka reprezentuje konkrétní vedoucí jednotlivých rozvrhových akcí. Vedoucí je popsán atributy:

- **vedouci_id** – unikátní celé číslo, primární klíč
- **jmeno** – řetězec až padesáti znaků, nemůže být prázdný řetězec ani null; reprezentuje křestní jméno vedoucího

- **prijmeni** – řetězec až padesáti znaků, nemůže být prázdný řetězec ani null; reprezentuje příjmení vedoucího
- **rodne_cislo** – celočíselný typ, nemůže být null; číslo reprezentující rodné číslo vedoucího (formát bez lomítka)
- **ulice** – řetězec až padesáti znaků, nemůže být null; reprezentuje název ulice trvalého bydliště vedoucího
- **cislo_popisne** – celočíselný typ, nemůže být null; číslo reprezentující číslo popisné domu trvalého bydliště vedoucího
- **mesto** – řetězec až padesáti znaků, nemůže být null; reprezentuje název obce, v které vedoucí bydlí
- **psc** – celočíselný typ, nemůže být null; číslo reprezentující poštovní směrovací číslo bydliště vedoucího
- **email** – řetězec až padesáti znaků, reprezentuje e-mailovou adresu vedoucího
- **telefon** – devítimístné celé číslo reprezentující telefonní kontakt na vedoucího

4.3. Kroužek

Tabulka reprezentuje obecný kroužek, který může být uveden ve více rozvrhových akcích. Tedy jeden kroužek může být veden vícero lidmi ve vícero místnostech v různých dnech a časech. Kroužek je popsán následujícími atributy:

- **krouzek_id** – unikátní celé číslo, primární klíč
- **nazev** – řetězec až padesáti znaků, nemůže být prázdný řetězec ani null; reprezentuje celý název kroužku

4.4. Místnost

Tabulka reprezentuje místnost, tedy místo, kde může nějaký kroužek v daném čase a dni probíhat. Místnost je popsána atributy:

- **mistnost_id** – unikátní celé číslo, primární klíč
- **kapacita** – celočíselný typ, nemůže být null; reprezentuje maximální počet lidí, kteří se do místnosti vejdou
- **oznaceni** – řetězec až dvaceti znaků, nemůže být prázdný řetězec ani null; reprezentuje označení dané místnosti

4.5. Rozvrhová akce

Tabulka reprezentuje rozvrhové akce – tedy konkrétní kroužky vyučované v konkrétní den a čas na konkrétním místě konkrétním vedoucím. Rozvrhová akce je popsána následujícími atributy:

- **rozvrhova_akce_id** – unikátní celé číslo, primární klíč
- **den** – řetězec dvou znaků, povolené hodnoty jsou pouze PO, ÚT, ST, ČT a PÁ, nemůže být null; reprezentuje den, v jaký se daná rozvrhová akce má konat
- **cas_od** – řetězec pěti znaků, nemůže být null; reprezentuje čas, kdy rozvrhová akce začíná, očekávaný formát je „HH:mm“, kde HH značí hodiny a mm minuty
- **cas_do** – řetězec pěti znaků. Nemůže být null; reprezentuje čas, kdy rozvrhová akce končí, očekávaný formát je „HH:mm“, kde HH značí hodiny a mm minuty
- **kapacita** – celočíselný typ, nemůže být null; reprezentuje maximální počet lidí, kteří se do kroužku vejdou

- **krouzek_id** – cizí klíč z tabulky krouzek; reprezentuje kroužek, který se v dané rozvrhové akci koná
- **mistnost_id** – cizí klíč z tabulky místnost; reprezentuje místo, kde se daná rozvrhová akce koná
- **vedouci_id** – cizí klíč z tabulky vedoucí; reprezentuje vedoucího dané rozvrhové akce

4.6. Přihláška

Tabulka reprezentuje jednotlivé přihlášky dětí na konkrétní rozvrhové akce. Jedná se o rozkladovou tabulku, spojuje primární klíče dětí a rozvrhových akcí. Přihláška je popsána následujícími atributy:

- **prihlaska_id** – unikátní celé číslo, primární klíč
- **datum_podani** – datový typ DATE, nemůže být null; reprezentuje datum, kdy byla přihláška podána
- **dite_id** – cizí klíč z tabulky dítě; reprezentuje dítě, které se přihlásilo na rozvrhovou akci
- **rozvrhova_akce_id** – cizí klíč z tabulky rozvrhová akce; reprezentuje rozvrhovou akci, na kterou se dítě přihlásilo

5. Pohledy

V této práci byly vytvořeny celkem tři pohledy – `v_prihlasene_deti_na_konkretni_rozvrhove_akce`, `v_ktere_krouzky_vede_konkretni_vedouci` a `v_pocet_deti_z_konkretniho_mesta`. Jejich název již dobře vypovídá o tom, co reprezentují.

5.1. Přihlášené děti na konkrétní rozvrhové akce

Pohled zobrazuje jméno a příjmení dětí, která jsou přihlášená na konkrétní kroužek, který je popsán názvem, dnem a časem, kdy probíhá. Jedná se tedy o spojení 4 tabulek – z tabulky dětí se bere jméno a příjmení, přes tabulku přihlášek se zjistí konkrétní rozvrhová akce, na kterou je dítě přihláшено, z tabulky rozvrhových akcí se vypíše den a čas, kdy probíhá, z tabulky kroužku se získává název daného kroužku.

5.1.1. SQL

```
CREATE OR REPLACE VIEW v_prihlasene_deti_na_konkretni_rozvrhove_akce AS
SELECT
    d.jmeno,
    d.prijmeni,
    k.nazev,
    ra.den,
    ra.cas_od,
    ra.cas_do
FROM
    dite d
    INNER JOIN prihlaska p
        ON d.dite_id = p.dite_id
    INNER JOIN rozvrhova_akce ra
        ON ra.rozvrhova_akce_id = p.rozvrhova_akce_id
    INNER JOIN krouzek k
        ON k.krouzek_id = ra.krouzek_id
ORDER BY d.jmeno, d.prijmeni;
```

5.1.2. Tabulka výsledků

	JMENO	PRIJMENI	NAZEV	DEN	CAS_OD	CAS_DO
1	Anna	Müllerová	Keramika	PO	15:00	16:00
2	Anna	Müllerová	Angličtina pro začátečníky	PO	15:00	16:00
3	Anna	Müllerová	Počítačová grafika	ČT	15:00	16:00
4	Anna	Müllerová	Počítačová grafika	PÁ	15:00	16:00
5	Jan	Novák	Počítačová grafika	ČT	15:00	16:00
6	Jan	Novák	Keramika	ČT	17:00	18:00
7	Jan	Novák	Angličtina pro začátečníky	PO	15:00	16:00
8	Jan	Novák	Keramika	PO	15:00	16:00
9	Jan	Novák	Angličtina pro pokročilé	ST	16:00	17:00
10	Jan	Novák	Angličtina pro pokročilé	PO	16:00	17:00
11	Josef	Jiný	HTML pro začátečníky	ÚT	16:30	18:00
12	Josef	Jiný	Keramika	PO	17:00	18:00
13	Josef	Jiný	Míčové hry	ST	15:00	17:00
14	Josef	Novák	Počítačová grafika	PÁ	15:00	16:00
15	Josef	Novák	Java pro začátečníky	ÚT	15:00	16:30
16	Josef	Novák	Angličtina pro začátečníky	PO	15:00	16:00
17	Petr	Starý	Java pro začátečníky	ÚT	15:00	16:30
18	Petr	Starý	Keramika	PO	15:00	16:00
19	Petr	Starý	Angličtina pro pokročilé	PO	16:00	17:00
20	Petr	Starý	Keramika	ČT	17:00	18:00

5.2. Které kroužky vede konkrétní vedoucí

Pohled zobrazuje jméno a příjmení vedoucích, kteří vedou konkrétní kroužek, který je popsán názvem a dnem a časem, kdy probíhá. Jedná se tedy o spojení 3 tabulek – z tabulky vedoucí se bere jméno a příjmení, z tabulky rozvrhových akcí se vyberou odpovídající rozvrhové akce danému vedoucímu a vezme se z nich den, čas od a čas do. Nakonec se z tabulky kroužek vybere název odpovídající dané vyučované rozvrhové akci.

5.2.1. SQL

```
CREATE OR REPLACE VIEW v_ktere_krouzky_vede_konkretni_vedouci AS
SELECT
    v.jmeno,
    v.prijmeni,
    k.nazev,
    ra.den,
    ra.cas_od,
    ra.cas_do
FROM
    vedouci v
    INNER JOIN rozvrhova_akce ra
        ON v.vedouci_id = ra.vedouci_id
    INNER JOIN krouzek k
        ON k.krouzek_id = ra.krouzek_id
ORDER BY v.jmeno, v.prijmeni;
```


5.2.2. Tabulka výsledků

	JMENO	PRIJMENI	NAZEV	DEN	CAS_OD	CAS_DO
1	Alena	Nová	Keramika	PO	17:00	18:00
2	Alena	Nová	Keramika	ČT	17:00	18:00
3	Alena	Nová	Keramika	PO	15:00	16:00
4	Dominik	Zappe	Java pro začátečníky	ÚT	15:00	16:30
5	Dominik	Zappe	HTML pro začátečníky	ÚT	16:30	18:00
6	Dominik	Zappe	Počítačová grafika	ČT	15:00	16:00
7	Martina	Hezká	Angličtina pro pokročilé	ST	16:00	17:00
8	Martina	Hezká	Angličtina pro pokročilé	PO	16:00	17:00
9	Martina	Hezká	Angličtina pro začátečníky	PO	15:00	16:00
10	Petr	Klíč	Počítačová grafika	PÁ	15:00	16:00
11	Petr	Klíč	Míčové hry	ST	15:00	17:00

5.3. Počet dětí z konkrétního města

Pohled zobrazuje město a počet dětí, které v něm bydlí. Jedná se o pohled vytvořený nad jednou jedinou tabulkou – dítě. Obsahuje agregační funkci – count.

5.3.1. SQL

```
CREATE OR REPLACE VIEW v_pocet_deti_z_konkretniho_mesta AS
SELECT
    d.mesto,
    COUNT(d.mesto) AS pocet_deti_ze_mesta
FROM
    dite d
GROUP BY d.mesto
ORDER BY COUNT(d.mesto) DESC;
```

5.3.2. Tabulka výsledků

	MESTO	POCET_DETI_ZE_MESTA
1	Plasy	2
2	Kaznějov	1
3	Plzeň	1
4	Obora	1

6. Scénáře

Byly vytvořeny tři scénáře. První dva ověřují funkčnost pohledů. Tedy vkládají, upravují a odebírají korektně data z tabulek. Třetí scénář ověřuje funkčnost navrženého modelu a integritu databáze. Tedy snaží se do databáze zavést buď nekorektní, nebo nepovolené hodnoty. Nebo např. odebrat hodnoty nadřazené tabulky, které jsou zapotřebí v podřazené tabulce.

6.1. Scénář 1

Jan Novák si chce odhlásit čtvrtěční Keramiku od 17:00.

```
DELETE FROM prihlaska
WHERE dite_id = 5 AND rozvrhova_akce_id = 10;
```

Navíc chce začít s Josefem chodit na úterní Javu od 15:00.

```
INSERT INTO prihlaska (prihlaska_id, datum_podani, dite_id,
rozvrhova_akce_id)
VALUES(21, TO_DATE('15.10.2021', 'DD.MM.YYYY'), 5, 5);
```

Aby toho nebylo málo, jeho maminka si změnila telefonní číslo na 666666666.

```
UPDATE dite
SET telefon_domu = 666666666
WHERE dite_id = 5;
```

6.2. Scénář 2

Martinu Hezkou už nebaví angličtina a chtěla by učit děti německy.

Proto zakládá nový kroužek – Němčina pro začátečníky.

```
INSERT INTO krouzek (krouzek_id, nazev)
VALUES (8, 'Němčina pro začátečníky');

INSERT INTO rozvrhova_akce
(rozvrhova_akce_id, den, cas_od, cas_do, kapacita,
krouzek_id, mistnost_id, vedouci_id)
VALUES (12, 'PÁ', '17:00', '18:00', 25, 8, 4, 1);
```

Petrovi Starému se to líbí a hodí, a tak se přihlásí do tohoto kroužku.

```
INSERT INTO prihlaska (prihlaska_id, datum_podani, dite_id,
rozvrhova_akce_id)
VALUES(22, TO_DATE('20.10.2021', 'DD.MM.YYYY'), 2, 12);
```

Petr do kroužku nalákal i své dva kamarády ze města, kteří jsou v DDM noví.

```
INSERT INTO dite
(dite_id, jmeno, prijmeni, rodne_cislo,
ulice, cislo_popisne, mesto, psc, telefon_domu)
VALUES (6, 'Martin', 'Dobrý', 9812121212,
'Spagátovka', 89, 'Kaznějov', 33151, 111000111);

INSERT INTO prihlaska (prihlaska_id, datum_podani, dite_id,
rozvrhova_akce_id)
VALUES(23, TO_DATE('22.10.2021', 'DD.MM.YYYY'), 6, 12);
```

```

INSERT INTO dite
    (dite_id, jmeno, prijmeni, rodne_cislo,
     ulice, cislo_popisne, mesto, psc, telefon_domu)
VALUES (7, 'Matěj', 'Čech', 0522102548,
       'Špičatá', 88, 'Kaznějov', 33151, 222111000);

INSERT INTO prihlaska (prihlaska_id, datum_podani, dite_id,
                       rozvrhova_akce_id)
VALUES(24, TO_DATE('22.10.2021', 'DD.MM.YYYY'), 7, 12);

```

6.3. Scénář 3

Tento scénář má otestovat funkčnost návrhu, tedy zavést nepovolené hodnoty a pokusit se vyvolat nekonzistenci.

Nejprve se pokusíme zavést do databáze třeba dítě beze jména.

```

INSERT INTO dite
    (dite_id, prijmeni, rodne_cislo,
     ulice, cislo_popisne, mesto, psc, telefon_domu)
VALUES (8, '', 123123123,
       'Test', 1, 'Test', 11111, 123456789);

```

Nyní zkusíme zavést dítě se stringovým PSČ.

```

INSERT INTO dite
    (dite_id, jmeno, prijmeni, rodne_cislo,
     ulice, cislo_popisne, mesto, psc, telefon_domu)
VALUES (9, 'Test', 'Test', 123123123,
       'Test', 1, 'Test', 'Test', 123456789);

```

Zkouška integritního omezení proti prázdnému řetězci.

```

INSERT INTO dite
    (dite_id, jmeno, prijmeni, rodne_cislo,
     ulice, cislo_popisne, mesto, psc, telefon_domu)
VALUES (9, '', 'Test', 123123123,
       'Test', 1, 'Test', '11111', 123456789);

```

```

INSERT INTO krouzek (krouzek_id, nazev) VALUES (9, '');

```

Zkouška integritního omezení proti telefonním číslům, která nemají 9 míst.

```

INSERT INTO dite
    (dite_id, jmeno, prijmeni, rodne_cislo,
     ulice, cislo_popisne, mesto, psc, telefon_domu)
VALUES (9, 'Test', 'Test', 123123123,
       'Test', 1, 'Test', '11111', 123);

```

```

UPDATE vedouci
SET telefon = 123
WHERE vedouci_id = 1;

```

Zkouška integritního omezení pro dny v týdnu

```
UPDATE rozvrhova_akce  
  SET den = 'SO'  
  WHERE rozvrhova_akce_id = 1;
```

```
UPDATE rozvrhova_akce  
  SET den = 'Pondělí'  
  WHERE rozvrhova_akce_id = 1;
```

Pro vyvolání nekonzistence zkusíme odebrat dítě, které je přihlášeno na více kroužků.

```
DELETE FROM dite  
  WHERE dite_id = 1;
```

7. Závěr

Cílem práce bylo seznámení se s jazykem SQL a prací s databází. K tvorbě této konkrétní databáze mě přivedla vlastní zkušenost, neboť s Domem dětí a mládeže v Plasích úzce spolupracuji již několik let. Databáze určitě není dokonalá, neboť by mohla být rozšířená např. o externí a interní pracovníky, namísto vedoucích. Dále se dělají kroužky i pro dospělé, tedy tabulka dětí také není postačující pro reálné užití v praxi. Cíl osvojení si základních dovedností v jazyce SQL byl ale splněn.