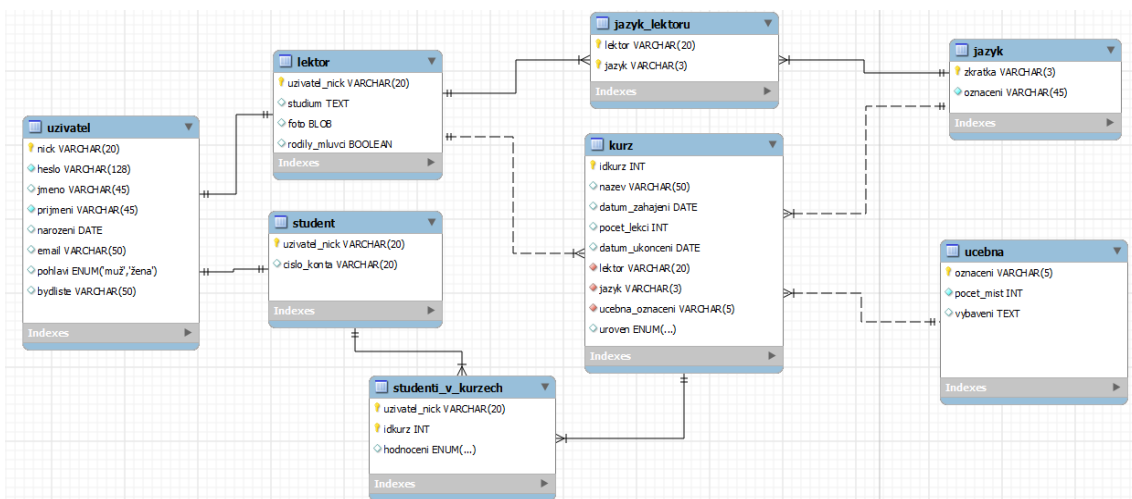


Praktická prověrka – dotazy SQL – jazyková škola

S využitím aplikace PHPMyAdmin proveďte úkoly související se správou MySQL databáze používané jazykovou školou. Při plnění úkolů vycházejte z následujícího databázového schématu:



Úkoly:

- Připojte se k lokálnímu MySQL serveru. Vypište seznam všech databází. Vytvořte v MySQL databázi s názvem *langschoool*. Importujte do databáze data ze souboru **langschoool.sql**. Ověřte správnost kódování znaků, případně proveďte potřebné úpravy, aby bylo zajištěno správné zobrazování českých písmen. (2 body)

```
mysql -u root -p
show databases;
create database langschoool;
mysql -u root -p langschoool < langschoool.sql
```

- Vytvořte v databázi tabulku *ucebnice*, která bude obsahovat následující atributy: (6 bodů)

Název atributu	Doména atributu	Vzorový záznam
ISBN	primární klíč	978-0-19-434074-8
Titul	neprázdný znakový řetězec o délce max. 100 znaků, povinně zadávaná hodnota	<i>Clockwise pre-intermediate</i>
Autor	znakový řetězec o délce max. 50 znaků	<i>Bruce McGowen</i>
Rok_vydani	celé číslo (4 numerické znaky)	2000
Uroven	volby: začátečník, mírně pokročilý, pokročilý	<i>mírně pokročilý</i>
Popis	dlouhý souvislý text	<i>This energic course develops fluency, refreshes key grammar areas, and extends active vocabulary.</i>
Cena	desetinné číslo s pevnou řadovou čárkou (max. hodnota 9999.99)	350,50

```
CREATE TABLE `langschoool`.`ucebnice`
(`ISBN` VARCHAR(20) NOT NULL ,
```

```

`titul` VARCHAR(100) NOT NULL ,
`autor` VARCHAR(50) NULL ,
`rok_vydani` INT(4) NULL ,
`uroven` ENUM('začátečník','mírně pokročilý','pokročilý') NULL ,
`popis` TEXT NULL ,
`cena` DECIMAL(6,2) NULL ,
PRIMARY KEY (`ISBN`) ENGINE = InnoDB

```

c. Vložte do tabulky *ucebnice* vzorový záznam. (2 body)

```

INSERT INTO `ucebnice` (`ISBN`, `titul`, `autor`, `rok_vydani`, `uroven`, `popis`, `cena`)
VALUES ('978-0-19-434074-8', 'Clockwise pre-intermediate', 'Bruce McGowen', '2000', 'mírně pokročilý', 'This energetic course develops fluency, refreshes key grammar areas, and extends active vocabulary', '350.50');

```

d. Vytvořte výběrový dotaz, který z databáze vypíše všechny přihlášené studenty kurzu *Angličtina - nic jednoduššího*, kteří *pocházejí z Brna nebo Ostravy*. Upravte výsledek dotazu podle níže uvedeného vzoru. Seřaďte záznamy primárně podle bydliště studentů (v abecedním pořadí) a sekundárně podle data narození (od nejmladších po nejstarší). (6 bodů)

```

SELECT
CONCAT(jmeno, " ", prijmeni) AS `Jméno studenta`,
student.cislo_konta AS `Číslo konta`,
DATE_FORMAT(narozeni, "%e. %c. %Y") AS `Datum narození`,
bydliste AS `Bydliště`
FROM `uzivatel`
JOIN student ON uzivatel.nick = student.uzivatel_nick
JOIN studenti_v_kurzech ON student.uzivatel_nick = studenti_v_kurzech.uzivatel_nick
JOIN kurz ON studenti_v_kurzech.idkurz = kurz.idkurz
WHERE kurz.nazev LIKE "Angličtina - nic jednoduššího"
AND bydliste IN ("Ostrava", "Brno")
ORDER BY `Bydliště`, narozeni DESC

```

Jméno studenta	Číslo konta	Datum narození	Bydliště
Jiří Šperk	02317002/0600	8. 10. 1993	Brno
Tomáš Rybníček	07879913/0600	22. 7. 1992	Brno
Mariana Seidlová	02331133/0300	8. 6. 1991	Ostrava
Tomáš Světlý	04142001/0600	28. 12. 1984	Ostrava

e. Vytvořte dotaz **se seskupením**, který vypíše všechny kurzy angličtiny a němčiny pro *začátečníky* a vedle dalších údajů (viz vzor) zobrazí počet studentů, kteří se do daného kurzu přihlásili. Výsledek dotazu bude uspořádaný podle počtu studentů sestupně. (6 bodů)

Kurz	Jazyk	Úroveň	Učebna	Počet studentů
Učíme se německy	NEJ	začátečníci	UC01	10
Angličtina - nic jednoduššího	ANJ	začátečníci	UC02	7
Angličtina snadno a rychle	ANJ	začátečníci	UC03	5

```

SELECT
kurz.nazev AS `Kurz`,
kurz.jazyk AS `Jazyk`,
kurz.uroven AS `Úroveň`,
kurz.ucebna_oznaceni AS `Učebna`,
COUNT(studenti_v_kurzech.uzivatel_nick) AS `Počet studentů`
FROM kurz
JOIN studenti_v_kurzech ON studenti_v_kurzech.idkurz = kurz.idkurz
WHERE kurz.uroven LIKE "začátečníci" AND (kurz.jazyk LIKE "ANJ" OR kurz.jazyk LIKE "NEJ")
GROUP BY `Kurz`
ORDER BY `Počet studentů` DESC

```

Varianta s IN

```

SELECT
kurz.nazev AS `Kurz`,

```

```

kurz.jazyk AS `Jazyk`,
kurz.uroven AS `Úroveň`,
kurz.ucebna_oznaceni AS `Učebna`,
COUNT(studenti_v_kurzech.uzivatel_nick) AS `Počet studentů`
FROM kurz
JOIN studenti_v_kurzech ON studenti_v_kurzech.idkurz = kurz.idkurz
WHERE kurz.uroven LIKE "začátečníci" AND kurz.jazyk IN ("ANJ","NEJ")
GROUP BY `Kurz`
ORDER BY `Počet studentů` DESC

```

- f. Vypište všechny kurzy, které absolvovala *Milada Frejová*, a uveďte všechny potřebné informace podle vzoru. Kurzy seřaďte *vzestupně* podle sloupce *Rok ukončení kurzu*. (6 bodů)

Kurz	Úroveň	Rok ukončení kurzu	Lektor
Učíme se německy	začátečníci	2010	P. Bassová
Byznys angličtina	pokročilí	2010	E. Grey (rodilý mluvčí)
Základy španělštiny	začátečníci	2012	D. Lopez (rodilý mluvčí)
Angličtina pro pokročilé	pokročilí	2012	E. Grey (rodilý mluvčí)

```

SELECT
kurz.nazev AS `Kurz`, kurz.uroven AS `Úroveň`,
YEAR(kurz.datum_ukonzeni) AS `Rok ukončení kurzu`,
CONCAT(LEFT(uzivatel.jmeno, 1), ".", " ",
uzivatel.prijmeni, IF(lektor.rodily_mluvci = 'ano', " (rodilý mluvčí)", "")) AS `Lektor`
FROM `kurz`
JOIN lektor ON lektor.uzivatel_nick = kurz.lektor
JOIN uzivatel ON uzivatel.nick = kurz.lektor
JOIN studenti_v_kurzech ON studenti_v_kurzech.idkurz = kurz.idkurz
WHERE studenti_v_kurzech.uzivatel_nick = 'frejova'
ORDER BY `Rok ukončení kurzu`

```

- g. Vytvořte seskupovací dotaz, pomocí kterého získáte následující údaje o vybavení a vytížení jednotlivých učeben. Záznamy budou uspořádány sestupně podle počtu kurzů. (5 bodů)

Učebna	Počet míst	Počet lekcí	Počet kurzů
UC02	15	70	5
UC03	8	20	2
UC01	10	10	1

```

SELECT
kurz.ucebna_oznaceni AS `Učebna`,
ucebna.pocet_mist AS `Počet míst`,
SUM(kurz.pocet_lekci) AS `Počet lekcí`,
COUNT(kurz.idkurz) AS `Počet kurzů`
FROM ucebna
JOIN kurz ON ucebna.oznaceni = kurz.ucebna_oznaceni
GROUP BY ucebna.oznaceni
ORDER BY `Počet kurzů` DESC

```

- h. Vytvořte aktualizací dotaz, který sníží počet lekcí o *dvě* u všech kurzů, které dosud nebyly ukončeny (tj. *datum ukončení* je *NULL*) a které jsou určeny pro *začátečníky*. (4 body)

```

UPDATE kurz SET pocet_lekci = pocet_lekci - 2
WHERE datum_ukonzeni IS NULL AND uroven LIKE "začátečníci"

```

- i. Navrhněte odstraňovací dotaz, kterým byste vymazali z databáze všechny *uživatelky* narozené po roce 1980, které *nepocházejí z místa, jež obsahuje ve svém názvu písmeno 'o'*. Uveďte, jaký krok je ještě nezbytný k tomu, aby bylo možné záznamy skutečně vymazat. (3 body)

```

DELETE FROM `uzivatel`
WHERE pohlavi LIKE "žena" AND YEAR(narozeni) > 1980 AND bydliste LIKE "%o%"

```

```
ALTER TABLE `student` DROP FOREIGN KEY `fk_student_uzivatell`; ALTER TABLE `student` ADD CONSTRAINT `fk_student_uzivatell` FOREIGN KEY (`uzivatel_nick`) REFERENCES `uzivatel` (`nick`) ON DELETE CASCADE ON UPDATE NO ACTION;
```

```
ALTER TABLE `lektor` DROP FOREIGN KEY `fk_lektor_uzivatel`; ALTER TABLE `lektor` ADD CONSTRAINT `fk_lektor_uzivatel` FOREIGN KEY (`uzivatel_nick`) REFERENCES `uzivatel` (`nick`) ON DELETE CASCADE ON UPDATE NO ACTION;
```