

ЛАБОРАТОРНА РОБОТА №1
Створення і редагування таблиць в СУБД MySQL

Мета:

- сформувати у студентів практичні навички створення бази даних та таблиць в ній;
- визначення ключових полів таблиць; встановлення зв'язків між таблицями, встановлення обмежень на дані в таблицях, сортування даних в таблицях.

Завдання:

1. Вивчити необхідний теоретичний матеріал.
2. Виконати практичне завдання до лабораторної роботи.
3. Відповісти на контрольні теоретичні запитання.
4. Оформити звіт, який містить завдання до лабораторної роботи та вказівки для його виконання.

Варіант №1

1. *Увійти до СУБД MySQL. Створити нову базу даних **geography**.*

```
CREATE DATABASE geography COLLATE utf8mb4_0900_ai_ci;
```

```
USE geography;
```

2. *Створити таблиці з назвами **continents** і **countries**. Таблиця **continents** містить відомості про континенти Землі, таблиця **countries** – про країни цих континентів.*

```
CREATE TABLE `continents`  
(  
  `id`      int,  
  `name`    varchar(30) COLLATE utf8mb4_unicode_ci,  
  `area`    int,  
  `population` int  
) ENGINE = InnoDB  
  DEFAULT CHARSET = utf8mb4  
  COLLATE = utf8mb4_unicode_ci;
```

```
CREATE TABLE `countries`  
(  
  `id`          int,  
  `continent_id` int,  
  `name`        varchar(20) COLLATE utf8mb4_unicode_ci,  
  `area`        int,  
  `population`  int,  
  `capital`     varchar(25) COLLATE utf8mb4_unicode_ci,  
  `capital_population` int  
) ENGINE = InnoDB  
  DEFAULT CHARSET = utf8mb4  
  COLLATE = utf8mb4_unicode_ci;
```

3. *Встановити такі обмеження на поля таблиці **continents**:*

- додати індекс **unique** на поле **continents.id**, додати **PRIMARY KEY** та **AUTO_INCREMENT** значень, поле має бути додатнім числом, зменшити довжину поля;
- додати індекс **unique** на поле **continents.name** та значення поля має бути **NOT NULL**;
- поле **continents.area** повинно бути **NOT NULL**, значення по замовчуванню 1000, поле має бути додатнім числом;
- значення поля **continents.population** має знаходитись в межах від 0 до $5 \cdot 10^9$.

```
ALTER TABLE `continents` ADD CONSTRAINT PRIMARY KEY (`id`);
ALTER TABLE `continents` ADD CONSTRAINT `unique_id` UNIQUE (`id`);
ALTER TABLE `continents` ADD CONSTRAINT `unique_name` UNIQUE (`name`);
ALTER TABLE `continents` MODIFY `id` tinyint unsigned AUTO_INCREMENT;
ALTER TABLE `continents` MODIFY `name` varchar(30) NOT NULL;
ALTER TABLE `continents` MODIFY `area` int unsigned NOT NULL DEFAULT 1000;
ALTER TABLE `continents` MODIFY `population` bigint unsigned;
```

4. Заповнити таблицю **continents** даними про всі континенти Землі. Відсортувати таблицю за полем **continents.population** в порядку зростання.

```
INSERT INTO `continents` (id, name, area, population) VALUES (1, 'Північна Америка', 24250000, 528720100);
INSERT INTO `continents` (id, name, area, population) VALUES (2, 'Південна Америка', 18280000, 382000000);
INSERT INTO `continents` (id, name, area, population) VALUES (3, 'Антарктида', 13970000, 1000);
INSERT INTO `continents` (id, name, area, population) VALUES (4, 'Африка', 30300000, 922011000);
INSERT INTO `continents` (id, name, area, population) VALUES (5, 'Європа', 10186000, 747636045);
INSERT INTO `continents` (id, name, area, population) VALUES (6, 'Австралія', 7687000, 32000000);
INSERT INTO `continents` (id, name, area, population) VALUES (7, 'Азія', 44614500, 4164252000);
```

```
SELECT * FROM `continents` ORDER BY population ASC;
```

5. Встановити такі обмеження на поля таблиці **countries**:
- додати індекс **unique** на поле **countries.id**, додати **PRIMARY KEY** та **AUTO_INCREMENT** значень, поле має бути додатнім числом, зменшити довжину поля;
 - поле **countries.name** повинно бути **NOT NULL**;
 - поле **countries.area** повинно бути **NOT NULL**, і має бути додатнім числом;
 - поле **countries.population** має бути додатнім числом;
 - поле **countries.capital** повинно бути **NOT NULL**;
 - значення поля **countries.capital_population** має знаходитись в межах від 0 до $15 \cdot 10^6$.

```
ALTER TABLE `countries` ADD CONSTRAINT PRIMARY KEY (`id`);
ALTER TABLE `countries` ADD CONSTRAINT `unique_id` UNIQUE (`id`);
ALTER TABLE `countries` MODIFY `id` smallint unsigned AUTO_INCREMENT;
ALTER TABLE `countries` MODIFY `name` varchar(30) NOT NULL;
ALTER TABLE `countries` MODIFY `area` int unsigned NOT NULL;
```

```
ALTER TABLE `countries` MODIFY `population` int unsigned;  
ALTER TABLE `countries` MODIFY `capital` varchar(25) NOT NULL;  
ALTER TABLE `countries` MODIFY `capital_population` mediumint unsigned;
```

6. Для таблиць бази створити схему даних, в якій пов'язати між собою відповідні поля створених таблиць, встановивши зв'язок між таблицями "один-до-багатьох" із забезпеченням цілісності даних.

```
ALTER TABLE `countries` MODIFY `continent_id` tinyint unsigned NOT NULL;  
ALTER TABLE `countries` ADD CONSTRAINT `foreign_key_continent_id` FOREIGN KEY  
(`continent_id`) REFERENCES `continents` (`id`) ON DELETE CASCADE;
```

7. Заповнити таблицю **countries** даними про країни, що знаходяться на різних континентах (не менше 20 записів).

```
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (5, 'Австрія', 83871, 8219743, 'Відень', 1911191);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (5, 'Бельгія', 32547, 10438353, 'Брюссель', 163210);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (5, 'Ліхтенштейн', 160, 36713, 'Вадуц', 5450);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (5, 'Монако', 2, 30510, 'Монако', 30510);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (5, 'Нідерланди', 41526, 16730632, 'Амстердам', 863315);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (5, 'Німеччина', 357578, 81305856, 'Берлін', 3664088);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (5, 'Україна', 603700, 35000000, 'Київ', 2952301);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (2, 'Аргентина', 2766890, 40677348, 'Буенос-Айрес', 2891082);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (2, 'Болівія', 1098580, 8857870, 'Сукре', 225000);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (2, 'Бразилія', 8514877, 191908598, 'Бразилія', 3039444);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (2, 'Венесуела', 912050, 26414815, 'Каракас', 2245744);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (2, 'Гаяна', 214970, 770794, 'Джорджтаун', 1796);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (2, 'Колумбія', 1138910, 45013674, 'Богота', 8550000);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (2, 'Парагвай', 406750, 6347884, 'Асунсьйон', 637249);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (2, 'Перу', 1285220, 27925628, 'Ліма', 9674755);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (1, 'Канада', 9984670, 33573000, 'Оттава', 934243);  
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)  
VALUES (1, 'Мексика', 1964375, 112322757, 'Мехіко', 8918653);
```

```
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)
VALUES (4, 'Алжир', 2381741, 43851044, 'Алжир', 1519570);
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)
VALUES (4, 'Судан', 1886068, 43849260, 'Хартум', 639598);
INSERT INTO `countries` (continent_id, name, area, population, capital, capital_population)
VALUES (6, 'Австралія', 7688287, 25250000, 'Канберра', 381448);
```

8. Створити окремого користувача для бази даних **geography**, з правами *INSERT*, *SELECT*, *UPDATE*, *DELETE*.

```
CREATE USER 'geography'@'localhost' IDENTIFIED BY 'secret';
GRANT INSERT, SELECT, UPDATE, DELETE ON geography.* TO 'geography'@'localhost';
```