

Код створення самої бд:

```
CREATE DATABASE IF NOT EXISTS to_do_list3;
CREATE TABLE IF NOT EXISTS to_do_list3.categories (
    category_id INT PRIMARY KEY,
    name VARCHAR(255) NOT NULL
);
CREATE TABLE IF NOT EXISTS to_do_list3.priorities (
    priority_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    value INT NOT NULL
);
CREATE TABLE IF NOT EXISTS to_do_list3.tasks (
    task_id INT PRIMARY KEY,
    title VARCHAR(255) NOT NULL,
    description TEXT,
    priority_id INT,
    category_id INT,
    FOREIGN KEY (priority_id) REFERENCES priorities(priority_id),
    FOREIGN KEY (category_id) REFERENCES categories(category_id)
);
CREATE TABLE IF NOT EXISTS to_do_list3.days (
    day_id INT AUTO_INCREMENT PRIMARY KEY,
    day_name VARCHAR(60) NOT NULL
);
CREATE TABLE IF NOT EXISTS to_do_list3.schedule (
    schedule_id INT AUTO_INCREMENT PRIMARY KEY,
    task_id INT NOT NULL,
    day_id INT NOT NULL,
    FOREIGN KEY (task_id) REFERENCES tasks(task_id),
    FOREIGN KEY (day_id) REFERENCES days(day_id)
);
```

```
INSERT INTO to_do_list3.categories (category_id , name)
VALUES
(1, 'Work'),
(2, 'Personal'),
(3, 'Study');
```

```
INSERT INTO to_do_list3.priorities (name, value)
VALUES
('Optionally', 25),
('Less importantly', 50),
('Importantly', 75),
('Necessarily', 100);
```

```
INSERT INTO to_do_list3.tasks (task_id, title, description, priority_id, category_id)
VALUES
(1, 'Working with a project report', 'Write a project report by the deadline', 3, 1),
(2, 'Go to the gym', 'Hit the gym for an hour', 2, 2),
(3, 'Prepare for the exam', 'Read chapter 5 of the textbook for the upcoming exam', 2, 3);
```

```
INSERT INTO to_do_list3.tasks (task_id, title, priority_id, category_id)
VALUES
(4, 'Go to the hairdresser', 1, 2),
(5, 'Go for a walk', 3, 2),
(6, 'Do homework', 4, 3);
```

```
INSERT INTO to_do_list3.days (day_name)
```

```
VALUES
('Monday'),
('Tuesday'),
('Wednesday'),
('Thursday'),
('Friday'),
('Saturday'),
('Sunday');

INSERT INTO to_do_list3.schedule (task_id, day_id)
VALUES
(1, 1),
(2, 1),
(3, 1),
(6, 1),
(4, 2),
(2, 2),
(3, 2),
(6, 2),
(2, 3),
(6, 3),
(1, 4),
(3, 4),
(6, 4),
(3, 5),
(5, 5),
(6, 5),
(2, 6),
(3, 6),
(5, 7),
(1, 7);
```

Створення dump

```
sh-4.4# mysqldump -uroot -p to_do_list3 > dump_to_do_list3.sql
```

Запити:

1) Отримати всі завдання, впорядковані за пріоритетом:

```
SELECT t.title, p.name AS priority
FROM to_do_list3.tasks t
JOIN to_do_list3.priorities p ON t.priority_id = p.priority_id
ORDER BY p.value DESC;
```

	title	priority
1	Do homework	Necessarily
2	Working with a project report	Importantly
3	Go for a walk	Importantly
4	Go to the gym	Less importantly
5	Prepare for the exam	Less importantly
6	Go to the hairdresser	Optionally

2) Знайти кількість завдань у кожній категорії:

```
SELECT c.name AS category, COUNT(*) AS task_count
FROM to_do_list3.tasks t
JOIN to_do_list3.categories c ON t.category_id = c.category_id
GROUP BY c.name;
```

category	task_count
Work	1
Personal	3
Study	2

3) Знайти загальну кількість завдань у кожній категорії за пріоритетом:

```
SELECT c.name AS category, p.name AS priority, COUNT(*) AS task_count
FROM to_do_list3.tasks t
JOIN to_do_list3.categories c ON t.category_id = c.category_id
JOIN to_do_list3.priorities p ON t.priority_id = p.priority_id
GROUP BY c.name, p.name;
```

category	priority	task_count
Personal	Optionally	1
Personal	Less importantly	1
Study	Less importantly	1
Work	Importantly	1
Personal	Importantly	1
Study	Necessarily	1

4) Отримати список завдань, які мають важливий пріоритет та плануються на початок тижня:

```
SELECT t.title, d.day_name
FROM to_do_list3.tasks t
JOIN to_do_list3.schedule s ON t.task_id = s.task_id
JOIN to_do_list3.days d ON s.day_id = d.day_id
JOIN to_do_list3.priorities p ON t.priority_id = p.priority_id
WHERE p.value >= 75
AND d.day_id IN (1, 2, 3)
ORDER BY d.day_id;
```

ks(+)	1	×
SELECT t.title, d.day_name FROM to_	Введіть вираз SQL для фільтрації результатів (використовуйте	
title	day_name	
Working with a project report	Monday	
Do homework	Monday	
Do homework	Tuesday	
Do homework	Wednesday	

5) Знайти кількість завдань кожного дня

```
SELECT d.day_name, COUNT(*) AS task_count
FROM to_do_list3.days d
JOIN to_do_list3.schedule s ON d.day_id = s.day_id
GROUP BY d.day_name
ORDER BY task_count DESC;
```

days	1	×
SELECT d.day_name, COUNT(*) AS t	Введіть вираз SQL для фільтрації результатів (ε	
day_name	task_count	
1 Monday	4	
2 Tuesday	4	
3 Thursday	3	
4 Friday	3	
5 Wednesday	2	
6 Saturday	2	
7 Sunday	2	

6) Знайти найбільш популярний день для планування завдань:

```
SELECT d.day_name, COUNT(*) AS task_count
FROM to_do_list3.days d
JOIN to_do_list3.schedule s ON d.day_id = s.day_id
GROUP BY d.day_name
HAVING COUNT(*) = (
    SELECT COUNT(*)
    FROM to_do_list3.schedule
    GROUP BY day_id
    ORDER BY COUNT(*) DESC
    LIMIT 1
);
```

ays	1	×
SELECT d.day_name, COUNT(*) AS t	Введіть вираз SQL для фільтрації результатів (використовуї	
day_name	task_count	
1 Monday	4	
2 Tuesday	4	

7) Отримати список всіх завдань з їх категорією та пріоритетом:

```
SELECT t.title, c.name AS category, p.name AS priority
FROM to_do_list3.tasks t
JOIN to_do_list3.categories c ON t.category_id = c.category_id
JOIN to_do_list3.priorities p ON t.priority_id = p.priority_id;
```

tasks(+) 1 ×

SELECT t.title, c.name AS category, p.name AS priority

	title	category	priority
1	Go to the hairdresser	Personal	Optionally
2	Go to the gym	Personal	Less importantly
3	Prepare for the exam	Study	Less importantly
4	Working with a project report	Work	Importantly
5	Go for a walk	Personal	Importantly
6	Do homework	Study	Necessarily

8) Отримати список завдань, що мають середній пріоритет:

```
SELECT t.title, p.name AS priority
FROM to_do_list3.tasks t
JOIN to_do_list3.priorities p ON t.priority_id = p.priority_id
WHERE p.value BETWEEN 50 AND 75;
```

cs(+) 1 ×

SELECT t.title, p.name AS priority

	title	priority
	Go to the gym	Less importantly
	Prepare for the exam	Less importantly
	Working with a project report	Importantly
	Go for a walk	Importantly

Всі стовпці доступні лише

9) Отримати список завдань, які не мають опису:

```
SELECT t.title
FROM to_do_list3.tasks t
WHERE t.description IS NULL OR t.description = '';
```

tasks 1 ×

SELECT t.title FROM to_do_list3.tasks

	title
1	Go to the hairdresser
2	Go for a walk
3	Do homework

10) Розклад:

```
SELECT d.day_name, GROUP_CONCAT(t.title) AS tasks
FROM to_do_list3.days d
LEFT JOIN to_do_list3.schedule s ON d.day_id = s.day_id
LEFT JOIN to_do_list3.tasks t ON s.task_id = t.task_id
GROUP BY d.day_name;
```

days 1 ×

SELECT d.day_name, GROUP_CONCAT(t.title) AS tasks

	day_name	tasks
1	Friday	Prepare for the exam,Go for a walk,Do homework
2	Monday	Working with a project report,Go to the gym,Prepare for the exam,Do homework
3	Saturday	Go to the gym,Prepare for the exam
4	Sunday	Go for a walk,Working with a project report
5	Thursday	Working with a project report,Prepare for the exam,Do homework
6	Tuesday	Go to the hairdresser,Go to the gym,Prepare for the exam,Do homework
7	Wednesday	Go to the gym,Do homework