**ЛР №6**

**Проектування багатотабличних запитів (внутрішнє з'єднання, inner join, left join, right join, self join, subquery, correlated subquery, exists, not exist, union)**

1. Вивести значення наступних колонок: назва книги, ціна, назва видавництва. Використовувати внутрішнє з'єднання, застосовуючи where.

**SELECT books.title, books.price, books.publisher FROM books, publishers WHERE books.publisher = publishers.publisherID**;

2. Вивести значення наступних колонок: назва книги, назва категорії. Використовувати внутрішнє з'єднання, застосовуючи inner join.

**SELECT books.title, categories.category FROM books INNER JOIN categories ON books.category = categoryID;**

3. Вивести значення наступних колонок: назва книги, ціна, назва видавництва, формат.

**SELECT books.title, books.price, publishers.publisher, books.format from books INNER JOIN publishers ON books.publisher = publisherID;**

4. Вивести значення наступних колонок: тема, категорія, назва книги, назва видавництва. Фільтр по темам і категоріям.

**SELECT**

**topics.topic,**

**categories.category,**

**books.title, publishers.**

**publisher**

**FROM books**

**INNER JOIN topics ON books.topic = topics.topicID**

**INNER JOIN categories ON books.category = categories.categoryID**

**INNER JOIN publishers ON books.publisher = publishers.publisherID**

**ORDER BY books.topic DESC, books.category DESC;**

5. Вивести книги видавництва 'BHV', видані після 2000 р

**SELECT**

**books.title,**

**publishers.publisher**

**FROM books**

**INNER JOIN publishers ON books.publisher = publishers.publisherID**

**WHERE publishers.publisher = 'BHV' AND YEAR(books.date) > 2000;**

6. Вивести загальну кількість сторінок по кожній назві категорії. Фільтр по спадаючій кількості сторінок.

**SELECT categories.category, SUM(books.pages)**

**FROM books**

**INNER JOIN categories ON books.category = categories.categoryID**

**GROUP BY books.category ORDER BY SUM(books.pages) DESC;**

7. Вивести середню вартість книг по темі 'Використання ПК' і категорії 'Linux'.

**SELECT AVG(books.price)**

**FROM books**

**INNER JOIN topics ON books.topic = topics.topicID**

**INNER JOIN categories ON books.category = categories.categoryID**

**WHERE topics.topic = 'Використання ПК в цілому' AND categories.category = 'Linux';**

8. Вивести всі дані універсального відношення. Використовувати внутрішнє з'єднання, застосовуючи where.

**SELECT**

**b.n,**

**b.id,**

**b.novelty,**

**b.title,**

**b.price,**

**p.publisher,**

**b.pages,**

**b.format,**

**b.date,**

**b.circulation,**

**t.topic,**

**c.category**

**FROM books AS b, publishers AS p, topics AS t, categories AS c**

**WHERE b.publisher = p.publisherID AND b.topic = t.topicID AND b.category = c.categoryID;**

9. Вивести всі дані універсального відношення. Використовувати внутрішнє з'єднання, застосовуючи inner join.

**SELECT**

**b.n,**

**b.id,**

**b.novelty,**

**b.title,**

**b.price,**

**p.publisher,**

**b.pages,**

**b.format,**

**b.date,**

**b.circulation,**

**t.topic,**

**c.category**

**FROM books AS b**

**INNER JOIN topics AS t ON b.topic = t.topicID**

**INNER JOIN categories AS c ON b.category = c.categoryID**

**INNER JOIN publishers AS p ON b.publisher = p.publisherID**

10. Вивести всі дані універсального відношення. Використовувати зовнішнє з'єднання, застосовуючи left join / rigth join.

**SELECT**

**b.n,**

**b.id,**

**b.novelty,**

**b.title,**

**b.price,**

**p.publisher,**

**b.pages,**

**b.format,**

**b.date,**

**b.circulation,**

**t.topic,**

**c.category**

**FROM books AS b**

**LEFT JOIN topics AS t ON b.topic = t.topicID**

**LEFT JOIN categories AS c ON b.category = c.categoryID**

**LEFT JOIN publishers AS p ON b.publisher = p.publisherID**

**SELECT**

**b.n,**

**b.id,**

**b.novelty,**

**b.title,**

**b.price,**

**p.publisher,**

**b.pages,**

**b.format,**

**b.date,**

**b.circulation,**

**t.topic,**

**c.category**

**FROM books AS b**

**RIGHT JOIN topics AS t ON b.topic = t.topicID**

**RIGHT JOIN categories AS c ON b.category = c.categoryID**

**RIGHT JOIN publishers AS p ON b.publisher = p.publisherID**

11. Вивести пари книг, що мають однакову кількість сторінок. Використовувати само об’єднання і аліаси (self join).

**SELECT b1.title AS book1, b2.title AS book2, b1.pages**

**FROM books AS b1**

**INNER JOIN books AS b2 ON b1.pages = b2.pages**

**WHERE b1.n <> b2.n;**

12. Вивести тріади книг, що мають однакову ціну. Використовувати самооб'єднання і аліаси (self join).

**SELECT b1.title AS book1, b2.title AS book2, b3.title AS book3, b1.price**

**FROM books AS b1**

**INNER JOIN books AS b2 ON b1.price = b2.price**

**INNER JOIN books AS b3 ON b1.price = b3.price**

**WHERE b1.n != b2.n AND b1.n != b3.n AND b2.n != b3.n;**

13. Вивести всі книги категорії 'C ++'. Використовувати підзапити (subquery).

**SELECT \***

**FROM books**

**WHERE category = (SELECT categoryID FROM categories WHERE categories.category = 'C&C++');**

14. Вивести книги видавництва 'BHV', видані після 2000 р Використовувати підзапити (subquery).

**SELECT \***

**FROM books**

**WHERE publisher = (SELECT publisherID FROM publishers WHERE publishers.publisher LIKE '%BHV%') AND YEAR(date) > 2000;**

15. Вивести список видавництв, у яких розмір книг перевищує 400 сторінок. Використовувати пов'язані підзапити (correlated subquery).

**SELECT publisher**

**FROM publishers AS p**

**WHERE EXISTS (**

**SELECT 1**

**FROM books AS b**

**WHERE b.publisher = p.publisherID**

**AND b.pages >= 400**

**);**

16. Вивести список категорій в яких більше 3-х книг. Використовувати пов'язані підзапити (correlated subquery).

**SELECT category**

**FROM categories AS c**

**WHERE (**

**SELECT COUNT(\*)**

**FROM books AS b**

**WHERE b.category = c.categoryID**

**) > 3;**

17. Вивести список книг видавництва 'BHV', якщо в списку є хоча б одна книга цього видавництва. Використовувати exists.

**SELECT \***

**FROM books AS b**

**WHERE EXISTS (**

**SELECT 1**

**FROM publishers AS p**

**WHERE p.publisherID = b.publisher**

**AND p.publisher LIKE '%BHV%'**

**);**

18. Вивести список книг видавництва 'BHV', якщо в списку немає жодної книги цього видавництва. Використовувати not exists.

**SELECT \***

**FROM books AS b**

**WHERE NOT EXISTS (**

**SELECT 1**

**FROM publishers AS p**

**WHERE p.publisherID = b.publisher**

**AND p.publisher LIKE '%BHV%'**

**);**

19. Вивести відсортований загальний список назв тем і категорій. Використовувати union.

**SELECT topic AS name**

**FROM topics**

**UNION**

**SELECT category AS name**

**FROM categories**

**ORDER BY name**

20. Вивести відсортований в зворотному порядку загальний список перших слів, назв книг і категорій що не повторюються. Використовувати union.

**SELECT SUBSTRING\_INDEX(title, ' ', 1) AS name**

**FROM books**

**UNION**

**SELECT SUBSTRING\_INDEX(category, ' ', 1) AS name**

**FROM categories**

**ORDER BY name DESC;**