Question & Answer

```
-- 1) Retrieve all books in the "Fiction" genre:
SELECT * FROM books
WHERE genre = 'Fiction';

-- 2) Find books published after the year 1950:
SELECT * FROM books
WHERE published_year>1950;

-- 3) List all customers from the Canada:
SELECT * FROM customers
WHERE country='Canada';

-- 4) Show orders placed in November 2023:
SELECT * FROM orders
WHERE order_date BETWEEN '2023-11-01' AND '2023-11-01';

-- 5) Retrieve the total stock of books available:
SELECT * FROM books;
SELECT SUM (stock) AS total_stock
FROM books;

-- 6) Find the details of the most expensive book:
SELECT * FROM books
ORDER BY price DESC
LIMIT 1;

-- 7) Show all customers who ordered more than 1 quantity of a book:
SELECT * FROM orders
WHERE quantity >1;
```

```
-- 8) Retrieve all orders where the total amount exceeds $20:
SELECT * FROM orders
WHERE total_amount>20;

-- 9) List all genres available in the Books table:
SELECT * FROM books
SELECT DISTINCT genre FROM books;

-- 10) Find the book with the lowest stock:
SELECT * FROM books
ORDER BY stock
LIMIT 1;

-- 11) Calculate the total revenue generated from all orders:
SELECT * FROM orders;
SELECT SUM (total_amount)
AS revenue
FROM orders;

-- 12) Retrieve the total number of books sold for each genre:
SELECT * FROM orders;
SELECT * FROM books;
SELECT b.genre, SUM(o.quantity) AS Total_book_sold
FROM orders o
JOIN books b ON o.book_id =b.book_id
GROUP BY b.genre;

-- 13) Find the average price of books in the "Fantasy" genre:
SELECT * FROM Books;
SELECT AVG(price) AS Average_price
FROM books
```

```
WHERE genre='Fantasy';

-- 14) List customers who have placed at least 2 orders:
SELECT * FROM orders
SELECT customer_id, COUNT(order_id) AS order_count
FROM orders
GROUP BY customer_id
HAVING COUNT(order_id)>=2;

-- 15) Find the most frequently ordered book:
SELECT * FROM books
SELECT * FROM orders
SELECT o.Book_id, b.title, COUNT(o.order_id) AS ORDER_COUNT
FROM orders o
JOIN books b ON o.book_id=b.book_id
GROUP BY o.book_id, b.title
ORDER BY ORDER_COUNT DESC LIMIT 1;

-- 16) Show the top 3 most expensive books of 'Fantasy' Genre :
SELECT * FROM books
WHERE genre ='Fantasy'
ORDER BY price DESC LIMIT 3;

-- 17) Retrieve the total quantity of books sold by each author:
SELECT * FROM books
SELECT * FROM orders
SELECT b.author, SUM(o.quantity) AS Total_Books_Sold
FROM orders o
JOIN books b ON o.book_id=b.book_id
GROUP BY b.Author;
```

```
-- 18) List the cities where customers who spent over $30 are located:
SELECT * FROM Customers;
SELECT * FROM orders;
SELECT DISTINCT c.city, total_amount
FROM orders o
JOIN customers c ON o.customer_id=c.customer_id
WHERE o.total_amount > 30;

-- 19) Find the customer who spent the most on orders:
SELECT * FROM Customers;
SELECT * FROM orders;
SELECT c.customer_id, c.name, SUM(o.total_amount) AS Total_Spent
FROM orders o
JOIN customers c ON o.customer_id=c.customer_id
GROUP BY c.customer_id, c.name
ORDER BY Total_spent Desc LIMIT 1;

--20) Calculate the stock remaining after fulfilling all orders:
SELECT * FROM books;
SELECT * FROM orders;

SELECT b.book_id, b.title, b.stock, COALESCE(SUM(o.quantity),0) AS Order_quantity,
   b.stock- COALESCE(SUM(o.quantity),0) AS Remaining_Quantity
FROM books b
LEFT JOIN orders o ON b.book_id=o.book_id
GROUP BY b.book_id ORDER BY b.book_id;
```