**Aim**:  
The aim of this practical is to learn how to use SQL clauses such as LIKE, AND, BETWEEN, NOT BETWEEN, IN, and NOT IN to filter and retrieve specific data from a database. It focuses on combining multiple conditions to refine query results and enhance data retrieval efficiency.

**Objective**:  
The purpose of this practical is to explore and apply different SQL clauses, including LIKE, AND, BETWEEN, NOT BETWEEN, IN, and NOT IN. These clauses are used to filter data based on certain conditions.

**Database Setup**:  
To begin, we will create a table named Books to store information about various books. The table will have the following columns:

1. **BookID**: This will be the unique identifier for each book (Primary Key).
2. **Title**: The title of the book.
3. **Author**: The author of the book.
4. **PublicationYear**: The year the book was published.
5. **Price**: The price of the book.

After the table creation, we will insert some sample data into the table.

**SQL Code Explanation**

1. **Creating the Books Table**:  
   The SQL query used to create the Books table is as follows:
   * BookID is an integer column, and it is set as the primary key, which ensures that each book will have a unique identifier.
   * Title and Author are text columns that will store the book’s title and author name.
   * PublicationYear is an integer column used to store the year the book was published.
   * Price is a decimal column to store the price of the book.
2. **Inserting Sample Data**:  
   The following data is inserted into the Books table:
   * BookID: 1, Title: "The Great Gatsby", Author: "F. Scott Fitzgerald", PublicationYear: 1925, Price: 10.99
   * BookID: 2, Title: "1984", Author: "George Orwell", PublicationYear: 1949, Price: 8.99
   * BookID: 3, Title: "To Kill a Mockingbird", Author: "Harper Lee", PublicationYear: 1960, Price: 7.99
   * BookID: 4, Title: "The Catcher in the Rye", Author: "J.D. Salinger", PublicationYear: 1951, Price: 6.49
   * BookID: 5, Title: "Moby Dick", Author: "Herman Melville", PublicationYear: 1851, Price: 12.49
   * BookID: 6, Title: "Pride and Prejudice", Author: "Jane Austen", PublicationYear: 1813, Price: 9.99
   * BookID: 7, Title: "War and Peace", Author: "Leo Tolstoy", PublicationYear: 1869, Price: 14.99
   * BookID: 8, Title: "Brave New World", Author: "Aldous Huxley", PublicationYear: 1932, Price: 11.49

**SQL Queries Using Different Clauses**

1. **LIKE Clause**:  
   The LIKE clause is used to search for a specified pattern in a column. The percentage symbol (%) is used as a wildcard to represent any sequence of characters.  
   **Query**: Find all books whose title contains the word "The".
   * Query: SELECT \* FROM Books WHERE Title LIKE '%The%';
2. **AND Clause**:  
   The AND clause is used to combine multiple conditions. All conditions specified must be true for a record to be returned.  
   **Query**: Find books published after 1900 and priced above $10.
   * Query: SELECT \* FROM Books WHERE PublicationYear > 1900 AND Price > 10;
3. **BETWEEN Clause**:  
   The BETWEEN clause is used to filter results within a specific range, inclusive of the boundaries.  
   **Query**: Find books published between the years 1900 and 1950.
   * Query: SELECT \* FROM Books WHERE PublicationYear BETWEEN 1900 AND 1950;
4. **NOT BETWEEN Clause**:  
   The NOT BETWEEN clause is used to exclude records that fall within a specified range.  
   **Query**: Find books whose publication year is not between 1800 and 1900.
   * Query: SELECT \* FROM Books WHERE PublicationYear NOT BETWEEN 1800 AND 1900;
5. **IN Clause**:  
   The IN clause allows you to specify a list of values to search for. It returns records where the column value matches any of the values in the list.  
   **Query**: Find books written by either George Orwell or Harper Lee.
   * Query: SELECT \* FROM Books WHERE Author IN ('George Orwell', 'Harper Lee');
6. **NOT IN Clause**:  
   The NOT IN clause is used to exclude records where the column value matches any value in a specified list.  
   **Query**: Find books whose author is neither George Orwell nor Harper Lee.
   * Query: SELECT \* FROM Books WHERE Author NOT IN ('George Orwell', 'Harper Lee');
7. **NOT LIKE Clause**:  
   The NOT LIKE clause is used to exclude records where the column value matches a specific pattern.  
   **Query**: Find books whose author is not J.D. Salinger.
   * Query: SELECT \* FROM Books WHERE Author NOT LIKE 'J.D. Salinger';

**Conclusion**

In this practical, we explored the use of different SQL clauses to filter data from the Books table. Each clause serves a specific purpose:

* **LIKE**: To search for a pattern in a column.
* **AND**: To combine multiple conditions.
* **BETWEEN**: To filter data within a specified range.
* **NOT BETWEEN**: To exclude data within a specific range.
* **IN**: To match values from a list.
* **NOT IN**: To exclude values from a list.
* **NOT LIKE**: To exclude records that match a specific pattern.

These clauses are powerful tools for querying databases and can help refine search results to meet specific criteria.

**Reflection**

This practical helped me gain a deeper understanding of how SQL queries can be tailored using various clauses. By using LIKE, AND, BETWEEN, IN, NOT IN, and other clauses, I learned how to filter data effectively to meet complex search requirements. It was also helpful in understanding how to apply multiple conditions to retrieve data more accurately and efficiently.