1. Get data from the database.

The easiest way to verify if a person knows the basics of SQL is to ask them to retrieve data from a database.

Solution

Given three tables: `customers`, `products` and `orders` you need to create queries which retrieve all rows from them. Listing 1.1 presents simple `SELECT` statements.

Listing 1.1 – Example of simple `SELECT` statements.

```
SELECT * FROM customers;

SELECT * FROM products;

SELECT * FROM orders;
```

Please notice that the wildcard (`*`) causes all columns to be retrieved. In many cases, it is necessary to retrieve only part of the data. Listing 1.2 presents an example `SELECT` statement which retrieves only the first and last name of a customer.

Listing 1.2 – The `SELECT` statement which gets specified columns.

```
SELECT first_name, last_name FROM customers;
```

2. Get data from the database using a conditional statement.

Preparing conditional statements is one of the necessary skills every programmer must have. A developer needs to create queries which return only those records that fulfill a specified condition. To retrieve filtered data, the `WHERE` clause combined with `AND`, `OR` and `NOT` operators should be used.

WHERE

First, you are asked to prepare a query which returns all customers which are from the USA.

Listing 2.1 – Example of `SELECT` statement with `WHERE` clause.

```
SELECT * FROM customers
  WHERE country = 'USA';
```

OR

Second, you are asked to prepare a query which returns all customers which are from the USA or Canada.

Listing 2.2 – Example of `SELECT` statement with `OR` operator.

```
SELECT * FROM customers
  WHERE country = 'USA' OR country = 'Canada';
```

AND

Finally, the last most basic operator. You need to prepare a query which returns all products from supplier 'Brandon' and price lower than $20.

Listing 2.3 – Example of `SELECT` statement with `AND` operator.

```
SELECT * FROM products
  WHERE supplier = 'Brandon' AND price < 20;
```

3. Get data from the database using the IN operator.

The `IN` operator is very often used in `SELECT` statements. The `IN` operator can be seen as shorthand for multiple `OR` conditions, but it can also take the results from other `SELECT` queries as input.

List of values

You are asked to prepare a query which returns all customers which are from the following list of countries:

- USA,
- Canada,
- Australia,
- Great Britain,
- New Zealand.

Listing 3.1 presents the correct implementation of such a query.

Listing 3.1 – Example of `SELECT` statement with `IN` operator.

```
SELECT * FROM customers
  WHERE country IN ('USA', 'Canada', 'Australia' , 'Great Britain',
'New Zealand');
```

Subquery

Another way to use the `IN` clause is to pass a list of values by selecting data from another table. For example, you may be asked to retrieve products which were sold in quantities higher than 100. Listing 3.2 presents an example of such a query. From the `order_items` table, you retrieve a list of products identifiers which sold more than 100 items in one order. Next, such a list is passed to the `IN` clause. Finally, the `SELECT` statement returns all products which match previously selected identifiers.

Listing 3.2 – Example of subquery.

```
SELECT * FROM products
  WHERE id IN (SELECT product_id FROM order_items
                 WHERE quantity > 100);
```