wbs
WARWICK BUSINESS SCHOOL
THE UNIVERSITY OF WARWICK

For the Change Makers

Programming for Data Analytics

Week 1 : SQL Exercises
Information Systems and Management
Warwick Business School

- **please write SQL queries to retrieve the following data from the w3school practice database (https://www.w3schools.com/sql/trysql.asp?filename=trysql_op_in).**
- 1. Find all the UK-based suppliers and their phone numbers.
- 2. Find all the employees that know French.
- 3. Find the names and prices of the top 5 expensive cheese.
- 4. Find all customers who had purchased seafood before.
- 5. Find the names of employees who processed the orders in November of 1996.

- 1. Find all the UK-based suppliers and their phone numbers.

```
SELECT SupplierID, SupplierName, Phone
FROM Suppliers
WHERE Country = 'UK'
```

- 2. Find all the employees that know French.

```
SELECT EmployeeID, LastName, FirstName
FROM Employees
WHERE Notes LIKE '%French%'
```

- 3. Find the names and prices of the top 5 expensive cheese.

```
SELECT Products.ProductName, Products.Price FROM Products
JOIN Categories ON Products.CategoryID = Categories.CategoryID
WHERE Categories.CategoryName LIKE '%Cheese%'
OR Categories.Description LIKE '%Cheese%'
ORDER BY Products.Price DESC
LIMIT 5
```

- 4. Find all customers who had purchased seafood before.

```
SELECT DISTINCT Customers.CustomerName
FROM Customers, Categories, OrderDetails, Orders, Products
WHERE Customers.CustomerID = Orders.CustomerID
AND Orders.OrderID = OrderDetails.OrderID
AND OrderDetails.ProductID = Products.ProductID
AND Products.CategoryID = Categories.CategoryID
AND Categories.CategoryName = 'Seafood'
```

- 5. Find the names of employees who processed the orders in November of 1996.

```
SELECT DISTINCT Employees.EmployeeID, Employees.LastName,
Employees.FirstName
FROM Employees, Orders
WHERE Employees.EmployeeID = Orders.EmployeeID
AND Orders.OrderDate LIKE '1996-11%'
```

The date may be stored and displayed differently in different web browsers, you may need to change the matching string accordingly.

- Please write SQL queries to answer the following questions based the dataset "bigquery-public-data.Austin_bikeshare".

1. Find the top 10 most used bikes based on the frequency.
2. Find the top 10 most used bikes based on total trip durations.
3. Find and rank the average trip durations for each subscriber type.
4. Find the top 5 most popular destination stations for single trip subscriber.
5. Find the top 5 most popular destination stations for annual subscriber that are now closed.

```
SELECT bikeid, count(trip_id) AS counts
FROM `bigquery-public-data.austin_bikeshare.bikeshare_trips`
GROUP BY bikeid
ORDER BY counts DESC
LIMIT 3
```

```
SELECT bikeid, sum(duration_minutes) AS durations
FROM `bigquery-public-data.austin_bikeshare.bikeshare_trips`
Group By bikeid
ORDER BY durations DESC
LIMIT 3
```

```
SELECT avg(duration_minutes) AS average,subscriber_type
FROM `bigquery-public-data.austin_bikeshare.bikeshare_trips`
GROUP BY subscriber_type
ORDER BY average DESC
```

```
SELECT end_station_id, count(trip_id) AS visits
FROM `bigquery-public-data.austin_bikeshare.bikeshare_trips`
WHERE subscriber_type = "Single Trip"
GROUP BY end_station_id
ORDER BY visits desc
limit 5
```

```
SELECT end_station_id, count(trip_id) AS visits
FROM `bigquery-public-data.austin_bikeshare.bikeshare_trips` AS t1
JOIN `bigquery-public-data.austin_bikeshare.bikeshare_stations` AS t2
ON CAST(t1.end_station_id AS INT64) = t2.station_id
WHERE subscriber_type LIKE "%Annual%"
AND t2.status = "closed"
GROUP BY end_station_id
ORDER BY visits desc
LIMIT 5
```

End_station_id is stored as string in the dataset, so it will not match with station_id that are stored as number. Therefore, we need convert the data type of end_station_id into number.

```
SELECT end_station_id, count(trip_id) AS visits
FROM `bigquery-public-data.austin_bikeshare.bikeshare_trips` as t1,
`bigquery-public-data.austin_bikeshare.bikeshare_stations` as t2
WHERE CAST(t1.end_station_id AS INT64) = t2.station_id
AND subscriber_type LIKE "%Annual%"
AND t2.status = "closed"
GROUP BY end_station_id
ORDER BY visits desc
LIMIT 5
```