Name: Ayano Yamamoto

Module: Working with Data

Section A: Data Warehouse Modelling
- a) Data Import Process
- b) Fact Table Design
- c) Transforming Data into Data Warehouse

Section B: Data Analysis and Queries Using SQL
- a) Data Analysis
- b) SQL Queries

Section C: Machine Learning using SQL
- a) Case Table and Preparations
- b) Machine Learning Models

Section A: Data Warehouse Modelling

a) Data Import Process

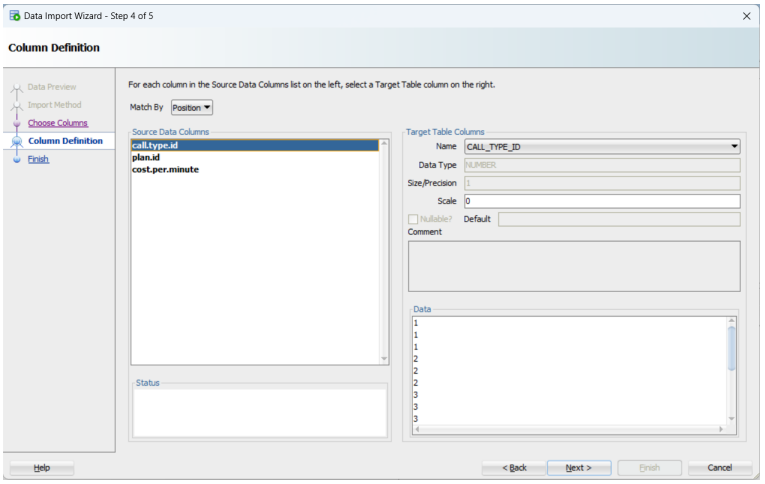
CSV files included in `data.zip` were imported into tables using the user interface in Oracle SQL Developer.

1. `call_rates.csv`

   Table creation:

```
CREATE TABLE call_rates (
       call_type_id NUMBER(1) NOT NULL,
       plan_id NUMBER(1) NOT NULL,
       cost_per_minute NUMBER(3, 2) NOT NULL,
       CONSTRAINT unq_call_rates UNIQUE (call_type_id, plan_id)
);
```

   Screenshot:
   Columns were matched by positions.

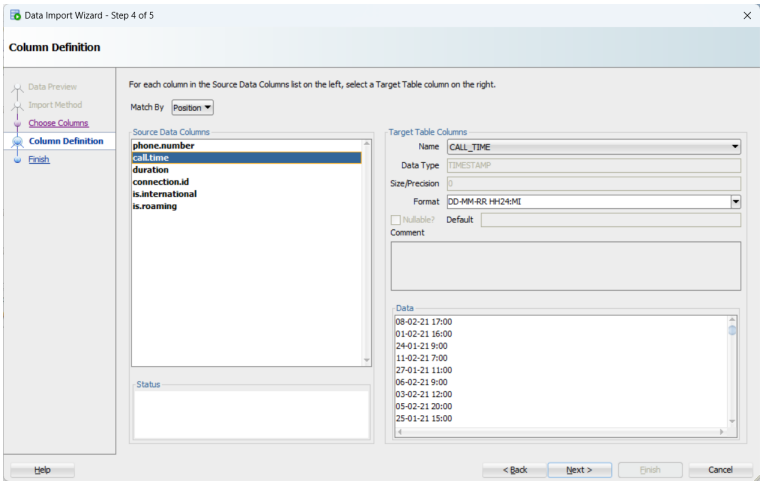


2. `calls.csv`

   Table creation:

```
CREATE TABLE calls (
       phone_number VARCHAR2(13) NOT NULL,
       call_time TIMESTAMP NOT NULL,
       duration NUMBER NOT NULL,
       connection_id CHAR(36) NOT NULL,
       is_international VARCHAR(5),
       is_roaming VARCHAR(5),
       CONSTRAINT pk_calls PRIMARY KEY (connection_id)
);
```

Screenshot:
Columns were matched by positions. For the Target Table Column `call_time`, the format was manually set to `DD-MM-RR HH24:MI`.

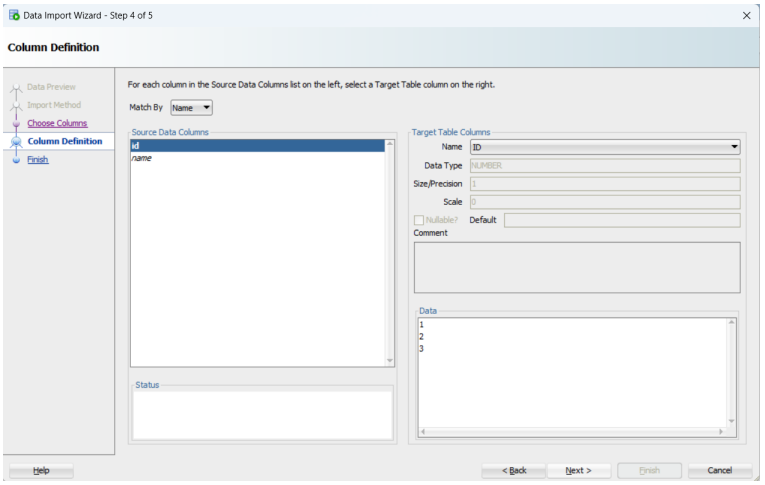


3. `contract_plans.csv`

Table creation:

```
CREATE TABLE contract_plans (
       id NUMBER(1) NOT NULL,
       name VARCHAR2(12) NOT NULL,
       CONSTRAINT pk_contract_plans PRIMARY KEY (id)
);
```

Screenshot:
Columns were matched by name.

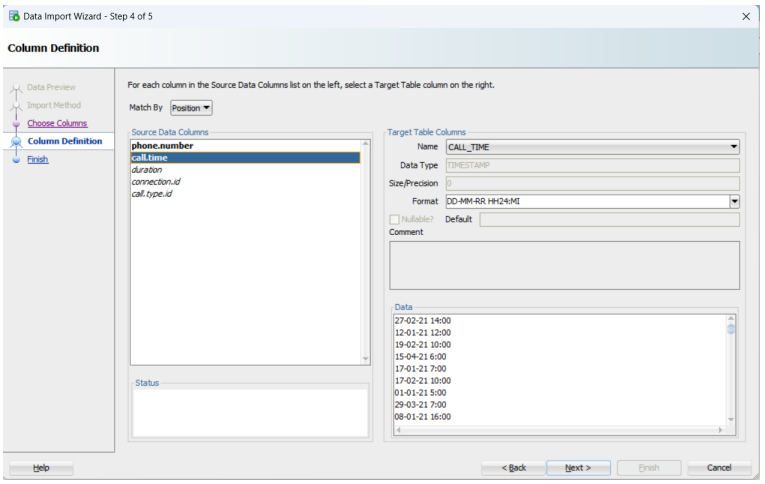


4. `customer_service.csv`

Table creation:

```
CREATE TABLE customer_service (
       phone_number VARCHAR2(13) NOT NULL,
       call_time TIMESTAMP NOT NULL,
       duration NUMBER NOT NULL,
       connection_id CHAR(36) NOT NULL,
       call_type_id NUMBER(1),
       CONSTRAINT pk_customer_service PRIMARY KEY (connection_id)
);
```

Screenshots:
Columns were matched by positions. For the Target Table Column `call_time`, the format was manually set to `DD-MM-RR HH24:MI`.

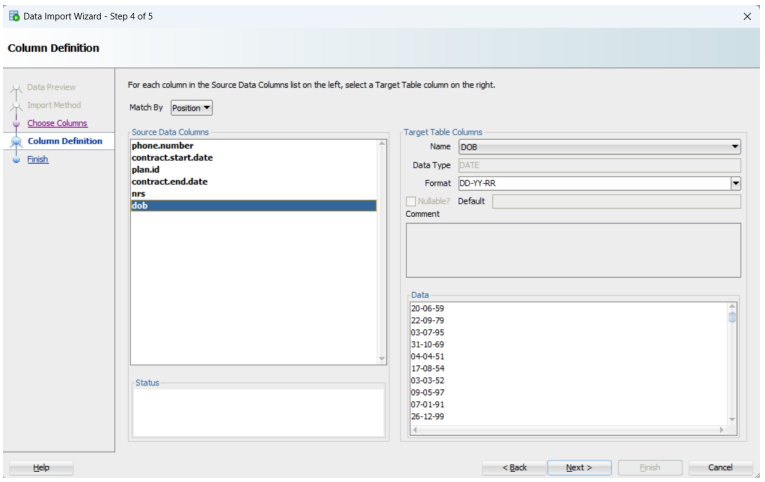


5. `customers.csv`

Table creation:

```
CREATE TABLE customers (
       phone_number VARCHAR2(13) NOT NULL,
       contract_start_date DATE NOT NULL,
       plan_id NUMBER(1) NOT NULL,
       contract_end_date DATE,
       nrs VARCHAR2(2) NOT NULL,
       dob DATE NOT NULL
);
```

Screenshot:
Columns were matched by positions. For the Target Table Columns `contract_start_date`, `contract_end_date`, and `dob`, the formats were manually set to `DD-MM-RR`.

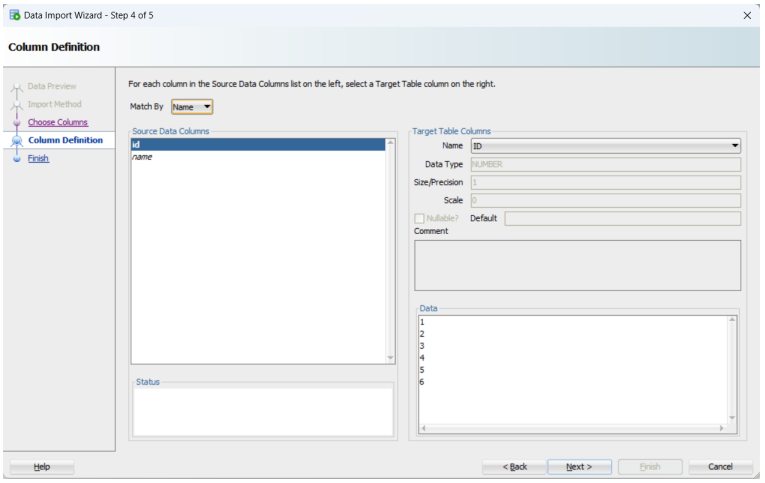


6. `rate_types.csv`

Table creation:

```
CREATE TABLE rate_types (
       id NUMBER(1) NOT NULL,
       name VARCHAR2(16) NOT NULL,
       CONSTRAINT pk_rate_types PRIMARY KEY (id)
);
```

Screenshot:
Columns were matched by name.

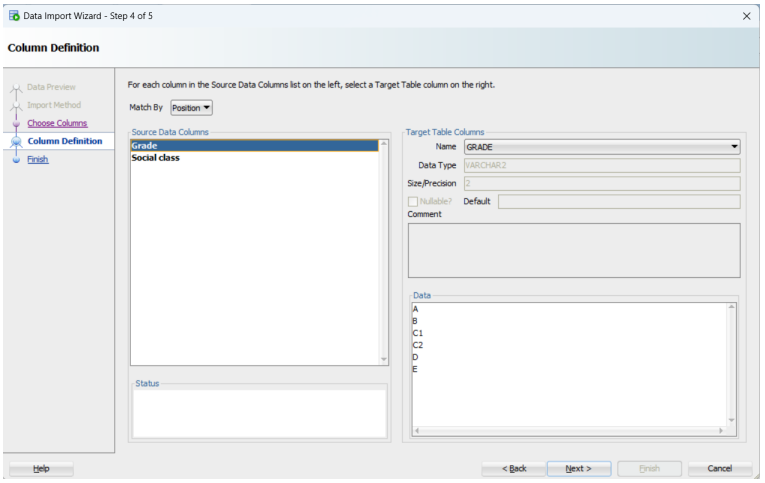


7. `social_grade.csv`

Table creation:

```
CREATE TABLE social_grade (
  grade VARCHAR2(2) NOT NULL,
  social_class VARCHAR2(21) NOT NULL,
  CONSTRAINT pk_social_grade PRIMARY KEY (grade)
);
```

Screenshot:
Columns were matched by positions.

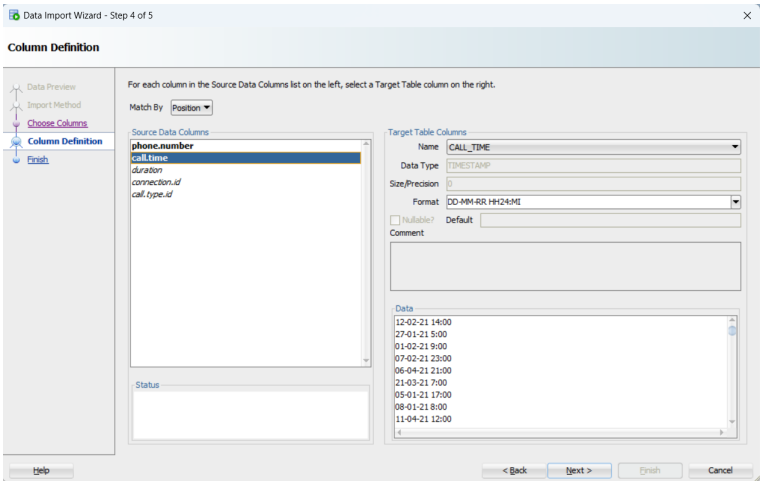


8. `voicemails.csv`

Table creation:

```
CREATE TABLE voicemails (
       phone_number VARCHAR2(13) NOT NULL,
       call_time TIMESTAMP NOT NULL,
       duration NUMBER NOT NULL,
       connection_id CHAR(36) NOT NULL,
       call_type_id NUMBER(1),
       CONSTRAINT pk_voicemails PRIMARY KEY (connection_id)
);
```

Screenshot:
Columns were matched by positions. For the Target Table Column `call_time`, the format was manually set to `DD-MM-RR HH24:MI`.



b) Fact Table Design

- What is the Business Process?

  Create a data warehouse to investigate customer churn to help customer service agents build a better picture of the customers they speak to on the phone. Some data are captured from calls made, others are captured through customer contract processes.

- What is the Grain?

  The finest meaningful grain possible is one row per call. It captures information on both customers and revenue. Other potential candidates are

  - Per customer (e.g. total value spent per customer throughout the relationship). This could be done, but we may lose information on the calls that made up the revenue per customer.

  - Per month or per year (e.g. total revenue per month). They would be of interest to the business, but too rough as grains. We could use a more granular row level and calculate them from the fact table.

  - Per contract plan (e.g. total revenue per contract plan). This is also possible but at risk of losing information on the calls that made up the revenue per contract plan.

- What are the Dimensions?

  Below are the different entities that make up the fact table. Plan Dimension and Social Grade Dimension have been included in Rate Dimension and Customer Dimension respectively since they are closely related.

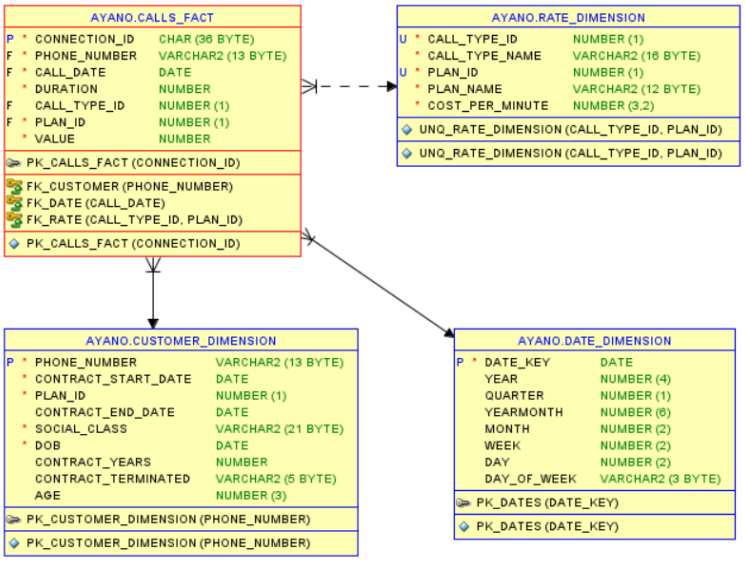
  - Date Dimension
  - Customer Dimension
  - Rate Dimension
  - Plan Dimension (Included in the Rate Dimension)
  - Social Grade Dimension (Included in the Customer Dimension)
  - Connection ID (Not a dimension but required as a Primary Key)

- What are the Facts?

  Below are the facts that correspond to the row per call grain that are generally numeric. The value of the call is derived from `duration` and `cost_per_minute`.

  - Connection ID (Required as a Primary Key)
  - Phone Number (Foreign Key for Customer Dimension)
  - Call Date (Foreign Key for Date Dimension)
  - Duration
  - Call Type ID (Composite Foreign Key for Rate Dimension with Plan ID)
  - Plan ID (Composite Foreign Key for Rate Dimension with Call Type ID)
  - Value of the call (Derived attribute)

  Star Schema:



- Types of Queries for the Fact Table

  - What is the historical value of a customer?
  - What is the value of a customer from the most recent month?
  - What is a customer's profile in age, social class, and the length of contract in years? What plan are they on, and are they still contracted with the company?
  - What types of calls does a customer make, their average duration, and total value per call type?
  - How many customer service calls does a customer make per month?
  - What is a customer's usage per month in the number of calls, total duration, and total value? Have they changed over time?
  - Which call plans bring in the most revenue overall?

- Which call plans bring in the most revenue in the most recent month (April 2021)?

c) Transforming Data into Data Warehouse

The SQL script used to transform the data from the imported tables into the final data-warehouse format is included in `D21125676_CA2.sql`. Few notes on the transformation below.

- Duplicates

  There are two rows from `customers.csv` that have the same phone number. The row with the later `contract_start_date` has been removed from the `customer_dimension` table since its `contract_start_date` is later than the `contract_end_date`.

	PHONE_NUMBER	CONTRACT_START_DATE	PLAN_ID	CONTRACT_END_DATE	NRS	DOB
1	01 495 7529	23-JAN-16	1	01-MAR-21	B	24-MAR-98
2	01 495 7529	24-MAR-20	1	01-MAR-21	B	20-NOV-53

- Errors in contract dates

  There are 32 remaining rows where `contract_start_date` is later than `contract_end_date`. They have been removed from the `customer_dimension` table. The screenshot below shows a few examples.

	PHONE_NUMBER	CONTRACT_START_DATE	PLAN_ID	CONTRACT_END_DATE	SOCIAL_CLASS	DOB
1	01 299 4078	18-APR-21	2	01-MAR-21	Working class	25-MAR-92
2	094 412 5291	12-APR-21	2	01-MAR-21	Working class	13-NOV-65
3	068 527 7919	08-APR-21	3	01-MAR-21	Lower middle class	29-AUG-64
4	069 753 3776	22-MAR-21	2	01-FEB-21	Lower middle class	27-MAY-59
5	022 970 4317	12-APR-21	2	01-MAR-21	Skilled working class	09-MAR-90
6	056 040 3415	30-MAR-21	2	01-FEB-21	Skilled working class	10-MAY-61
7	049 189 5110	16-APR-21	2	01-FEB-21	Lower middle class	13-FEB-98
8	024 856 4613	15-APR-21	3	01-MAR-21	Lower middle class	05-MAR-62
9	053 011 6713	06-APR-21	3	01-FEB-21	Middle middle class	09-OCT-67
10	042 701 7565	20-MAR-21	3	01-FEB-21	Middle middle class	21-OCT-76

  689 rows for the calls made from these phone numbers are also removed from the `calls_fact` table.

- Calls made earlier than Contract Start Date

  There are 5,690 rows for calls where the `call_date` is earlier than `contract_start_date`. They have been removed from the `calls_fact` table.

Section B: Data Analysis and Queries Using SQL

a) Data Analysis

Descriptive statistics for `calls_fact` table

- `call_date`
  The data is from 2021-01-01 to 2021-04-29.

	MINIMUM_DATE	MAXIMUM_DATE	MODE_DATE
1	01-JAN-21	29-APR-21	18-JAN-21

- `duration`
  The shortest call is 0.06 seconds, and the longest call is 1 hour 19 minutes. The median duration for people to stay on a call is 11 minutes and 17 seconds.

	MINIMUM_DURATION	MAXIMUM_DURATION	MEAN_DURATOIN	MEDIAN_DURATION
1	0.06	4744.99	906.4	677.93

- `call_type_id`
  The most common call type is off-peak, followed by peak and voice mail.

	CALL_TYPE_ID	CALL_TYPE_NAME	COUNT_OF_CALL_TYPE_ID	PERCENTAGE
1	2	off-peak	53177	27.18%
2	1	peak	51576	26.37%
3	5	voice mail	29080	14.87%
4	4	roaming	27983	14.31%
5	3	international	23778	12.16%
6	6	customer service	10020	5.12%

- `plan_id`
  Calls are almost equally spread between the three plans.

	PLAN_ID	PLAN_NAME	COUNT_OF_PLAN_ID	PERCENTAGE
1	2	off peak	68157	34.84%
2	1	standard	64670	33.06%
3	3	cosmopolitan	62787	32.1%

- `value`
  The minimum value of a call is 0 since customer service calls are not charged. The maximum value is 2,594.89 and the median value is 74.3.

	MINIMUM_VALUE	MAXIMUM_VALUE	MEAN_VALUE	MEDIAN_VALUE
1	0	2594.89	146.89	74.3

Descriptive statistics for `customer_dimension` table

- `plan_id`
  `customer_dimension` has a very similar percentage distribution as `calls_fact` when grouped by plans.

	PLAN_ID	PLAN_NAME	COUNT_OF_PLAN_ID	PERCENTAGE
1	2	off peak	10170	34.13%
2	1	standard	9990	33.52%
3	3	cosmopolitan	9642	32.35%

- `social_class`
  The most common social class is the lower middle class, followed by the middle middle class.

	SOCIAL_CLASS	COUNT_OF_SOCIAL_CLASS	PERCENTAGE
1	Lower middle class	1406	28.31%
2	Middle middle class	1183	23.82%
3	Skilled working class	952	19.17%
4	Working class	753	15.16%
5	Non-working	474	9.54%
6	Upper middle class	199	4.01%

- `contract_years`
  The minimum length of a contract is 0 years, and the maximum is 7 years till today (`sysdate`). People stay on a contract for 3.76 years on average.

	MINIMUM_CONTRACT_YEARS	MAXIMUM_CONTRACT_YEARS	MEAN_CONTRACT_YEARS	MEDIAN_CONTRACT_YEARS
1	0	7	3.76	3.8

- `contract_terminated`
  31% of customers have terminated their contracts, while 68.17% are still with the company.

	CONTRACT_TERMINATED	COUNT_OF_CONTRACT_TERMINATED	PERCENTAGE
1	FALSE	3386	68.17%
2	TRUE	1581	31.83%

- `age`
  The youngest customer today (till `sysdate`) is 22 years old, and the oldest is 72 years old. The average customer's age is 47.77 years old.

	MINIMUM_AGE	MAXIMUM_AGE	MEAN_AGE	MEDIAN...
1	22	72	47.77	48

Cross-table analysis

- Total value per call type and their percentage of overall revenue

  Almost 45% of overall revenue comes from roaming calls which have high cost-per-minute values, followed by peak and international calls. Customer service calls have no charge, hence the total value of 0.

	CALL_TYPE_NAME	TOTAL_VALUE	PERCENTAGE_REVENUE
1	roaming	12920107.73	44.97%
2	peak	5136457.26	17.88%
3	international	4709012.22	16.39%
4	off-peak	3327506.6	11.58%
5	voice mail	2640389.39	9.19%
6	customer service	0	0%

- Average age per social class and total value spent

  Average age is around 48 across all social classes. The distribution of `percentage_revenue` is almost identical to the distribution of social class seen in the descriptive statistics for the `customer_dimension` table. The lower middle class makes up 28.6% of revenue followed by the middle middle class accounting for 23.31%.

	SOCIAL_CLASS	MEAN_AGE	TOTAL_VALUE	PERCENTAGE_REVENUE
1	Lower middle class	47.9	8219162.77	28.6%
2	Middle middle class	47.73	6696985.38	23.31%
3	Skilled working class	47.95	5391525.78	18.76%
4	Working class	48	4395967.21	15.3%
5	Non-working	48.66	2797665.31	9.74%
6	Upper middle class	48.37	1232166.74	4.29%

- The most popular plan per age bracket

  Age brackets were defined as

```
CASE WHEN age BETWEEN 21 AND 40 THEN 'adult'
     WHEN age BETWEEN 41 AND 60 THEN 'middle age adult'
     ELSE 'older adult' END
```

  Standard plan is the most popular among people between age 21 and 40. For middle age adults and older adults, the off peak plan is the most popular.

	AGE_BRACKETS	MOST_POPUALR_PLAN	COUNT
1	adult	standard	3660
2	middle age adult	off peak	4080
3	older adult	off peak	2562

b) SQL Queries

The SQL queries in this section are interactive scripts that ask for a customer's phone number.

- What is the historical value of a customer, with a percent rank compared to other customers?

  Example result with phone number `024 296 1755`. Their historical value is 3,556.2 which is in the top 62.96% of all customers.

	PHONE_NUMBER	TOTAL_VALUE	PERCENT_RANK
1	024 296 1755	3556.2	61.57%

- What is the value of a customer from the most recent month (April 2021), with a percent rank compared to other customers?

  Example result with phone number `01 765 1683`. Their value from April 2021 is 2,359.51 which is in the top 24.71% of all customers.

	PHONE_NUMBER	APR21_TOTAL_VALUE	PERCENT_RANK
1	01 765 1683	2359.51	24.53%

- What is a customer's profile in age, social class, and the length of contract in years? What plan are they on, and are they still contracted with the company?

  Example result with phone number `01 895 3095`. They are a 68 years-old upper middle class customer on the off peak plan. They are currently in a contract with the company, and they have been for 3.4 years.

	PHONE_NUMBER	PLAN_NAME	AGE	SOCIAL_CLASS	CONTRACT_YEARS	CONTRACT_TERMINATED
1	01 895 3095	off peak	68	Upper middle class	3.4	FALSE

- What types of calls does a customer make, their average duration, and total value per call type?

  Example result with phone number `01 117 6176`, they make almost as many off-peak calls as they do peak calls. They also spend the highest value on roaming calls.

	PHONE_NUMBER	CALL_TYPE_NAME	CALL_TYPE_COUNT	AVG_DURATION	TOTAL_VALUE
1	01 117 6176	peak	23	805.34	2222.75
2	01 117 6176	off-peak	21	802.54	505.6
3	01 117 6176	voice mail	12	1225.39	1470.47
4	01 117 6176	international	7	883.63	1608.21
5	01 117 6176	customer service	6	1338.3	0
6	01 117 6176	roaming	6	745.81	2550.68

- How many customer service calls does a customer make per month?

  Example result with phone number `01 979 5106`. They tend to call customer service once a month, except for April 2021 when they called twice.

	PHONE_NUMBER	YEAR	MONTH	CS_CALLS
1	01 979 5106	2021	1	1
2	01 979 5106	2021	2	1
3	01 979 5106	2021	3	1
4	01 979 5106	2021	4	2

- What are the customer's monthly usages in value? Have they changed over time in three-months moving average?

  Example result with phone number `056 760 3848`, their usage was high in March 2021 compared to other months, the moving average is increasing over time.

	PHONE_NUMBER	YEAR	MONTH	MONTHLY_VALUE	MOVING_AVERAGE
1	056 760 3848	2021	1	2304.14	2304.14
2	056 760 3848	2021	2	3616.82	2960.48
3	056 760 3848	2021	3	3996.2	3305.72
4	056 760 3848	2021	4	3722.55	3409.93

- Which call plans bring in the most revenue overall? What are their percentages of overall revenue?

  The standard plan brought in the most revenue historically, making up 36.29% of overall revenue, followed by off peak and cosmopolitan.

	PLAN_NAME	TOTAL_VALUE	PERCENTAGE_REVENUE
1	standard	10427376.44	36.29%
2	off peak	10357511.47	36.05%
3	cosmopolitan	7948585.29	27.66%

- Which call plans bring in the most revenue in the most recent month (April 2021)? What are their percentages of overall revenue?

  The standard plan brought in the most revenue in April 2021, making up 36.26% of overall revenue, followed by off peak and cosmopolitan.

	PLAN_NAME	TOTAL_VALUE	APR21_PERCENTAGE_REVENUE
1	off peak	2150701.84	36.3%
2	standard	2143835.2	36.18%
3	cosmopolitan	1630295.43	27.52%

Section C: Machine Learning using SQL

a) Case Table and Preparations

- Case table

  The aim is to produce a churn model which will predict customers who are likely to churn this month. Since the model will be trying to answer if a customer will churn monthly, each row in the case table represents information listed below per customer per month.

  - Concatenated string of `phone_number` and `year_month` as a unique ID
  - Number of calls
  - Total duration of all calls
  - Value spent
  - Number of peak calls
  - Number of off peak calls
  - Number of international calls
  - Number of roaming calls
  - Number of voicemail calls
  - Number of customer service calls
  - Plan ID
  - Social class
  - Age
  - If the customer churned in the following month

- Target level imbalance

  As commonly seen in customer churn datasets, there is a dominant target level of 'no churn' (`next_month_churn = 0`) which makes up 90.34% of the dataset. This makes it likely for any models to be biased towards predicting 'no churn'.

	NEXT_MONTH_CHURN	COUNT	PERCENTAGE
1	0	14595	90.34%
2	1	1561	9.66%

  To improve this, Support Vector Machine (SVM) Classification and Generalised Linear Model Classification were chosen for the availability to specify class weights. We will set a higher class weight to 'churn' (`next_month_churn = 1`) compared to 'no churn' in order to influence the training phase of the process.

We will also use the Confusion Matrix and F1 score to measure the performance of the models instead of accuracy. This is to maximise the possibility of the model predicting true positives, which is in line with the business objective to predict customers who are likely to churn in a given month.

- Train test split

  Since we have a large enough dataset of 16,156 rows in the `case_table`, it was split into the `train_table` containing 80% of randomly selected rows, and the `test_table` containing the remaining 20%. The cross-validation method would have been ideal given the target level imbalance, but we have continued with the train test split after confirming that the target level distributions for both train and test tables are similar to the distribution seen in the `case_table`.

	TABLE_NAME	NEXT_MONTH_CHURN	COUNT	PERCENTAGE
1	train_table	0	11628	90.29%
2	train_table	1	1251	9.71%
3	test_table	0	2967	90.54%
4	test_table	1	310	9.46%

b) Machine Learning Models

- Support Vector Machine (SVM)

  Class weights table was created with `class_weight` of 0.099 for ‘no churn’ and 0.901 for ‘churn’ to bias the model against the target level imbalance.

```
CREATE TABLE svm_class_wt (
       target_value NUMBER,
       class_weight NUMBER);
INSERT INTO svm_class_wt VALUES (0, 0.099);
INSERT INTO svm_class_wt VALUES (1, 0.901);
```

  With `algo_support_vector_machines` specified and `prep_auto_on`, the model was able to predict 122 true positives with an F1 score of .1811. Considering the business objective, this is an improvement on non-weighted models which would predict all rows as ‘no churn’ to get a 90.54% accuracy.

	ACTUAL_TARGET_VALUE	PREDICTED_TARGET_VALUE	VALUE
1	0	1	915
2	0	0	2052
3	1	0	188
4	1	1	122

  The number of false negatives is 188, which is the number of customers who are going to churn next month that are not highlighted to the customer service agents. Reducing this number as well as increasing the true positives would be an improvement on this model.

  The number of false positives is 915, meaning the customer service agents will be falsely highlighted to address 915 customers who are not going to churn. Since the goal is to enable them to address customers who are about to churn, we’re going to consider false positives not as significant a measure as others unless the number becomes extremely high.

- Generalised Linear Model

  Class weights table was created with `class_weight` of 0.09 for ‘no churn’, and 0.91 for ‘churn’ to bias the model against the target level imbalance.

```
CREATE TABLE glm_class_wt (
       target_value NUMBER,
       class_weight NUMBER);
INSERT INTO glm_class_wt VALUES (0, 0.09);
INSERT INTO glm_class_wt VALUES (1, 0.91);
```

  With `algo_generalized_linear_model` specified and `prep_auto_on`, the model was able to predict 212 true positives with an F1 score of .1926. This would be an improvement on the SVM model since we can predict 90 more true positives.

	ACTUAL_TARGET_VALUE	PREDICTED_TARGET_VALUE	VALUE
1	0	1	1679
2	0	0	1288
3	1	1	212
4	1	0	98

  The number of false negatives has decreased to 98, which is also an improvement since there will be 90 fewer customers who are falsely highlighted as ‘no churn’ compared to the SVM model.

  The number of false positives however is 1,679, which is higher than the true negatives of 1,288. We have to consider that there are also costs to customer service agents reaching out to customers to try and stop them from churning, either individually or as automated campaigns. At this point, it would be a decision up to the business to decide if this number is acceptable.