LONDON
METROPOLITAN
UNIVERSITY

islington college
(इस्लिङ्टन कलेज)

CC5051NI Databases

50% Individual Coursework

2022 Autumn

Student Name: Abhijan Basyal

London Met ID: 22015639

College ID: np01cp4s220138

Assignment Submission Date: 5th January 2023

Assignment Due Date: 5thJanuary 2023

Word Count: 1242

Table of Contents

Abhijan Basyal

Abhijan Basyal

TABLE OF FIGURES

Abhijan Basyal

TABLE OF TABLES

Abhijan Basyal

1. INTRODUCTION

This task involves building a database model for the Pirke transportation firm so that its clients can reserve a car and driver for transportation in a way like a taxi and take use of services like food delivery and package delivery, among others. Canada-based Pirke is a technology-driven logistics and transportation platform. Since its establishment in 2004, Pirke has grown to be the nation's top ride-hailing service, providing millions of customers with dependable transportation. It has changed over the past few years and now provides a wide range of services, such as food delivery, motorbike and automobile trips, and courier services. This transportation and logistics have changed because of Pirke, making them more effective, available, and inexpensive for all.

This report includes the database system used by the Pirke transportation and delivery service to keep track of all its drivers and client. This report also includes information about the business roles, the project's scope, how to implement data correctly, how to properly set up an ERD diagram, and how to perform those processes.

1.1 AIMS AND OBJECTIVES

The main aims of this project are to create a database system for Pirke Transportation that can manage all its clients and drivers. This system will add a customer reward points system to the client and allow customers to accrue points when they use the service and drivers to work full or part time schedules. Along with the customer incentive points, each customer should also receive a charge for each journey taken or service used.

The business objective to achieve its aims are as follows:

- ➢ To store the details information about the customer, vehicle, and driver.
- ➢ To keep the track of drivers and customers.
- ➢ To keep the records off the service types assigned by the drivers.
- ➢ To keep the details of the half and full type of the drivers.
- ➢ To store the record of the customer rewards points.
- ➢ To check the record for the customer who uses this system most.
- ➢ To store the annual profit, current revenue, and total revenue.

Abhijan Basyal

1.2 CURRENT BUSINESS ACTIVITIES AND OPERATION

Online businesses like Pirke Transportation Company offer services including booking, meal delivery, and courier service. According on the customer's preference, this transportation service allows for either automobile or motorcycle reservations. This business gives prompt, efficient service. This business offers the right career opportunities for the driver, food delivery, and other positions. Here, the database system lists all the driver and customer's records. Each customer receives rewards points based on how frequently the system is utilized and receives the corresponding discount percentage when using the company's services.

1.3 BUSINESS RULES

Each business has its own set of guidelines that both the client and the business must abide by. Without business regulations, the organization will not be positioned for growth and there will be several problems, including a risk of financial loss and reputational damage. Therefore, the firm Pirke Transportation also has a set of rules, such as:

For Drivers:

➢ A driver may drive many vehicles, but each vehicle and a service are used by only one driver at a time.
➢ A driver writes a single invoice for each service he provides.
➢ A drivers must possess a valid driver's license and maintain a clean driving record.
➢ Drivers must maintain the vehicle.
➢ Driver must provide 10 percent of profit to the company.
➢ Drivers must be polite and professional when interacting with the customers.

For Customers:

➢ Customer must have an account to use this service.
➢ Customer must agree to the terms of services and privacy policy before using the service.
➢ Once the customer books the service, they cannot cancel the service.

Abhijan Basyal

- ➢ Service ticket is issued once the customer books the vehicle and the service and will include details like driver name, type of service, total charge, estimated duration of the destination.
- ➢ Customer must enter a valid pick up and drop off location within the service area.
- ➢ According to the customer category, the discount percent is granted like for staff 40 percent discount is granted.
- ➢ Customer can use the rewards points and gain discounts percent according to the rewards points.

1.4 ASSUMPTION

- ➢ Every customer will get a bill for each ride, or the service used.
- ➢ In comparison to motorcycle and taxi services, food delivery on bikes and in taxis is included where taxi is more expensive.
- ➢ Reward point is granted according to the uses of the system.
- ➢ Discount percent is given according to the customer category or from the reward point.

Abhijan Basyal

2. ENTITIES AND ATTRIBUTES

A person, place, or thing that exists independently and has a unique identity is referred to as an entity. An organization or business that has been lawfully established and recognized as a distinct legal entity from its owners or shareholders is referred to as a "entity" in both commerce and law (IBM Corporation, 2022). Characteristics or features used to describe something, or someone are called attributes. When referring to data, attributes refer to particular facts about an entity (IBM Corporation, 2022). After the right normalization and processing, the final entities and attributes will be generated as these initial entities together with their attributes, which are as follows. We can initially establish an entity and its attributes from the business rules, but they are not fixed ones which are as follows:

2.1 CUSTOMER

Eight characteristics, including customer ID, customer name, customer address, customer email, phone number, customer category, reward point, and discount percent, are contained in the customer entity and display all variety of functions to the customer in the database system. The attributes of the customer entity are described in the table below along with its data type, constraints, and description.

Table 1 Customer table

ATTRIBUTES	DATA TYPE	CONSTRAINTS	DESCRIPTION
customer_ID	varchar2(10)	primary key, unique	It stores the unique ID of the customer.
customer_name	varchar2(35)	not null	It stores the name of the customer.
customer_address	varchar2(15)	not null	It stores the location of the customer.
customer_email	varchar2(30)	not null	It stores the email of the customer.

Abhijan Basyal

phone_number	varchar2(10)	not null	It stores the phone number of the customer.
customer_category	varchar2(10)	not null	It stores and shows the category of the customer.
reward_point	varchar2(10)	not null	It stores the reward point gain by the customer.
discount_percent	varchar2(10)	null	It stores the discount percent for the customer.

2.2 VEHICLE

Twelve attributes, including vehicle ID, vehicle name, vehicle type, vehicle model, vehicle rate, fuel type, driver ID, driver name, driver address, driver number, driver job type, and driver earnings, are contained in the vehicle entity and display all information pertaining to the vehicle and the driver who drives that vehicle, in the database system. The attributes of the vehicle entity are described below in the table along with its data type, constraints, and description.

Table 2 vehicle table

ATTRIBUTES	DATA TYPE	CONSTRAINTS	DESCRIPTION
vehicle_ID	varchar2(10)	primary key, unique	It uniquely identifies the ID of the vehicle.
vehicle_name	varchar2(35)	not null	It stores the name of the vehicle.
vehicle_model	varchar2(30)	not null	It stores the model of the vehicle.

vehicle_type	varchar2(30)	not null	It stores the type of the vehicle whether it is motorcycle or car.
fuel_type	varchar2(10)	not null	It stores the type of the fuel used in the vehicle.
vehicle_price	number (9,2)	not null	It stores the price of the particular vehicle.
driver_ID	varchar2(10)	not null	It stores id of the driver.
driver_name	varchar2(35)	Not null	It stores the name of the driver.
driver_address	varchar2(15)	not null	It stores the address of the driver.
driver_number	varchar2(10)	not null	It stores the phone number of the driver.
driver_jobtype	varchar2(10)	not null	It stores the information of the job type whether it is full or half.
driver_earnings	number (9,2)	not null	It stores the earning of the driver in a month.

2.3 BILL

Seven characteristics, including bill ID, issued date, delivery location, service ID, service name, service cost and total cost are contained in the bill entity and display all information pertaining to the bill and the services in the database system. The customer ID and vehicle ID are also included in the bill entity as a foreign key therefore total ten attribute is included in the bill. The attributes of the bill entity are described in the table below, along with the data type, restrictions, and description.

Abhijan Basyal

Table 3 bill table

ATTRIBUTES	DATA TYPE	CONSTRAINTS	DESCRIPTION
bill_ID	varchar2(10)	primary key, unique	It stores the unique id of the bill.
issued_date	date	not null	It stores the date of the bill issued.
delivery_location	varchar2(10)	not null	It stores the location of the where the delivery was taken.
service_ID	varchar2(10)	not null	It stores the service ID taken by the customer.
service_name	varchar2(35)	not null	It stores the service name taken by the customer.
service_cost	number (9,2)	not null	It stores the data for the cost of the service given to the customer.
total_cost	number (9,2)	not null	It stores the data of the total cost of the customer.
customer_ID	varchar2(10)	not null, foreign key	It gives the data of the customer who have taken the bill.
vehicle_ID	varchar2(10)	not null, foreign key	It gives the data of the vehicle for the customer who have taken service from it.

Abhijan Basyal

3. ENTITY RELATONSHIP DIAGRAM

An entity relationship diagram (ERD), also known as an entity relationship model, is a graphical representation that depicts relationships among people, objects, places, concepts, or events within an information technology (IT) system. An ERD uses data modelling techniques that can help define business processes and serve as the foundation for a relational database (Biscobing, 2019).

3.1 INITIAL ERD

CUSTOMER	
PK	customer_ID
	customer_name
	customer_address
	customer_email
	phone_number
	customer_category
	reward_point
	discount_percent

BILL	
PK	bill_ID
	issued_date
	delievery_location
	service_ID
	service_name
	service_cost
	total_cost
FK	customer_ID
FK	vehicle_ID

VEHICLE	
PK	vehicle_ID
	vehicle_name
	vehicle_model
	vehicle_type
	fuel_type
	vehicle_price
	driver_ID
	driver_name
	driver_address
	driver_number
	driver_jobtype
	driver_earnings

Figure 1 initial ERD

Abhijan Basyal
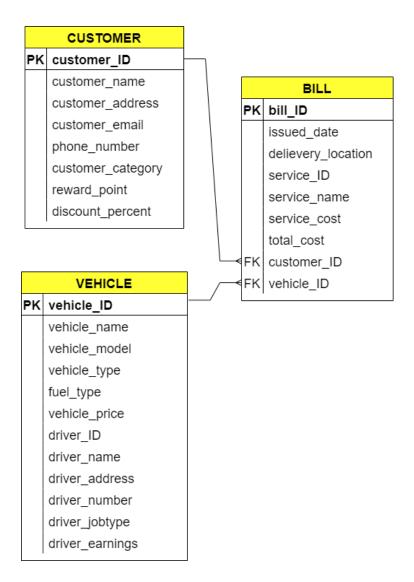
4. NORMALIZATION

Normalization is the process of arranging data into tables so that queries are always answered in a clear and intended manner. Such normalization is fundamental to the theory of relational databases. It frequently leads to the establishment of new tables and may have the consequence of duplicating data in the database (Rouse & Vaughan, 2019). The process of normalizing a database involves setting it up to reduce dependencies and redundancies. A database is often divided into two or more tables during normalization, and the linkages between the tables are established. The goal is to separate the data so that changes to a field can be performed in just one table and then propagate to the other tables in the database using the stated relationships (Rouse & Vaughan, 2019).

For normalization there are following steps such as:

4.1 Un-Normalised Form (UNF)

Un-Normalised Form is the first steps of the normalised form in which we should include all the attributes of all entity with repeating groups. Here, we should first write all the attribute from above initial ERD.

Customer = (customer_ID(PK), customer_name, customer_address, customer_email, phone_number, customer_category, reward_point, discount_percent,{bill_ID, issued_date, delivery_location, service_ID, service_name, service_cost, total_cost, vehicle_ID, vehicle_name, vehicle_model, vehicle_type, fuel_type, vehicle_price, driver_ID, driver_name, driver_address, driver_number, driver_jobtype, driver_earnings})

The entity in the above is called Customer, and the attribute's unique identifier is customer_ID which is denoted by the PK (i.e., Primary Key). All the attributes were extracted from the initial ERD, and the repeated group was separated and retained in the curly bracket. Now, going through the next step of normalization which is First Normal Form (1NF).

Abhijan Basyal

4.2 First Normal Form (1NF)

First Normal Form is the second steps of the normalization where the repeating groups should be removed to separate relation and there will be two groups with two entity one for repeating and one for non-repeating groups where primary key of non-repeating group should be the foreign key of the repeating group.

Customer = customer_ID(PK), customer_name, customer_address, customer_email, phone_number, customer_category, reward_point, discount_percent

Bill = bill_ID(PK), issued_date, delivery_location, service_ID, service_name, service_cost, total_cost, vehicle_ID, vehicle_name, vehicle_model, vehicle_type, fuel_type, vehicle_price, driver_ID, driver_name, driver_address, driver_number, driver_jobtype, driver_earnings, customer_ID(FK)

The repeating groups and non-repeating groups are divided in the scenario above, where the entity for the repeating group is the customer, and the primary key is the customer_ID, and the entity for the non-repeating group is the bill, and the primary key is the bill_ID. The foreign key is contained in the bill entity as customer_ID which is denoted as FK (i.e., Foreign Key) because, according to the 1NF, the primary key of a non-repeating group is the foreign key of a repeating group. Now, going through the next step of normalization which is Second Normal Form (2NF).

Abhijan Basyal

4.3 Second Normal Form (2NF)

Second Normal Form is the third steps of the normalization where all the non-key attributes must be fully functional dependent on Primary key and not on only a part (portion) of Primary Key. A separate Relation should be created for any attributes that are completely dependent on just one component of the Composite identification. Prevents situations when each row contains single-valued information regarding many objects. Since partial FDs on an identifier duplicate data, they should be avoided.

Customer = customer_ID(PK), customer_name, customer_address, customer_email, phone_number, customer_category, reward_point, discount_percent

Bill = bill_ID(PK), issued_date, delivery_location, service_ID, service_name, service_cost, total_cost, vehicle_ID, vehicle_name, vehicle_model, vehicle_type, fuel_type, vehicle_price, driver_ID, driver_name, driver_address, driver_number, driver_jobtype, driver_earnings, customer_ID(FK)

Customer_Bill = customer_ID, bill_ID

There is no composite primary key in the Customer entity. As a result, there is no possibility of partial dependency, and there is also no composite primary key for the bill entity, therefore there is no partial dependency in the Bill entity. Then, Customer_Bill, an entity is created which keeps the primary key of both entity which is customer_ID and bill_ID. Now, going through the next step of normalization which is Third Normal Form (3NF).

Abhijan Basyal

4.4 Third Normal Form (3NF)

Third Normal Form is the fourth and the final steps of the normalization where there should be no transitive dependencies. For each determinant in a transitive dependency, a separate relation should be created, and determinants primary keys should be made in the respective relation.

Now, separating the transitive dependency from the 2NF

customer_ID =>customer_name, customer_address, customer_email, phone_number, customer_category, reward_point, discount_percent

bill_ID =>service_ID =>service_name, service_cost

bill_ID =>vehicle_ID =>vehicle_name, vehicle_model, vehicle_type, fuel_type, vehicle_price

bill_ID =>driver_ID, driver_name, driver_address, driver_number, driver_jobtype, driver_earnings

Final 3NF

Customer =>customer_ID(PK), customer_name, customer_address, customer_email, phone_number, customer_category, reward_point, discount_percent

Bill =>bill_ID(PK), customer_ID(FK), service_ID(FK), vehicle_ID(FK), driver_ID(FK), issued_date, delivery_location, total_cost

Service => service_ID(PK), service_name, service_cost

Vehicle =>vehicle_ID(PK), vehicle_name, vehicle_model, vehicle_type, fuel_type, vehicle_price

Driver =>driver_ID(PK), driver_name, driver_address, driver_number, driver_jobtype, driver_earnings

Abhijan Basyal

The bill contains every primary key of every other entity as a foreign key on it, so from the above 3NF normalization it states that there are 1 to many relations to the bill by every other entity. In the above all the transitive dependency is separated and is now finalize the proper entity and attribute where there are five entities which are Customer, Bill, Services, Vehicle, and Driver which have relation and where the Customer entity has customer_ID, Bill entity has bill_ID, Service entity has service_ID, Vehicle entity has vehicle_ID and Driver entity has driver_ID as a primary key.

5. FINAL ERD

The final ERD is an entity relationship diagram that has been created after the initial ERD has been properly normalized and has been finalized with the correct attributes and entities.
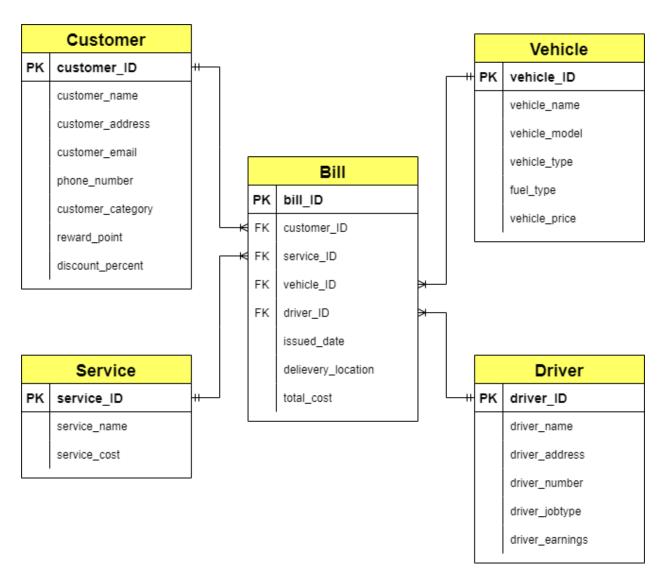

Figure 2 Final ERD

Abhijan Basyal

6. IMPLEMENTATION

In the implementation, a table is made and the normalized data from the appropriate entities is entered into it.

Before implementing the data for the respective entity. Firstly, we must connect to the system in the SQL command line and create the new user named as cw_abhijan by granting the privilege and then connecting to the user cw_abhijan.

```
SQL> conn system/abhijan
Connected.
SQL> create user cw_abhijan identified by abhijan
  2  ;

User created.
```

Figure 3 connecting to the system and creating user

```
SQL> grant connect,resource to cw_abhijan;

Grant succeeded.

SQL> connect cw_abhijan/abhijan;
Connected.
```

Figure 4 granting connection and connecting to created user

Abhijan Basyal

6.1 Customer

Below is a creation of the table of the customer entity, including their attributes and descriptions after normalization:

Table 4 Final customer table

ATTRIBUTES	DATA TYPE	CONSTRAINTS	DESCRIPTION
customer_ID	varchar2(10)	primary key, unique	It stores the unique ID of the customer.
customer_name	varchar2(35)	not null	It stores the name of the customer.
customer_address	varchar2(15)	not null	It stores the location of the customer.
customer_email	varchar2(30)	not null	It stores the email of the customer.
phone_number	varchar2(10)	not null	It stores the phone number of the customer.
customer_category	varchar2(10)	not null	It stores and shows the category of the customer.
reward_point	varchar2(10)	not null	It stores the reward point gain by the customer.
discount_percent	varchar2(10)	null	It stores the discount percent for the customer.

a) Relation

One customer receives numerous bills according to the service; however, one bill only receives one customer, owing to the one(mandatory)-to-many(mandatory) relationships between the customer table and the bill table. The customer table displays the detail information of the customer.

Abhijan Basyal

b) Creating and inserting the values in the table customer

Here, the table customer is created in the SQL command line by using various command and the data is put according to the needs of the assignment.

```
SQL> create table Customer(
  2  customer_ID varchar2(10) primary key,
  3  customer_name varchar2(35) not null,
  4  customer_address varchar2(15) not null,
  5  customer_email varchar2(30) not null,
  6  phone_number varchar2(10) not null,
  7  customer_category varchar2(10) not null,
  8  reward_point varchar2(10) not null,
  9  discount_percent varchar2(10) null);

Table created.

SQL> describe Customer;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 CUSTOMER_ID                               NOT NULL VARCHAR2(10)
 CUSTOMER_NAME                             NOT NULL VARCHAR2(35)
 CUSTOMER_ADDRESS                          NOT NULL VARCHAR2(15)
 CUSTOMER_EMAIL                            NOT NULL VARCHAR2(30)
 PHONE_NUMBER                              NOT NULL VARCHAR2(10)
 CUSTOMER_CATEGORY                         NOT NULL VARCHAR2(10)
 REWARD_POINT                              NOT NULL VARCHAR2(10)
 DISCOUNT_PERCENT                                   VARCHAR2(10)
```

Figure 5 creating and describing the table customer

```
SQL> insert into Customer values(
  2  'c1','abhijan basyal','maitidevi','basyalabhijan@gmail.com','9840041397','staff','123','40');
```

Figure 6 inserting the values in customer table

```
SQL> insert all
  2  into customer values ('c2','anup acharya chhetri','thamel','anupac@gmail.com','9841583459','normal','111','0')
  3  into customer values ('c3','sahish singh thakuri','maitidevi','sahish@gmail.com','9842234567','normal','234','0')
  4  into customer values ('c4','renu tamang','ghattekulo','renutamang@gmail.com','9818870509','staff','98','40')
  5  into customer values ('c5','tayama rai','ghattekulo','raitayama@gmail.com','9857011004','normal','233','0')
  6  into customer values ('c6','sikha singh sijapati','baneshwor','sikhasijapati@gmail.com','9849509433','staff','321','40')
  7  into customer values ('c7','kritan kunwar chhetri','shantinagar','kritankunwar@gmail.com','9841395299','normal','455','0')
  8  select * from dual;

6 rows created.
```

Figure 7 inserting the value in customer table completely

```
CUSTOMER_I CUSTOMER_NAME                  CUSTOMER_ADDRES CUSTOMER_EMAIL            PHONE_NUMB CUSTOMER_C REWARD_POI DISCOUNT_P
---------- ------------------------------ --------------- ------------------------- ---------- ---------- ---------- ----------
c1         abhijan basyal                 maitidevi       basyalabhijan@gmail.com   9840041397 staff      123        40
c2         anup acharya chhetri           thamel          anupac@gmail.com          9841583459 normal     111        0
c3         sahish singh thakuri           maitidevi       sahish@gmail.com          9842234567 normal     234        0
c4         renu tamang                    ghattekulo      renutamang@gmail.com      9818870509 staff      98         40
c5         tayama rai                     ghattekulo      raitayama@gmail.com       9857011004 normal     233        0
c6         sikha singh sijapati           baneshwor       sikhasijapati@gmail.com   9849509433 staff      321        40
c7         kritan kunwar chhetri          shantinagar     kritankunwar@gmail.com    9841395299 normal     455        0

7 rows selected.
```

Figure 8 Customer table with the data

Abhijan Basyal

6.2 Service

The following is a creation of the table of the service entity, including their attributes and descriptions:

Table 5 service table

ATTRIBUTES	DATA TYPE	CONSTRAINTS	DESCRIPTION
service_ID	varchar2(10)	primary key, unique	It stores unique ID of service of the company.
service_name	varchar2(35)	not null	It stores the name of the services.
service_cost	number (9,2)	not null	It stores the cost of the services present in the company.

a) Relation

The relationship between the service table and the bill table is one(mandatory) to many(mandatory), where the many customers use many services, which generates the bridge as bills table, making one service have many bills and one bill have one customer. The service table provides the name of the service and its cost.

Abhijan Basyal

b) Creating and inserting the values in the table service

Here, the table service is created in the SQL command line by using various command and the data is put according to the needs of the assignment.

```
SQL> create table Service(
  2   service_ID varchar2(10) primary key,
  3   service_name varchar2(35) not null,
  4   service_cost number(9,2) not null);

Table created.

SQL> describe Service;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 SERVICE_ID                                NOT NULL VARCHAR2(10)
 SERVICE_NAME                              NOT NULL VARCHAR2(35)
 SERVICE_COST                              NOT NULL NUMBER(9,2)
```

Figure 9 creating and describing service table

```
SQL> insert all
  2   into service values ('s1','bike ride',150.00)
  3   into service values ('s2','taxi ride',350.00)
  4   into service values ('s3','food delivery',200.00)
  5   into service values ('s4','courier service',200.00)
  6   select * from dual;

4 rows created.

SQL> set linesize 500;
SQL> set pagesize 50;
SQL> select * from service;

SERVICE_ID SERVICE_NAME                        SERVICE_COST
---------- ----------------------------------- ------------
s1         bike ride                                    150
s2         taxi ride                                    350
s3         food delivery                                200
s4         courier service                              200
```

Figure 10 inserting the values and showing data of the service table

Abhijan Basyal

6.3 Vehicle

Below is a creation of the table of the vehicle entity with their attributes and descriptions:

Table 6 final vehicle table

ATTRIBUTES	DATA TYPE	CONSTRAINTS	DESCRIPTION
vehicle_ID	varchar2(10)	primary key, unique	It uniquely identifies the ID of the vehicle.
vehicle_name	varchar2(35)	not null	It stores the name of the vehicle.
vehicle_model	varchar2(30)	not null	It stores the model of the vehicle.
vehicle_type	varchar2(30)	not null	It stores the type of the vehicle whether it is motorcycle or car.
fuel_type	varchar2(10)	not null	It stores the type of the fuel used in the vehicle.
vehicle_price	number (9,2)	not null	It stores the price of the vehicle.

a) Relation

One vehicle is used to implement the service for the customer in the vehicle table, which also has a one(mandatory) to many(mandatory) relationships with the bill table. As a result, the vehicle table provides detailed vehicle information regarding which customers receive which services and which vehicles are used to implement those service.

Abhijan Basyal

b) Creating and inserting the values in the table vehicle

Here, the table vehicle is created in the SQL command line by using various command
and the data is put according to the needs of the assignment.

```
SQL> create table Vehicle(
  2  vehicle_ID varchar2(10) primary key,
  3  vehicle_name varchar2(35) not null,
  4  vehicle_model varchar2(30) not null,
  5  vehicle_type varchar2(30) not null,
  6  fuel_type varchar2(10) not null,
  7  vehicle_price number(9,2) not null);

Table created.

SQL> describe Vehicle;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 VEHICLE_ID                                NOT NULL VARCHAR2(10)
 VEHICLE_NAME                              NOT NULL VARCHAR2(35)
 VEHICLE_MODEL                             NOT NULL VARCHAR2(30)
 VEHICLE_TYPE                              NOT NULL VARCHAR2(30)
 FUEL_TYPE                                 NOT NULL VARCHAR2(10)
 VEHICLE_PRICE                             NOT NULL NUMBER(9,2)
```

Figure 11 creating and describing the table vehicle

```
SQL> insert all
  2  into vehicle values ('v1','splender','hero honda','car','petrol',350.00)
  3  into vehicle values ('v2','appache','discover','bike','petrol',150.00)
  4  into vehicle values ('v3','pulsar','bajaj','bike','petrol',150.00)
  5  into vehicle values ('v4','hurricane','lamborgini','car','disel',350.00)
  6  into vehicle values ('v5','alto','suzuki','car','disel',350.00)
  7  into vehicle values ('v6','adventure','bmw','bike','petrol',150.00)
  8  into vehicle values ('v7','bullet','shanks','bike','disel',150.00)
  9  select * from dual;

7 rows created.

SQL> select * from vehicle;

VEHICLE_ID VEHICLE_NAME                    VEHICLE_MODEL                VEHICLE_TYPE                 FUEL_TYPE  VEHICLE_PRICE
---------- ------------------------------- ---------------------------- ---------------------------- ---------- -------------
v1         splender                        hero honda                   car                          petrol             350
v2         appache                         discover                     bike                         petrol             150
v3         pulsar                          bajaj                        bike                         petrol             150
v4         hurricane                       lamborgini                   car                          disel              350
v5         alto                            suzuki                       car                          disel              350
v6         adventure                       bmw                          bike                         petrol             150
v7         bullet                          shanks                       bike                         disel              150

7 rows selected.
```

Figure 12 inserting the values and showing the data of the vehicle table

Abhijan Basyal

6.4 Driver

Below is a creation of the table of the driver entity with their attributes and descriptions:

Table 7 driver table

ATTRIBUTES	DATA TYPE	CONSTRAINTS	DESCRIPTION
driver_ID	varchar2(10)	primary key, unique	It stores the unique ID of the driver.
driver_name	varchar2(35)	Not null	It stores the name of the driver.
driver_address	varchar2(15)	not null	It stores the address of the driver.
driver_number	varchar2(10)	not null	It stores the phone number of the driver.
driver_jobtype	varchar2(10)	not null	It stores the information of the job type whether it is full or half.
driver_earnings	number (9,2)	not null	It stores the earning of the driver in a month.

a) Relation

The driver table and the bill table have a one(mandatory) to many(mandatory) relationships in which one driver provides many bills to many customers while only one bill is generated for one client. To give services to each consumer, a driver also rides in one car and provides the bill for the customer.

Abhijan Basyal

b) Creating and inserting the values in the table driver

Here, the table driver is created in the SQL command line by using various command and the data is put according to the needs of the assignment.

```
SQL> create table Driver(
  2  driver_ID varchar2(10) primary key,
  3  driver_name varchar2(35) not null,
  4  driver_address varchar2(15) not null,
  5  driver_number varchar2(10) not null,
  6  driver_jobtype varchar2(10) not null,
  7  driver_earnings number(9,2) not null);

Table created.

SQL> describe Driver;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 DRIVER_ID                                 NOT NULL VARCHAR2(10)
 DRIVER_NAME                               NOT NULL VARCHAR2(35)
 DRIVER_ADDRESS                            NOT NULL VARCHAR2(15)
 DRIVER_NUMBER                             NOT NULL VARCHAR2(10)
 DRIVER_JOBTYPE                            NOT NULL VARCHAR2(10)
 DRIVER_EARNINGS                           NOT NULL NUMBER(9,2)
```

Figure 13 creating and describing driver table

```
SQL> insert all
  2  into driver values ('d1','gumaney chhetri','balaju','9809876543','full time',80000.00)
  3  into driver values ('d2','ashish pandey','thamel','9854176543','part time',45000.00)
  4  into driver values ('d3','aman biswokarma','thapagaun','9802376544','part time',45000.00)
  5  into driver values ('d4','aschin parajuli','shantinagar','9819237689','full time',80000.00)
  6  into driver values ('d5','amrit basyal','anamnagar','9789237690','full time',80000.00)
  7  into driver values ('d6','sayush aryal','putalisadak','9866623795','full time',80000.00)
  8  into driver values ('d7','anusha pokhrel','buddhanagar','9856623789','part time',45000.00)
  9  select * from dual;

7 rows created.

SQL> select * from driver;

DRIVER_ID  DRIVER_NAME                         DRIVER_ADDRESS  DRIVER_NUM DRIVER_JOB DRIVER_EARNINGS
---------- ----------------------------------- --------------- ---------- ---------- ---------------
d1         gumaney chhetri                     balaju          9809876543 full time            80000
d2         ashish pandey                       thamel          9854176543 part time            45000
d3         aman biswokarma                     thapagaun       9802376544 part time            45000
d4         aschin parajuli                     shantinagar     9819237689 full time            80000
d5         amrit basyal                        anamnagar       9789237690 full time            80000
d6         sayush aryal                        putalisadak     9866623795 full time            80000
d7         anusha pokhrel                      buddhanagar     9856623789 part time            45000

7 rows selected.
```

Figure 14 initializing the values and showing the data of the driver table

Abhijan Basyal

6.5 Bill

Below is a creation of the bill entity table, with their properties and descriptions.

Table 8 final bill table

ATTRIBUTES	DATA TYPE	CONSTRAINTS	DESCRIPTION
bill_ID	varchar2(10)	primary key, unique	It stores the unique id of the bill.
customer_ID	varchar2(10)	not null, foreign key	It gives the data of the customer who have taken the bill.
service_ID	varchar2(10)	not null, foreign key	It gives the data of the customer who have taken the respective service.
vehicle_ID	varchar2(10)	not null, foreign key	It gives the data of the vehicle for the customer who have taken service from it.
driver_ID	varchar2(10)	not null, foreign key	It gives the data of the driver who drives the vehicle to give the services to the customer.
issued_date	date	not null	It stores the date of the bill issued.
delivery_location	varchar2(10)	not null	It stores the location of the where the delivery was taken.
total_cost	number (9,2)	not null	It stores the data of the total cost of the customer.

Abhijan Basyal

a) Relation

Here, the bill serves as the main link between the entities and contains each entity's primary key as a foreign key, providing a many-to-one relationship to each entity in turn. The bill also includes specific details about the customer using the service, as well as information about the vehicle and the driver, the customer's total cost, and the date on which the bill was issued.

b) Creating and inserting the values in the table bill

Here, the table bill is created in the SQL command line by using various command and the data is put according to the needs of the assignment.

```
SQL> create table Bill(
  2  bill_ID varchar2(10) primary key,
  3  customer_ID varchar2(10) not null,
  4  foreign key (customer_ID) references Customer(customer_ID),
  5  service_ID varchar2(10) not null,
  6  foreign key (service_ID) references Service(service_ID),
  7  vehicle_ID varchar2(10) not null,
  8  foreign key (vehicle_ID) references Vehicle(vehicle_ID),
  9  driver_ID varchar2(10) not null,
 10  foreign key (driver_ID) references Driver(driver_ID),
 11  issued_date date not null,
 12  delivery_location varchar2(10) not null,
 13  total_cost number(9,2) not null);

Table created.

SQL> describe Bill;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 BILL_ID                                   NOT NULL VARCHAR2(10)
 CUSTOMER_ID                               NOT NULL VARCHAR2(10)
 SERVICE_ID                                NOT NULL VARCHAR2(10)
 VEHICLE_ID                                NOT NULL VARCHAR2(10)
 DRIVER_ID                                 NOT NULL VARCHAR2(10)
 ISSUED_DATE                               NOT NULL DATE
 DELIVERY_LOCATION                         NOT NULL VARCHAR2(10)
 TOTAL_COST                                NOT NULL NUMBER(9,2)
```

Figure 15 creating and describing the bill table

Abhijan Basyal

```
SQL> insert into bill values(
  2  'b1','c1','s3','v6','d2',date'2022-03-05','maitidevi',80);

1 row created.
```

Figure 16 inserting the values in the bill

```
SQL> insert all
  2  into bill values ('b2','c2','s3','v5','d4',date'2022-03-20','thamel',200)
  3  into bill values ('b3','c3','s4','v7','d7',date'2022-03-27','maitidevi',200)
  4  into bill values ('b4','c4','s4','v2','d6',date'2022-04-02','ghattekulo',80)
  5  into bill values ('b5','c5','s2','v4','d1',date'2022-04-27','lugu',350)
  6  into bill values ('b6','c6','s1','v3','d3',date'2022-05-01','budantha',60)
  7  into bill values ('b7','c7','s2','v1','d5',date'2022-05-15','babarmahal',350)
  8  select * from dual;

6 rows created.

SQL> select * from bill;

BILL_ID    CUSTOMER_I SERVICE_ID VEHICLE_ID DRIVER_ID  ISSUED_DA DELIVERY_L TOTAL_COST
---------- ---------- ---------- ---------- ---------- --------- ---------- ----------
b1         c1         s3         v6         d2         05-MAR-22 maitidevi          80
b2         c2         s3         v5         d4         20-MAR-22 thamel            200
b3         c3         s4         v7         d7         27-MAR-22 maitidevi         200
b4         c4         s4         v2         d6         02-APR-22 ghattekulo         80
b5         c5         s2         v4         d1         27-APR-22 lugu              350
b6         c6         s1         v3         d3         01-MAY-22 budantha           60
b7         c7         s2         v1         d5         15-MAY-22 babarmahal        350

7 rows selected.
```

Figure 17 inserting the values and showing the data of the bill table

6.6 Displaying the total table creating in the user.

Here, it displays the total number of tables that have been produced, with a total of 5 entities and 5 tables present in the user cw_abhijan because of the process. Therefore, the figure below displays the total number of tables present in the user cw_abhijan.

```
SQL> select * from tab;

TNAME                          TABTYPE  CLUSTERID
------------------------------ -------  ----------
BILL                           TABLE
CUSTOMER                       TABLE
DRIVER                         TABLE
SERVICE                        TABLE
VEHICLE                        TABLE
```

Figure 18 list of tables in the created user

Abhijan Basyal

7. DATABASE QUERY

The practice of using a database query to extract data in accordance with the requirements of assignments is being done here. For this assignment, two database queries are needed: one for informational inquiries and the other for transaction queries that also include algebra.

7.1 Information queries

Information queries are SQL statements that are used to retrieve information from a database where according to the assignment following information are needed to retrieve:

a) List all the customer according to category

```
SQL> select * from customer order by customer_category;

CUSTOMER_I CUSTOMER_NAME                    CUSTOMER_ADDRES CUSTOMER_EMAIL                  PHONE_NUMB CUSTOMER_C REWARD_POI DISCOUNT_P
---------- -------------------------------- --------------- ------------------------------ ---------- ---------- ---------- ----------
c2         anup acharya chhetri             thamel          anupac@gmail.com               9841583459 normal     111        0
c5         tayama rai                       ghattekulo      raitayama@gmail.com            9857011004 normal     233        0
c3         sahish singh thakuri             maitidevi       sahish@gmail.com               9842234567 normal     234        0
c7         kritan kunwar chhetri            shantinagar     kritankunwar@gmail.com         9841395299 normal     455        0
c6         sikha singh sijapati             baneshwor       sikhasijapati@gmail.com        9849509433 staff      321        40
c1         abhijan basyal                   maitidevi       basyalabhijan@gmail.com        9840041397 staff      123        40
c4         renu tamang                      ghattekulo      renutamang@gmail.com           9818870509 staff      98         40

7 rows selected.
```

Figure 19 listing the customer detail by customer category

Purpose: The main purpose of this query is to display all the customer details according ordered by the customer category and from above we can see that the customer details is displayed with the customer category.

Abhijan Basyal

b) Find model and vehicle variants and sort by price in descending order.

```
SQL> select vehicle_model, vehicle_type, vehicle_price from vehicle order by vehicle_price desc;

VEHICLE_MODEL                   VEHICLE_TYPE                    VEHICLE_PRICE
------------------------------  ------------------------------  -------------
hero honda                      car                                       350
lamborgini                      car                                       350
suzuki                          car                                       350
shanks                          bike                                      150
bmw                             bike                                      150
bajaj                           bike                                      150
discover                        bike                                      150

7 rows selected.
```

Figure 20 listing the model and vehicle variants

Purpose: To display the vehicle model, name, and price, the query's primary goal is to sort the data in descending order. This provides details on the vehicle model, vehicle name, and price, with the price listed in descending order.

c) Display the number of total vehicles that use petrol.

```
SQL> select count(*) from vehicle where fuel_type='petrol';

  COUNT(*)
----------
         4
```

Figure 21 counting the total number of vehicle who use petrol

Purpose: The primary goal of this query is to present the total number of petrol-using vehicles, where it displays the total amount of car which uses petrol.

Abhijan Basyal

d) List all tickets issued from 2022/03/05 to 2022/04/05

```
SQL> select * from bill where issued_date between date'2022-03-05' and date'2022-04-05';

BILL_ID    CUSTOMER_I SERVICE_ID VEHICLE_ID DRIVER_ID  ISSUED_DA DELIVERY_L TOTAL_COST
---------- ---------- ---------- ---------- ---------- --------- ---------- ----------
b1         c1         s3         v6         d2         05-MAR-22 maitidevi          80
b2         c2         s3         v5         d4         20-MAR-22 thamel            200
b3         c3         s4         v7         d7         27-MAR-22 maitidevi         200
b4         c4         s4         v2         d6         02-APR-22 ghattekulo         80
```

Figure 22 listing the ticket issued from 2022/03/05 to 2022/04/05

Purpose: This query's primary goal is to present the bill that was issued on the relevant date, which includes the correct information present in the bill about the customer, vehicle, driver ID, and service ID that was issued on that day by the driver.

e) List the name of the driver who has the character 's' between their names.

```
SQL> select driver_name from driver where driver_name like'%s%';

DRIVER_NAME
---------------------------------
ashish pandey
aman biswokarma
aschin parajuli
amrit basyal
sayush aryal
anusha pokhrel

6 rows selected.
```

Figure 23 listing the name of the driver who have s in their middle

Purpose: The main purpose of this query is to display the name of the driver from the driver table where the name of the driver must contain s in the middle.

Abhijan Basyal

7.2 Transactional queries

Transaction queries are SQL statements that are used to manage transaction in a database where we must show the proper transaction along with relational algebra from the given queries present in the assignment such as:

a) Show the total cost and the type of service of a particular customer in a year that has used the service.


```
SQL> select customer.customer_name, service.service_name, total_cost from bill join service on bill.service_ID = service
.service_ID join customer on bill.customer_ID = customer.customer_ID where customer.customer_ID = 'c1' and extract(year
from bill.issued_date) = '2022';

CUSTOMER_NAME                    SERVICE_NAME                         TOTAL_COST
-------------------------------- ------------------------------------ ----------
abhijan basyal                   food delivery                                80
```

Figure 24 listing the total cost, service type of customer

Purpose: The main purpose of this query is to display the total cost and the type of service of a one customer that uses service in a year.

Explanation: Here in the above figure the name 'abhijan basyal' is displayed as the customer's name which take the service type or name such as 'food delivery' along with the total cost spend in the service by the customer where the total cost is '80'.

Abhijan Basyal

b) List the details of services that have been provided by a driver for the current month whose first name starts with the letter 'A'.

```
SQL> select service.service_ID, service.service_name, driver.driver_name, bill.issued_date from bill join service on bil
l.service_ID = service.service_ID join driver on bill.driver_ID = driver.driver_ID where driver.driver_name like 'a%' an
d to_char(bill.issued_date, 'MM') = 3 order by bill.bill_ID asc;

SERVICE_ID SERVICE_NAME                         DRIVER_NAME                         ISSUED_DA
---------- ------------------------------------ ----------------------------------- ---------
s3         food delivery                        ashish pandey                       05-MAR-22
s3         food delivery                        aschin parajuli                     20-MAR-22
s4         courier service                      anusha pokhrel                      27-MAR-22
```

Figure 25 listing the details of service provided by driver whose name start with a

Purpose: The main purpose of this query is to display the services provided by driver whose name starts with 'a'.

Explanation: here in above the detail information is displayed of the driver whose name start with a which gives the respective service along with the service_ID and issued_date.

Abhijan Basyal

c) List the details of customers who have used only courier service and their location of delivery.

```
SQL> select customer.customer_name, customer.customer_email, customer.phone_number, bill.delivery_location, bill.service_ID, service.service_name from bill join customer on bill.customer_ID = customer.customer_I
D join service on bill.service_ID = service.service_ID where bill.service_ID = 's4' and customer.customer_ID not in (select customer_ID from bill where service_ID != 's4');

CUSTOMER_NAME                    CUSTOMER_EMAIL                  PHONE_NUMB DELIVERY_L SERVICE_ID SERVICE_NAME
------------------------------- ------------------------------- ---------- ---------- ---------- --------------------------------
sahish singh thakuri            sahish@gmail.com                9842234567 maitidevi  s4         courier service
renu tamang                     renutamang@gmail.com            9818870509 ghattekulo s4         courier service
```

Figure 26 displaying the details of customer who used courier service along with location

Purpose: the main purpose of this query is to display the details of customer who have only use courier service along with their location.

Explanation: Here, the three table are join bill, customer, and service then the data is extracted from it where the demand of the data is to extract the details of customer who have used only courier service and their location of delivery. So, from above all the need is fulfilled and display.

37

Abhijan Basyal

d) List all the details of the top 3 highest earning drivers.

```
SQL> select driver_name,driver_earnings from driver order by driver_earnings desc;

DRIVER_NAME                      DRIVER_EARNINGS
-------------------------------- ---------------
gumaney chhetri                            80000
aschin parajuli                            80000
sayush aryal                               80000
amrit basyal                               80000
anusha pokhrel                             45000
aman biswokarma                            45000
ashish pandey                              45000

7 rows selected.
```

Figure 27 displaying the highest earning of driver.

Purpose: this query main purpose is to display the details of the top 3 highest earning drivers from the table

Explanation: Here in the above table display all the driver name with their earnings where the earning will be order by the top most highest earning of the driver from the given data in the Pirke transportation.

Abhijan Basyal

e) Display the rate of all vehicles for staff and normal customers on a particular destination.


```
SQL> select customer.customer_ID, customer.customer_name, customer.customer_category, bill.delivery_location, bill.total_cost from bill join customer on bill.customer_ID = customer.customer_ID;

CUSTOMER_I CUSTOMER_NAME                   CUSTOMER_C DELIVERY_L TOTAL_COST
---------- ------------------------------- ---------- ---------- ----------
c1         abhijan basyal                  staff      maitidevi          80
c2         anup acharya chhetri            normal     thamel            200
c3         sahish singh thakuri            normal     maitidevi         200
c4         renu tamang                     staff      ghattekulo         80
c5         tayama rai                      normal     lugu              350
c6         sikha singh sijapati            staff      budantha           60
c7         kirtan kunwar chhetri           normal     babarmahal        350

7 rows selected.
```

Figure 28 displaying the rate of vehicles

Purpose: The main purpose of this query is to display the rate of all vehicles for staff and normal customer on a particular destination where the customer category differs the price of same location.

Explanation: Here, in the above query, the customer total cost is extracted along with the customer's details, and the customer's details are displayed alongside the total cost as determined by the customer category, which varies by the discount percentage and delivery location.

Abhijan Basyal

8. File creation

Here, upon file creation, a dump file is made to store all of the user cw_abhijan's data. With the use of a dump file, we may easily recover SQL in the event of a crash or problem. Therefore, the dump file is a copy of the original Pike Transportation database system.

Figure 29 creating the dump file

Figure 30 dump file creation successfully

Abhijan Basyal

9. CRITICAL EVOLUTION

9.1 Module

This section discusses the idea of a database and contains the relational diagram, erd diagram, and normalization. Where we also learned more about the appropriate entity and the qualities that have been applied to it. This module introduces the idea of using a SQL command to design a database where we may create a system for any specific organization, store a variety of data, and use a query to extract the data as needed. This module will show us how to create an Entity Relationship Diagram (ERD) from the provided scenario, do normalization correctly, and create the final Entity Relationship Diagram (ERD) in addition to implementing the data in SQL and running various queries.

9.2 Usage

The "DATABASE" module name indicates that the major focus of this module is the database system, which has a wide range of uses such as:

- ➢ It helps us to develop the proper Entity Relational Diagram (ERD) for the organization.
- ➢ It is used to implement the proper structure for the development of database in the system.
- ➢ It teaches us how to create table and initialize the data through the SQL.
- ➢ It helps to give the proper job opportunity related to the database system.
- ➢ It helps to track the data at any time according to the need.

9.3 Relation

In the same way that the information system module "INFORMATION SYSTEM" teaches a brief introduction to the database where we implement the data on the website, the modules "DATABASE" and "INFORMATION SYSTEM" are related. The database includes the data storage in the proper format, and we can extract any data according to

Abhijan Basyal

the needs. To implement the data, a database is crucial since it must hold a lot of data. To build and initialize the data, we must use the appropriate query, just like with web designers. Therefore, there is a fundamental relationship between the database module and the information system module.

9.4 Assessment of coursework

As my transportation company's name is "Pirke," we must design a database correctly for this assignment to record all the information about the customer, driver, vehicle, service, and bill. This assignment also taught us how to design queries correctly in commands and create a database system for a transportation company that would allow us to store data conveniently and access it quickly when we need it. We learned about the Entity Relationship Diagram (ERD) and the normalization through this training as well. In essence, this course work taught us how to create the ERD, normalization, and SQL table, which is the foundation of the database system.

This coursework focuses on the appropriate operation of the database system, requiring us to create ERD, normalization, and work on SQL, which was a time-consuming task. We also had to create reports for each step of the process and present them. This course assignment was a little challenging because I struggled with the normalization, but thanks to the study and the lecturers' advice, it was resolved. The ERD diagram was challenging to create, but with the aid of the draw.io software and its instructions, it was simple. The data storage for the query, however, was challenging because I had to store the data and create a table in rough form before implementing it in SQL. Because the data and characteristics were to be maintained in SQL, the query was also challenging for me. However, by working through the rough, it was simple.

Abhijan Basyal

Bibliography

Biscobing, J., 2019. *What is Entity Relationship Diagram (ERD)? - Definition from WhatIs.com.* [Online]
Available at: https://www.techtarget.com/searchdatamanagement/definition/entity-relationship-diagram-ERD
[Accessed 3 December 2023].

IBM Corporation, 2022. *Key concepts: Entity, attribute, and entity type - IBM Documentation.* [Online]
Available at: https://www.ibm.com/docs/en/imdm/12.0?topic=concepts-key-entity-attribute-entity-type
[Accessed 3 Janauary 2023].

Rouse, M. & Vaughan, J., 2019. *What is Database Normalization?.* [Online]
Available at:
https://www.techtarget.com/searchdatamanagement/definition/normalization
[Accessed 3 January 2023].

Abhijan Basyal

APPENDIX

Figure 31 originality check

Abhijan Basyal