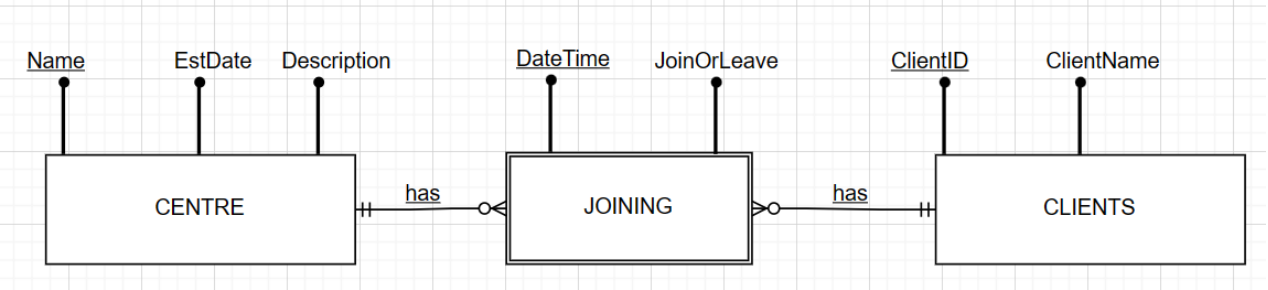
INF10025 Data Management and Analytics

Task 5 – Credit Submission

Student Number: 104082552
Student Name: Nhat Minh Tran

Credit 5a

Paste your screen capture(s) for this task here.



CENTRE (Name, EstDate, Description)

PK (Name)

CLIENTS (ClientID, ClientName)

PK (ClientID)

JOINING (DateTime, JoinOrLeave, Name, ClientID)

PK (Name, ClientID, DateTime)

FK (Name) references CENTRE

FK (ClientID) references CLIENTS

Test Data for Centre

Name	EstDate	Description
Hawthorn	20-03-1999	Hawthorn Center
FlinderStreet	19-09-1955	FlinderStreet Center
Dandenong	20-08-2006	Dandenong Center

Test Data for Clients

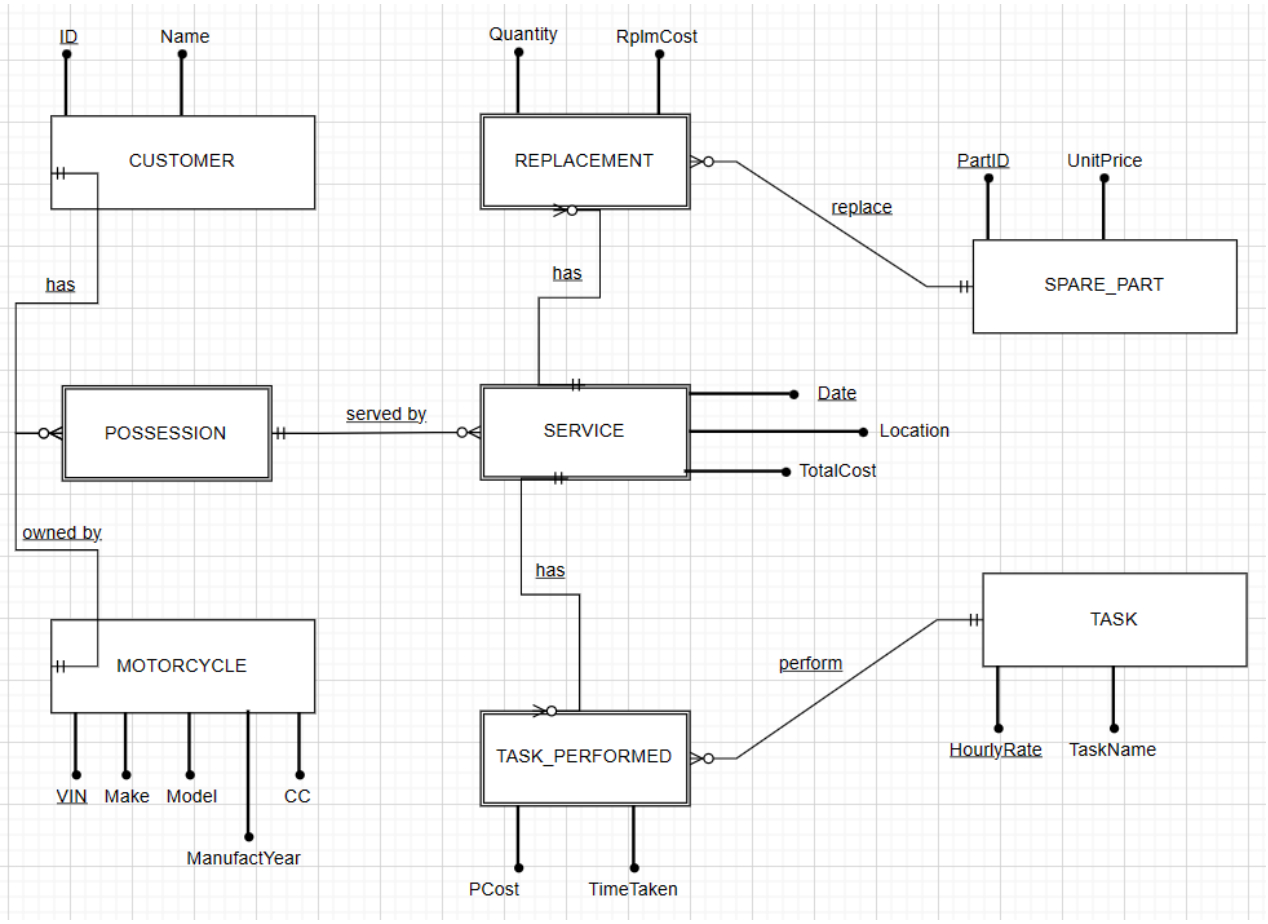
ClientID	ClientName
01	Nhat Minh
02	Anlee Nguyen
03	Danny Dor

Test Data for Joining

ClientID	Name	DateTime	JoinOrLeave
01	Hawthorn	12-04-2024	Leave
02	FlinderStreet	22-04-2024	Leave
03	Dandenong	18-03-2024	Join

Credit 5b

Paste your screen capture(s) for this task here.



Credit 5c

Paste your screen capture(s) for this task here.

Relational Schema

CUSTOMER (ID, Name)

PK (ID)

MOTORCYCLE (VIN, Make, Model, ManufactYear, CC)
PK (VIN)

POSSESSION (ID, VIN)
PK (ID, VIN)
FK (ID) references CUSTOMER
FK (VIN) references MOTORCYCLE

SERVICE (Date, Location, TotalCost, ID, VIN)
PK (Date, ID, VIN)
FK (ID) references CUSTOMER
FK (VIN) references MOTORCYCLE

SPARE_PART (PartID, UnitPrice)
PK (PartID)

REPLACEMENT (Quantity, RplmCost, PartID, ID, VIN, Date)
PK (Date, ID, VIN, PartID)
FK (ID) references CUSTOMER
FK (VIN) references MOTORCYCLE
FK (Date) references SERVICE
FK (PartID) references SPARE_PART

TASK (HourlyRate, TaskName)
PK (HourlyRate)

TASK_PERFORMED (Pcost, TimeTaken, HourlyRate, ID, VIN, Date)
PK (Date, ID, VIN, HourlyRate)
FK (ID) references CUSTOMER
FK (VIN) references MOTORCYCLE
FK (Date) references SERVICE
FK (HourlyRate) references TASK

Credit 5d

Paste your screen capture(s) for this task here.

```
CREATE TABLE CUSTOMER (
        ID INTEGER PRIMARY KEY,
        Name VARCHAR(50)
);
CREATE TABLE MOTORCYCLE (
        VIN INTEGER PRIMARY KEY,
```

```
    Make VARCHAR(50),
    Model VARCHAR(50),
    ManufactYear INTEGER,
    CC INTEGER
);
CREATE TABLE POSSESSION (
    ID INTEGER,
    VIN INTEGER,
    PRIMARY KEY (ID, VIN),
    FOREIGN KEY (ID) REFERENCES CUSTOMER(ID),
    FOREIGN KEY (VIN) REFERENCES MOTORCYCLE(VIN)
);
CREATE TABLE SERVICE (
    Date DATETIME,
    Location VARCHAR(150),
    TotalCost FLOAT,
    ID INTEGER,
    VIN INTEGER,
    PRIMARY KEY (Date, ID, VIN),
    FOREIGN KEY (ID) REFERENCES CUSTOMER(ID),
    FOREIGN KEY (VIN) REFERENCES MOTORCYCLE(VIN)
);
CREATE TABLE SPARE_PART (
    PartID INTEGER PRIMARY KEY,
    UnitPrice FLOAT
);
CREATE TABLE REPLACEMENT (
    Quantity INTEGER,
    RplmCost FLOAT,
    PartID INTEGER,
    ID INTEGER,
    VIN INTEGER,
    Date DATETIME,
```

```
        PRIMARY KEY (Date, ID, VIN, PartID),
        FOREIGN KEY (ID) REFERENCES CUSTOMER(ID),
        FOREIGN KEY (VIN) REFERENCES MOTORCYCLE(VIN),
        FOREIGN KEY (Date) REFERENCES SERVICE(Date),
        FOREIGN KEY (PartID) REFERENCES SPARE_PART(PartID)
);
CREATE TABLE TASK (
        HourlyRate FLOAT,
        TaskName VARCHAR(150),
        PRIMARY KEY (HourlyRate)
);
CREATE TABLE TASK_PERFORMED (
        Pcost FLOAT,
        TimeTaken FLOAT,
        HourlyRate FLOAT,
        ID INTEGER,
        VIN INTEGER,
        Date DATETIME,
        PRIMARY KEY (Date, ID, VIN, HourlyRate),
        FOREIGN KEY (ID) REFERENCES CUSTOMER(ID),
        FOREIGN KEY (VIN) REFERENCES MOTORCYCLE(VIN),
        FOREIGN KEY (Date) REFERENCES SERVICE(Date),
        FOREIGN KEY (HourlyRate) REFERENCES TASK(HourlyRate)
);
```

Credit 5e

Paste your screen capture(s) for this task here.

CustID	Name	Phone

125	John Coles	0401112233
278	Erin Trump	0466121455
721	Emma Knox	0423544117

CarRego	**MakeModel**
1AU8HK	Mazda 3
1KA2CA	Toyota Camry
1CZ8JK	Mazda 3
1LM3AB	Hyundai i30

CustID	**CarRego**	**StartDate**	**ReturnDate**
125	1AU8HK	31/08/2020	7/09/2020
125	1LM3AB	14/11/2020	21/11/2020
278	1AU8HK	12/09/2020	19/09/2020
278	1KA2CA	1/10/2020	8/10/2020
278	1CZ8JK	10/11/2020	12/11/2020
278	1AU8HK	26/11/2020	1/12/2020
721	1LM3AB	10/09/2020	13/09/2020

Credit 5f

Paste your screen capture(s) for this task here.

```
START TRANSACTION;

    INSERT INTO Action (ActionID, ActionDateTime, Action, ProdID, ProdQty, ProdCost)

    VALUES (1008, TO_DATE('21/01/2021', 'dd/mm/yyyy), 'Purchase', 'G43536', 2, 2100.00);

    UPDATE Product SET QtylnStock = QtylnStock - 2

   WHERE ProdID = 'G43546';

END TRANSACTION;
```

```
START TRANSACTION;

    INSERT INTO Action (ActionID, ActionDateTime, Action, ProdID, ProdQty, ProdCost)

    VALUES (1026, TO_DATE('23/01/2021', 'dd/mm/yyyy), 'Return', 'G43536', -1, 1050.00);


    UPDATE Product SET QtylnStock = QtylnStock + 1


    WHERE ProdID = 'G43546';

END TRANSACTION;
```

A sale or refund operation can go smoothly if:

- There are sufficient items in stock for the transaction.
- The quantity of brought products is always lower than the quantity of available products.
- The database system is functioning properly without any errors or interruptions.
- SQL commands are written correctly.

A sale or refund operation can go wrong if:

- There are insufficient items in stock for the transaction.
- The product ID does not exist in the database.
- The quantity of brought products is always higher than the quantity of available products.
- A refund fails if product refunds 0.
- Syntax errors in SQL commands.

Transaction committed refers to the successful transaction and the changes have been permanently saved to the database. It ensures the database remains in a consistent state.

Transaction rollback refers to the unsuccessful transaction and the changes are cancelled. The database is restored back to its previous state before the transaction began.