TO DO: CONNECT EVERYTHING, RELATIONSHIPS, REST OF TABLES

==========

```
DROP TABLE CarModel;
DROP TABLE Car;
DROP TABLE Member;
DROP TABLE Pod;
DROP TABLE Trip;
DROP TABLE MembershipPlan;
DROP TABLE Account;
DROP TABLE PrivateAccount;
DROP TABLE CompanyAccount;
DROP TABLE Review;
```

CREATE TABLE CarModel (

```
model    VARCHAR(20) NOT NULL,
make     VARCHAR(20) NOT NULL,
category VARCHAR(10),
capacity INTEGER NOT NULL,

PRIMARY KEY (make, model)

);
```

CREATE TABLE Car (

```
carName      VARCHAR (20) NOT NULL,
carYear      INTEGER NOT NULL,
transmission VARCHAR(10) NOT NULL,
regNo        INTEGER NOT NULL,
model    VARCHAR(20) NOT NULL,
make     VARCHAR(20) NOT NULL,
podid      VARCHAR(20) NOT NULL,
boardcomputerid INTEGER NOT NULL,



PRIMARY KEY(regNo)
FOREIGN KEY (make, model) REFERENCES carmodel (make, model)
FOREIGN KEY (id) REFERENCES Pod (id)
FOREIGN KEY (id) REFERENCES on board computer (id)
);
```

CREATE TABLE Member (

memberNo INTEGER NOT NULL,
passwd   VARCHAR(10) NOT NULL,
address   VARCHAR(50),
birthdate VARCHAR(20),
accountNo INTEGER NOT NULL,
podid   VARCHAR(20) NOT NULL,

PRIMARY KEY (memberNo),
FOREIGN KEY (accountno) REFERENCES Account (accountno),
FOREIGN KEY (id) REFERENCES pod (id)

);

CREATE TABLE Review (

memberNo   INTEGER NOT NULL,
regNo   INTEGER NOT NULL,
rating   NUMERIC CHECK (rating 1-5)
description VARCHAR(50),
whenDone   DATE NOT NULL,

PRIMARY KEY (memberNo, regNo)

);

CREATE TABLE Pod (
podId   VARCHAR(20) NOT NULL,
podName VARCHAR (20),
addr   VARCHAR (50),
descr   VARCHAR (20),
latitude FLOAT NOT NULL,
longitude FLOAT NOT NULL,

PRIMARY KEY (podId)
/*To-do: Position(Latitude, longitude)*/
);

CREATE TABLE Trip (

tripDate VARCHAR (20) NOT NULL,
startTime VARCHAR(20) NOT NULL,
endTime   VARCHAR(20) NOT NULL,
startOdo INTEGER NOT NULL,
distance INTEGER NOT NULL,
tripNo   INTEGER NOT NULL,
memberNo INTEGER NOT NULL,
regNo   INTEGER NOT NULL,
onboardcomputerid INTEGER NOT NULL,

PRIMARY KEY (tripNo)
PRIMARY KEY (onboardcomputerid)
FOREIGN KEY (memberno) REFERENCES member (memberno),
FOREIGN KEY (regno) REFERENCES car (regno),
FOREIGN KEY (onboardcomputerid) REFERENCES on board computer (onboardcomputerid)
ON UPDATE CASCADE
ON DELETE SET DEFAULT
);

CREATE TABLE MembershipPlan (

title   VARCHAR(10) NOT NULL,
monthly_fee   NUMERIC CHECK (monthly_fee>0),
hourly_rate   NUMERIC CHECK (hourly_rate>0),
km_rate   NUMERIC CHECK (km_rate>0),
daily_rate   NUMERIC CHECK (daily_rate>0),
daily_km_rate   NUMERIC CHECK (daily_km_rate>0),
daily_km_included NUMERIC CHECK (daily_km_included>0),
Vehicle_type_fee NUMERIC CHECk (Vehicle_type_fee>0)

PRIMARY KEY (title)

);

CREATE TABLE Account (

accountNo INTEGER NOT NULL,
accName   VARCHAR (20) NOT NULL,
since   INTEGER NOT NULL,
title   VARCHAR(10) NOT NULL,

PRIMARY KEY (accountNo),
FOREIGN KEY (title) REFERENCES membership plan (title)
);

CREATE TABLE PrivateAccount (

address   VARCHAR(50) NOT NULL,
walkscore INTEGER,
category VARCHAR(50) NOT NULL,
accountNo INTEGER NOT NULL,
PRIMARY KEY (address, accountNo)

);

CREATE TABLE CompanyAccount (

abn INTEGER UNIQUE ,
gst INTEGER NOT NULL,
accountNo INTEGER NOT NULL,

PRIMARY KEY (gst, accountNo)

);

CREATE TABLE Payment Method (

nr INTEGER NOT NULL,
type VARCHAR (20) NOT NULL ,
preferred VARCHAR (20) NOT NULL,
name VARCHAR (20),
number INTEGER,
bsb INTEGER,

```
expires INTEGER,
accountNo  INTEGER NOT NULL,

PRIMARY KEY (nr),
PRIMARY KEY (accountno),
FOREIGN KEY (accountno) REFERENCES account (accountno)
ON UPDATE CASCADE
ON DELETE SET DEFAULT



);
```

CREATE TABLE Booking (

```
id INTEGER NOT NULL,
status VARCHAR (10),
starttime TIME,
endtime TIME,
memberNo  INTEGER NOT NULL,
regNo       INTEGER NOT NULL,
whenbooked DATE,



PRIMARY KEY (id),
FOREIGN KEY (memberno) REFERENCES member (memberno),
FOREIGN KEY (regno) REFERENCES car (regno)

);
```

CREATE TABLE onBoardComputer (

```
onBoardComputerId INTEGER NOT NULL,
regNo INTEGER NOT NULL,


PRIMARY KEY (id),
FOREIGN KEY (regno) REFERENCES car (regno)
);
```

EXTENSIONS

Our group researched extensions while researching "car-sharing."

Extension 1 (done)

For this part, we had to extend our vehicle section so that it covers a wide variety of motor vehicles. This was done through ISA so we could indiciate that a vehicle could be a car, segway or a 2-wheel vehicle but not at the same time. Then we had to divide up the type of vehicles that belonged to each category. Finally, we adjusted for these changes through requirements in that are necessary in the pods as well as the "Type of Vehicle" Fee in the membership plan.

Extension 2

We felt that the system we had already covered for what extension 2 was asking for: booking car for a specific time period for a fee which was owned by an external person. This was done through "private" or "business" owned pods/vehicles. We also extended our membership plan to incorporate specialised services and pricing for governments, charity organisations and businesses. These plans can better cater to the needs of the market, an example being cheaper and more affordable deals for NGOs and mass leasing for large businesses which can be discounted in our plans due to economies of scale. We also established this extension to allow the possibility of any corporate/financial partnerships in particularly with the business segments.

EXTENSION 1

CREATE TABLE Vehicle (

vehicleName VARCHAR (20) NOT NULL,
vehicleYear INTEGER NOT NULL,
regNo INTEGER NOT NULL,
model VARCHAR(20) NOT NULL,
make VARCHAR(20) NOT NULL,
podid VARCHAR(20) NOT NULL,
boardcomputerid INTEGER NOT NULL,

PRIMARY KEY(regNo)
FOREIGN KEY (make, model) REFERENCES carmodel (make, model)

FOREIGN KEY (id) REFERENCES Pod (id)
FOREIGN KEY (id) REFERENCES on board computer (id)
);

CREATE TABLE Car(

model VARCHAR(20) NOT NULL,
make VARCHAR(20) NOT NULL,
category VARCHAR(10),
capacity INTEGER NOT NULL,
transmission VARCHAR(10) NOT NULL,

FOREIGN KEY (regNo) REFERENCES Vehicle (regNo)

);

CREATE TABLE Segway(

Adventurer VARCHAR (20),
Commuter VARCHAR (20),
Glider VARCHAR (20),

FOREIGN KEY (regNo) REFERENCES Vehicle (regNo)

);

CREATE TABLE 2Wheel(

Scooter VARCHAR (20),
Motorcyle VARCHAR (20),
Topspeed NUMERIC,

FOREIGN KEY (regNo) REFERENCES Vehicle (regNo)

);

EXTENSION 2

CREATE TABLE MembershipPlan (

```
title            VARCHAR(10) NOT NULL,
monthly_fee      NUMERIC CHECK (monthly_fee>0),
hourly_rate      NUMERIC  CHECK (hourly_rate>0),
km_rate          NUMERIC CHECK (km_rate>0),
daily_rate       NUMERIC  CHECK (daily_rate>0),
daily_km_rate    NUMERIC CHECK (daily_km_rate>0),
daily_km_included NUMERIC  CHECK (daily_km_included>0),
Vehicle_type_fee NUMERIC CHECK (Vehicle_type_fee>0),
Corporate_Partnership NUMERIC CHECK (Corporate_Partnership>0)
discounts NUMERIC CHECK (Discounts>0)

PRIMARY KEY (title)

);
```

```
CREATE TABLE CharityOrganisation (
goal VARCHAR (50),
budget NUMERIC,
discount NUMERIC,
totalcost NUMERIC,

CHECK (budget > total cost),
FOREIGN KEY (title) REFERENCES MembershipPlan (title)

);
```

```
CREATE TABLE government (
local VARCHAR (20),
state VARCHAR (20),
Federal VARCHAR (20),

FOREIGN KEY (title) REFERENCES MembershipPlan (title)

);
```

```
CREATE TABLE businesses (
small VARCHAR (20),
medium VARCHAR (20),
large VARCHAR (20),
CorporatePartnerships NUMERIC,
```

```
FOREIGN KEY (title) REFERENCES MembershipPlan (title)

);
```