FORM 1

FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY

BACS1053 DATABASE MANAGEMENT

Assignment

Semester Oct 2019

Programme	:	RMM – Bachelor of Science (Honours) in Management Mathematics With Computing
Tutorial Group	:	1
Date Submitted	:	20/Dec/2019

Team members:

No	Name (Block Letters)	Registration No.	Signature	Marks
1	CHEW HWA ERN	19WMR04184		
2	HEE SZE WEI	19WMR05920		
3	LEE SHU ERN	19WMR05933		

FORM 2

TUNKU ABDUL RAHMAN UNIVERSITY COLLEGE

FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY

Plagiarism Statement and Guideline for Late Submission of Coursework

Read, complete, and sign this statement to be submitted with the written report.

We confirm that we have read and shall comply with all the terms and conditions of TAR University College's plagiarism policy.

We declare that this assignment is free from all forms of plagiarism and for all intents and purposes is my own properly derived work.

Declaration Statement Acknowledged by

No	Name (Block Letters)	Registration No.	Signature	Date
1	CHEW HWA ERN	19WMR04184		
2	HEE SZE WEI	19WMR05920		
3	LEE SHU ERN	19WMR05933		

Table of Contents

Task 1: Business Rules

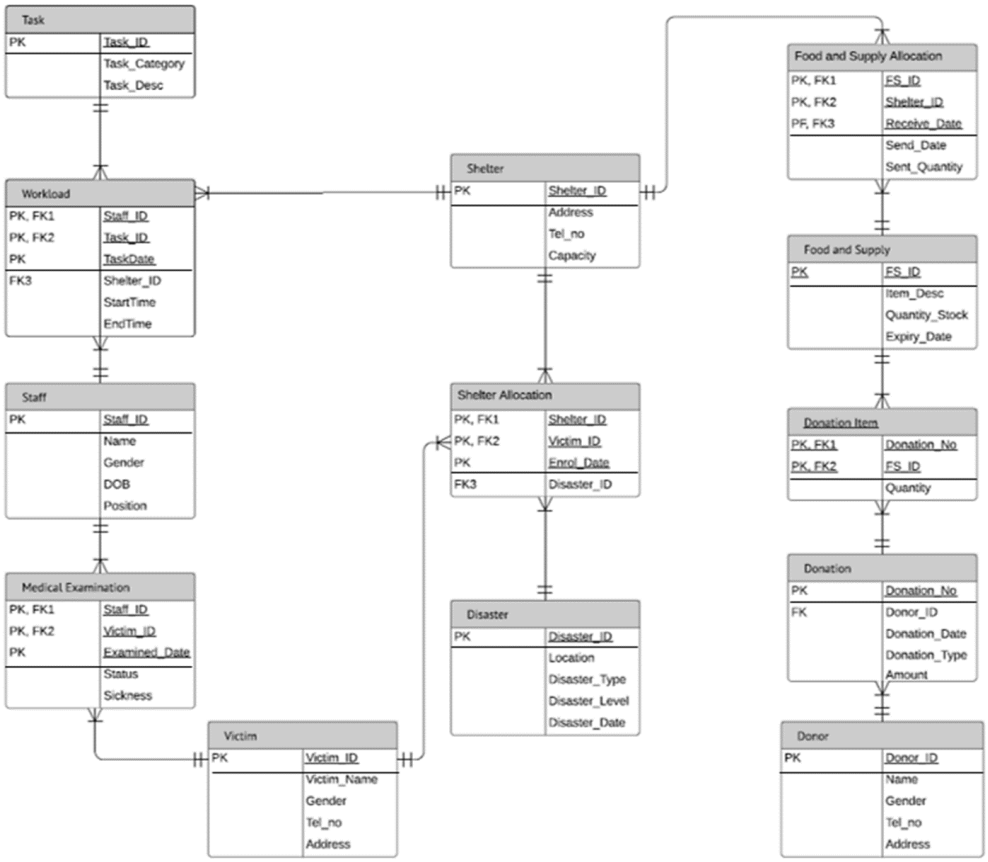
Business Rules

1. Shelter operates for 24 hours everyday.
2. Each shelter can have many victims and victims can go to different shelters. (many to many)
3. Each victim can be examined by one or many staff and each staff can examine one or many victims. (many to many)
4. Each staff can handle one or many tasks and each task can be handled by one or many staff. (many to many)
5. Each disaster enrolls many victims in the shelter, each victim is enrolled due to one or many disasters. (many to many)
6. Each donor can make one or many donations, one donation can be made by one donor only. (one to many)
7. Donor can donate either money, food and supply or both.
8. Each donation can have one or many food and supply, and each type of food and supply can exist in one or many donations. (many to many)
9. Each shelter can receive one or many food and supply, and each type of food and supply can send to one or many shelters. (many to many)

Assumption

1. All victims are alive when allocated in each shelter.
2. Victims are reallocated once maximum capacity in shelter is met.
3. Food and supply can be distributed equally to each victim.
4. All cash donations are used on purchasing food and supplies.

Task 2: Entity-Relationship Diagram (ERD)



Task 3: Normalization – 3NF

Shelter (Shelter_ID, Address, Tel_no, Capacity)

Victim (Victim_ID, Victim_Name, Gender, Tel_no, Address)

Disaster (Disaster_ID, Location, Disaster_Type, Disaster_Level, Disaster_Date)

Staff (Staff_ID, Name, Gender, DOB, Position)

Task (Task_ID, Task_Desc)

Food_and_Supply (FS_ID, Item_Desc, Quantity_Stock, Expiry_Date)

Donation (Donation_ID, Donor_ID, Donation_Type, Amount, Donation_Date)

Donor (Donor_ID, Name, Gender, Tel_no, Address)

Shelter_Allocation (Shelter_ID*, Victim_ID*, Enrol_Date, Diasater_ID*)

Medical_Examination (Staff_ID*, Victim_ID*, Examined_Date, Status, Sickness)

Workload (Staff_ID*, Task_ID*, TaskDate, StartTime, EndTime, Shelter_ID*)

Food_and_Supply_Allocation(FS_ID*, Shelter_ID*, Received_Date, Send_Date, Sent_Quantity)

Donation_Item (FS_ID*, Donation_No*, Quantity)

Task 4: Create database tables in Oracle

4.1 Shelter Table

```
create table Shelter (
      Shelter_ID VARCHAR(10) NOT NULL,
      Address VARCHAR(50) NOT NULL,
      Tel_No VARCHAR(20),
      Capacity INT NOT NULL,
PRIMARY KEY (Shelter_ID)
);
```

4.2 Disaster Table

```
create table Disaster (
      Disaster_ID VARCHAR(20) NOT NULL,
      Location VARCHAR(50) NOT NULL,
      Disaster_Type VARCHAR(30) NOT NULL,
      Disaster_Level INT NOT NULL,
      Disaster_Date DATE NOT NULL,
PRIMARY KEY(Disaster_ID)
);
```

4.3 Victim Table

```
create table Victim (
      Victim_ID VARCHAR(15) NOT NULL,
      Victim_Name VARCHAR(30) NOT NULL,
      Gender VARCHAR(15) NOT NULL,
      Tel_No VARCHAR(25),
      Address VARCHAR(50) NOT NULL,
PRIMARY KEY(Victim_ID),
```

```
CONSTRAINT chk_gender check (UPPER(gender) IN ('M', 'F'))
);
```

4.4 Staff Table

```
create table Staff (
      Staff_ID VARCHAR(15) NOT NULL,
      Name VARCHAR(30) NOT NULL,
      Gender VARCHAR(15) NOT NULL,
      DOB DATE,
      Position VARCHAR(15) NOT NULL,
PRIMARY KEY (Staff_ID),
constraint chk_staffgender check (UPPER(gender) in ('M','F'))
);
```

4.5 Task Table

```
create table Task (
      Task_ID VARCHAR(5) NOT NULL,
      Task_Category VARCHAR(15),
      Task_Desc VARCHAR(40),
PRIMARY KEY (Task_ID)
);
```

4.6 Food and Supply Table

```
create table Food_and_Supply (
      FS_ID VARCHAR(15) NOT NULL,
      Item_Desc VARCHAR(30) NOT NULL,
      Quantity_Stock INT,
      Expiry_Date DATE NULL,
PRIMARY KEY (FS_ID)
```

```
);
```

4.7 Donor Table

```
create table Donor (
      Donor_ID VARCHAR(10) NOT NULL,
      Name VARCHAR(25) NOT NULL,
      Gender VARCHAR(10),
      Tel_No VARCHAR(25),
      Address VARCHAR(25),
PRIMARY KEY (Donor_ID)
);
```

4.8 Donation Table

```
create table Donation (
      Donation_No VARCHAR(10) NOT NULL,
      Donor_ID VARCHAR(10) NOT NULL,
      Donation_Date DATE NOT NULL,
      Donation_Type VARCHAR(10) NOT NULL,
      Amount INT,
PRIMARY KEY(Donation_No),
FOREIGN KEY (Donor_ID) REFERENCES DONOR(Donor_ID),
CONSTRAINT chk_Donation_Type check (Donation_type IN ('Food', 'Supply',
'Cash'))
);
```

4.9 Shelter Allocation Table

```
create table Shelter_Allocation (
      Shelter_ID VARCHAR(10) NOT NULL,
      Victim_ID VARCHAR(15) NOT NULL,
      Enrol_Date DATE,
      Disaster_ID VARCHAR(20) NOT NULL,
PRIMARY KEY (Shelter_ID, Victim_ID, Enrol_Date),
FOREIGN KEY (Shelter_ID) REFERENCES Shelter(Shelter_ID),
FOREIGN KEY (Victim_ID) REFERENCES Victim(Victim_ID),
FOREIGN KEY (Disaster_ID) REFERENCES Disaster(Disaster_ID)
);
```

4.10 Food and Supply Allocation Table

```
create table FOOD_AND_SUPPLY_ALLOCATION (
      FS_ID VARCHAR(15) NOT NULL,
      Shelter_ID VARCHAR(10) NOT NULL,
      Receive_Date DATE,
      Send_Date DATE,
      Sent_Quantity INT,
PRIMARY KEY(FS_ID, Shelter_ID, Receive_Date),
FOREIGN KEY (FS_ID) REFERENCES Food_and_Supply(FS_ID),
FOREIGN KEY (Shelter_ID) REFERENCES Shelter(Shelter_ID)
);
```

4.11 Medical Examination Table

```
create table Medical_Examination (
      Staff_ID VARCHAR(15) NOT NULL,
      Victim_ID VARCHAR(15) NOT NULL,
      Examined_Date DATE,
      Status VARCHAR(8),
```

```
      Sickness VARCHAR(27),
PRIMARY KEY (Staff_ID, Victim_ID, Examined_Date),
FOREIGN KEY (Staff_ID) REFERENCES Staff(Staff_ID),
FOREIGN KEY (Victim_ID) REFERENCES Victim(Victim_ID)
);
```

4.12 Donation Item Table

```
create table Donation_Item (
      Donation_No VARCHAR(10) NOT NULL,
      FS_ID VARCHAR(15) NOT NULL,
      Quantity INT,
PRIMARY KEY (Donation_No, FS_ID),
FOREIGN KEY (Donation_No) REFERENCES Donation(Donation_No),
FOREIGN KEY (FS_ID) REFERENCES Food_and_Supply(FS_ID)
);
```

4.13 Workload Table

```
create table Workload (
      Staff_ID VARCHAR(15) NOT NULL,
      Task_ID VARCHAR(5) NOT NULL,
      TaskDate DATE,
      Shelter_ID VARCHAR(10) NOT NULL,
      StartTime VARCHAR(50) NOT NULL,
      EndTime VARCHAR(50),
PRIMARY KEY (Staff_ID, Task_ID, TaskDate),
FOREIGN KEY (Staff_ID) REFERENCES Staff(Staff_ID),
FOREIGN KEY (Task_ID) REFERENCES Task(Task_ID),
FOREIGN KEY (Shelter_ID) REFERENCES Shelter(Shelter_ID)
);
```

Task 5: Sample Data / Records

5.1 Shelter Table

```
insert into Shelter (Shelter_ID, Address, Tel_No, Capacity) values ('SH-01',
'35 Havey Plaza', '930-973-8396', 28);
insert into Shelter (Shelter_ID, Address, Tel_No, Capacity) values ('SH-02',
'0669 Clemons Terrace', '900-675-1517', 46);
insert into Shelter (Shelter_ID, Address, Tel_No, Capacity) values ('SH-03', '7
Porter Park', '102-808-6487', 78);
insert into Shelter (Shelter_ID, Address, Tel_No, Capacity) values ('SH-04',
'303 Truax Way', '823-955-2107', 36);
insert into Shelter (Shelter_ID, Address, Tel_No, Capacity) values ('SH-05', '3
Clove Junction', '653-228-7994', 75);
insert into Shelter (Shelter_ID, Address, Tel_No, Capacity) values ('SH-06',
'3299 Fordem Center', '859-311-5304', 29);
insert into Shelter (Shelter_ID, Address, Tel_No, Capacity) values ('SH-07',
'9598 Shasta Lane', '734-304-2672', 32);
insert into Shelter (Shelter_ID, Address, Tel_No, Capacity) values ('SH-08', '6
Northport Alley', '650-591-9205', 29);
insert into Shelter (Shelter_ID, Address, Tel_No, Capacity) values ('SH-09',
'07950 Brickson Park Plaza', '879-604-6510', 55);
insert into Shelter (Shelter_ID, Address, Tel_No, Capacity) values ('SH-10', '4
Lotheville Crossing', '750-548-0453', 26);
```

5.2 Disaster Table

```
insert into Disaster (Disaster_ID, Location, Disaster_Type, Disaster_Level,
Disaster_Date) values ('DS-01', '2 Old Gate Hill', 'Cyclonic Storms', 5,
'19-Dec-2016');
insert into Disaster (Disaster_ID, Location, Disaster_Type, Disaster_Level,
Disaster_Date) values ('DS-02', '8717 Truax Street', 'Floods', 5,
'13-Apr-2017');
insert into Disaster (Disaster_ID, Location, Disaster_Type, Disaster_Level,
Disaster_Date) values ('DS-03', '2695 Redwing Lane', 'Cyclonic Storms', 1,
'09-Jun-2017');
insert into Disaster (Disaster_ID, Location, Disaster_Type, Disaster_Level,
Disaster_Date) values ('DS-04', '5 Forest Run Point', 'Forest Fires', 3,
'15-Jul-2017');
insert into Disaster (Disaster_ID, Location, Disaster_Type, Disaster_Level,
Disaster_Date) values ('DS-05', '97 Sage Alley', 'Forest Fires', 3,
'04-Sep-2017');
insert into Disaster (Disaster_ID, Location, Disaster_Type, Disaster_Level,
Disaster_Date) values ('DS-06', '87662 Shasta Circle', 'Landslides', 4,
'05-Oct-2017');
insert into Disaster (Disaster_ID, Location, Disaster_Type, Disaster_Level,
Disaster_Date) values ('DS-07', '5462 Dottie Park', 'Forest Fires', 1,
'20-Dec-2017');
```

```
insert into Disaster (Disaster_ID, Location, Disaster_Type, Disaster_Level,
Disaster_Date) values ('DS-08', '275 Lunder Alley', 'Floods', 4,
'09-Jan-2018');
insert into Disaster (Disaster_ID, Location, Disaster_Type, Disaster_Level,
Disaster_Date) values ('DS-09', '9963 Ruskin Hill', 'Earthquake', 2,
'12-Mar-2018');
insert into Disaster (Disaster_ID, Location, Disaster_Type, Disaster_Level,
Disaster_Date) values ('DS-10', '2 Summerview Pass', 'Floods', 5,
'08-Apr-2018');
```

5.3 Victim Table

```
insert into Victim (Victim_ID, Victim_Name, Gender, Tel_No, Address) values
('V-001', 'Abie Olrenshaw', 'M', '907-418-1437', '1461 Westend Lane');
insert into Victim (Victim_ID, Victim_Name, Gender, Tel_No, Address) values
('V-002', 'Jock Daudray', 'M', '723-720-0375', '980 Elgar Lane');
insert into Victim (Victim_ID, Victim_Name, Gender, Tel_No, Address) values
('V-003', 'Terri Cockran', 'M', '579-250-6144', '1 Packers Crossing');
insert into Victim (Victim_ID, Victim_Name, Gender, Tel_No, Address) values
('V-004', 'Agosto Ughi', 'M', '704-898-0351', '23847 Rutledge Terrace');
insert into Victim (Victim_ID, Victim_Name, Gender, Tel_No, Address) values
('V-005', 'Pepito Shimon', 'M', '857-578-8011', '72293 Scofield Park');
insert into Victim (Victim_ID, Victim_Name, Gender, Tel_No, Address) values
('V-006', 'Dave Bohden', 'M', '622-277-3729', '961 Porter Alley');
insert into Victim (Victim_ID, Victim_Name, Gender, Tel_No, Address) values
('V-007', 'Julio Gildersleeve', 'M', '296-490-5434', '7 Ronald Regan Avenue');
insert into Victim (Victim_ID, Victim_Name, Gender, Tel_No, Address) values
('V-008', 'Datha Dartan', 'F', '658-780-0730', '1 Bluejay Plaza');
insert into Victim (Victim_ID, Victim_Name, Gender, Tel_No, Address) values
('V-009', 'Kearney Cameron', 'M', '436-829-0168', '9689 Waywood Pass');
insert into Victim (Victim_ID, Victim_Name, Gender, Tel_No, Address) values
('V-010', 'Leonhard Groom', 'M', '365-784-7811', '782 Little Fleur Parkway');
```

5.4 Staff Table

```
insert into Staff (Staff_ID, Name, Gender, DOB, Position) values ('S01',
'Sharlene Bunnell', 'F', '10-Sep-1991', 'Supervisor');
insert into Staff (Staff_ID, Name, Gender, DOB, Position) values ('S02',
'Elfreda Burwell', 'F', '22-Mar-1995', 'Facilitator');
insert into Staff (Staff_ID, Name, Gender, DOB, Position) values ('S03',
'Garrik Pryer', 'M', '03-Apr-1991', 'Facilitator');
insert into Staff (Staff_ID, Name, Gender, DOB, Position) values ('S04', 'Gal
Woolston', 'M', '05-May-1995', 'Paramedic');
insert into Staff (Staff_ID, Name, Gender, DOB, Position) values ('S05', 'Jeana
Drohan', 'F', '25-Oct-1998', 'Nurse');
insert into Staff (Staff_ID, Name, Gender, DOB, Position) values ('S06', 'Dot
Jacobowits', 'F', '30-Apr-2000', 'Staff Member');
insert into Staff (Staff_ID, Name, Gender, DOB, Position) values ('S07',
'Antonin Spelwood', 'M', '22-Mar-1997', Facilitator');
```

```
insert into Staff (Staff_ID, Name, Gender, DOB, Position) values ('S08',
'Beckie Roberti', 'F', '24-Feb-1990', 'Staff Member');
insert into Staff (Staff_ID, Name, Gender, DOB, Position) values ('S09', 'Linn
Flanagan', 'M', '19-Feb-1998', 'Facilitator');
insert into Staff (Staff_ID, Name, Gender, DOB, Position) values ('S10',
'Osbourne Ruzek', 'M', '25-Nov-1993', 'Nurse');
```

5.5 Task Table

```
insert into TASK (Task_ID, Task_Category, Task_Desc) values ('T01', 'CLEANING',
'CLEAN SHELTER HALL');
insert into TASK (Task_ID, Task_Category, Task_Desc) values ('T02', 'CLEANING',
'CLEAN SHELTER RESTROOMS');
insert into TASK (Task_ID, Task_Category, Task_Desc) values ('T03', 'SECURITY',
'DAY PATROL');
insert into TASK (Task_ID, Task_Category, Task_Desc) values ('T04', 'SECURITY',
'NIGHT PATROL');
insert into TASK (Task_ID, Task_Category, Task_Desc) values ('T05', 'DONATION',
'DONATION REGISTRATION');
insert into TASK (Task_ID, Task_Category, Task_Desc) values ('T06', 'DONATION',
'DONATION ALLOCATION');
insert into TASK (Task_ID, Task_Category, Task_Desc) values ('T07', 'DONATION',
'ACCOUNTS');
insert into TASK (Task_ID, Task_Category, Task_Desc) values ('T08',
'FOOD_SUPPLY', 'DISTRIBUTE FOOD');
insert into TASK (Task_ID, Task_Category, Task_Desc) values ('T09',
'FOOD_SUPPLY', 'DISTRIBUTE DAILY SUPPLIES');
insert into TASK (Task_ID, Task_Category, Task_Desc) values ('T10',
'FOOD_SUPPLY', 'DISTRIBUTE WOMENS SUPPLIES');
```

5.6 Food and Supply Table

```
insert into Food_and_Supply (FS_ID, Item_Desc, Quantity_Stock, Expiry_Date)
values ('FS001', 'Toilet papers', 200, NULL);
insert into Food_and_Supply (FS_ID, Item_Desc, Quantity_Stock, Expiry_Date)
values ('FS002', 'Clothes', 200, NULL);
insert into Food_and_Supply (FS_ID, Item_Desc, Quantity_Stock, Expiry_Date)
values ('FS003', 'Towels', 150, NULL);
insert into Food_and_Supply (FS_ID, Item_Desc, Quantity_Stock, Expiry_Date)
values ('FS004', 'Dental hygiene essentials', 160, NULL);
insert into Food_and_Supply (FS_ID, Item_Desc, Quantity_Stock, Expiry_Date)
values ('FS005', 'Bath essentials', 150, NULL);
insert into Food_and_Supply (FS_ID, Item_Desc, Quantity_Stock, Expiry_Date)
values ('FS006', 'First-aid items', 150, NULL);
```

```
insert into Food_and_Supply (FS_ID, Item_Desc, Quantity_Stock, Expiry_Date)
values ('FS007', 'Pads and tampons', 200, NULL);
insert into Food_and_Supply (FS_ID, Item_Desc, Quantity_Stock, Expiry_Date)
values ('FS008', 'Blankets', 170, NULL);
insert into Food_and_Supply (FS_ID, Item_Desc, Quantity_Stock, Expiry_Date)
values ('FS009', 'Pillows', 170, NULL);
insert into Food_and_Supply (FS_ID, Item_Desc, Quantity_Stock, Expiry_Date)
values ('FS010', 'Undergarments', 200, NULL);
```

5.7 Donor Table

```
insert into DONOR (Donor_ID, Name, Gender, Tel_No, Address) values ('D01',
'Gael Nevet', 'M', '901-505-6934', '5228 Manley Lane');
insert into DONOR (Donor_ID, Name, Gender, Tel_No, Address) values ('D02',
'Nevins Cramer', 'M', '102-485-0381', '4 Sullivan Hill');
insert into DONOR (Donor_ID, Name, Gender, Tel_No, Address) values ('D03',
'Paola Langstrath', 'F', '636-450-9825', '799 Trailsway Pass');
insert into DONOR (Donor_ID, Name, Gender, Tel_No, Address) values ('D04',
'Irma Wantling', 'F', '266-292-1656', '5097 Mallory Road');
insert into DONOR (Donor_ID, Name, Gender, Tel_No, Address) values ('D05',
'Alina Carreck', 'F', '895-710-9381', '645 Birchwood Court');
insert into DONOR (Donor_ID, Name, Gender, Tel_No, Address) values ('D06',
'Arri Goodswen', 'M', '657-509-3965', '96578 David Terrace');
insert into DONOR (Donor_ID, Name, Gender, Tel_No, Address) values ('D07',
'Nerta Fadian', 'F', '673-514-3348', '537 Fulton Way');
insert into DONOR (Donor_ID, Name, Gender, Tel_No, Address) values ('D08',
'Darill Philippault', 'M', '336-981-5280', '1547 Fulton Court');
insert into DONOR (Donor_ID, Name, Gender, Tel_No, Address) values ('D09',
'Maryrose Swannell', 'F', '322-814-8457', '6 Tony Road');
insert into DONOR (Donor_ID, Name, Gender, Tel_No, Address) values ('D10',
'Lanni Whines', 'F', '993-974-8721', '72 Union Terrace');
```

5.8 Donation Table

```
insert into Donation (Donation_No, Donor_ID, Donation_Date, Donation_Type,
Amount) values ('DN-001', 'D08', '20-Dec-2016', 'Cash', 232);
insert into Donation (Donation_No, Donor_ID, Donation_Date, Donation_Type,
Amount) values ('DN-002', 'D26', '31-Dec-2016', 'Supply', NULL);
insert into Donation (Donation_No, Donor_ID, Donation_Date, Donation_Type,
Amount) values ('DN-003', 'D01', '16-Jan-2017', 'Cash', 172);
insert into Donation (Donation_No, Donor_ID, Donation_Date, Donation_Type,
Amount) values ('DN-004', 'D09', '26-Jan-2017', 'Supply', NULL);
insert into Donation (Donation_No, Donor_ID, Donation_Date, Donation_Type,
Amount) values ('DN-005', 'D24', '06-Mar-2017', 'Supply', NULL);
insert into Donation (Donation_No, Donor_ID, Donation_Date, Donation_Type,
Amount) values ('DN-006', 'D16', '03-May-2017', 'Supply', NULL);
insert into Donation (Donation_No, Donor_ID, Donation_Date, Donation_Type,
Amount) values ('DN-007', 'D29', '08-Jul-2017', 'Food', NULL);
insert into Donation (Donation_No, Donor_ID, Donation_Date, Donation_Type,
Amount) values ('DN-008', 'D02', '24-Sep-2017', 'Food', NULL);
```

```
insert into Donation (Donation_No, Donor_ID, Donation_Date, Donation_Type,
Amount) values ('DN-009', 'D11', '22-Oct-2017', 'Cash', 26);
insert into Donation (Donation_No, Donor_ID, Donation_Date, Donation_Type,
Amount) values ('DN-010', 'D21', '12-Oct-2017', 'Food', NULL);
```

5.9 Shelter Allocation Table

```
insert into Shelter_Allocation (Shelter_ID, Victim_ID, Enrol_Date, Disaster_ID)
values ('SH-06', 'V-010', '19-Dec-2016', 'DS-01');
insert into Shelter_Allocation (Shelter_ID, Victim_ID, Enrol_Date, Disaster_ID)
values ('SH-07', 'V-039', '19-Dec-2016', 'DS-01');
insert into Shelter_Allocation (Shelter_ID, Victim_ID, Enrol_Date, Disaster_ID)
values ('SH-03', 'V-011', '19-Dec-2016', 'DS-01');
insert into Shelter_Allocation (Shelter_ID, Victim_ID, Enrol_Date, Disaster_ID)
values ('SH-07', 'V-048', '22-Dec-2016', 'DS-01');
insert into Shelter_Allocation (Shelter_ID, Victim_ID, Enrol_Date, Disaster_ID)
values ('SH-08', 'V-017', '24-Dec-2016','DS-01');
insert into Shelter_Allocation (Shelter_ID, Victim_ID, Enrol_Date, Disaster_ID)
values ('SH-03', 'V-040', '27-Jan-2017', 'DS-01');
insert into Shelter_Allocation (Shelter_ID, Victim_ID, Enrol_Date, Disaster_ID)
values ('SH-09', 'V-003', '28-Jan-2017', 'DS-01');
insert into Shelter_Allocation (Shelter_ID, Victim_ID, Enrol_Date, Disaster_ID)
values ('SH-04', 'V-020', '17-Feb-2017', 'DS-01');
insert into Shelter_Allocation (Shelter_ID, Victim_ID, Enrol_Date, Disaster_ID)
values ('SH-04', 'V-022', '17-Feb-2017', 'DS-01');
insert into Shelter_Allocation (Shelter_ID, Victim_ID, Enrol_Date, Disaster_ID)
values ('SH-05', 'V-036', '20-Mar-2017', 'DS-01');
insert into Shelter_Allocation (Shelter_ID, Victim_ID, Enrol_Date, Disaster_ID)
values ('SH-06', 'V-029', '13-Apr-2017', 'DS-02');
```

5.10 Food and Supply Allocation Table

```
insert into Food_and_Supply_Allocation (FS_ID, Shelter_ID, Send_Date,
Receive_Date, Sent_Quantity) values ('FS001', 'SH-03', '31-Dec-2016',
'04-Jan-2017', 26);
insert into Food_and_Supply_Allocation (FS_ID, Shelter_ID, Send_Date,
Receive_Date, Sent_Quantity) values ('FS003', 'SH-08', '31-Dec-2016',
'05-Jan-2017', 30);
insert into Food_and_Supply_Allocation (FS_ID, Shelter_ID, Send_Date,
Receive_Date, Sent_Quantity) values ('FS005', 'SH-01', '31-Dec-2016',
'07-Jan-2017', 27);
insert into Food_and_Supply_Allocation (FS_ID, Shelter_ID, Send_Date,
Receive_Date, Sent_Quantity) values ('FS007', 'SH-10', '31-Dec-2016',
'02-Jan-2017', 20);
```

```
insert into Food_and_Supply_Allocation (FS_ID, Shelter_ID, Send_Date,
Receive_Date, Sent_Quantity) values ('FS009', 'SH-07', '31-Dec-2016',
'03-Jan-2017', 25);
insert into Food_and_Supply_Allocation (FS_ID, Shelter_ID, Send_Date,
Receive_Date, Sent_Quantity) values ('FS002', 'SH-02', '31-Dec-2016',
'03-Jan-2017', 15);
insert into Food_and_Supply_Allocation (FS_ID, Shelter_ID, Send_Date,
Receive_Date, Sent_Quantity) values ('FS004', 'SH-04', '31-Dec-2016',
'03-Jan-2017', 20);
insert into Food_and_Supply_Allocation (FS_ID, Shelter_ID, Send_Date,
Receive_Date, Sent_Quantity) values ('FS006', 'SH-06', '31-Dec-2016',
'03-Jan-2017', 25);
insert into Food_and_Supply_Allocation (FS_ID, Shelter_ID, Send_Date,
Receive_Date, Sent_Quantity) values ('FS008', 'SH-09', '31-Dec-2016',
'03-Jan-2017', 20);
insert into Food_and_Supply_Allocation (FS_ID, Shelter_ID, Send_Date,
Receive_Date, Sent_Quantity) values ('FS010', 'SH-05', '31-Dec-2016',
'03-Jan-2017', 30);
```

5.11 Medical Examination Table

```
insert into Medical_Examination (Staff_ID, Victim_ID, Examined_Date, Status,
Sickness) values ('S04', 'V-010', '21-Dec-2016', 'Moderate', 'Dengue fever');
insert into Medical_Examination (Staff_ID, Victim_ID, Examined_Date, Status,
Sickness) values ('S04', 'V-039', '24-Dec-2016', 'Severe', 'Typhoid Fever');
insert into Medical_Examination (Staff_ID, Victim_ID, Examined_Date, Status,
Sickness) values ('S05', 'V-011', '25-Dec-2016', 'Good', NULL);
insert into Medical_Examination (Staff_ID, Victim_ID, Examined_Date, Status,
Sickness) values ('S05', 'V-048', '27-Dec-2016', 'Severe', 'Dengue fever');
insert into Medical_Examination (Staff_ID, Victim_ID, Examined_Date, Status,
Sickness) values ('S05', 'V-017', '29-Dec-2016', 'Moderate', 'Measles');
insert into Medical_Examination (Staff_ID, Victim_ID, Examined_Date, Status,
Sickness) values ('S10', 'V-040', '28-Jan-2017', 'Moderate', 'Asthma');
insert into Medical_Examination (Staff_ID, Victim_ID, Examined_Date, Status,
Sickness) values ('S10', 'V-003', '29-Jan-2017', 'Moderate', 'Malaria');
insert into Medical_Examination (Staff_ID, Victim_ID, Examined_Date, Status,
Sickness) values ('S11', 'V-020', '23-Feb-2017', 'Good', NULL);
insert into Medical_Examination (Staff_ID, Victim_ID, Examined_Date, Status,
Sickness) values ('S11', 'V-022', '27-Feb-2017', 'Good', NULL);
insert into Medical_Examination (Staff_ID, Victim_ID, Examined_Date, Status,
Sickness) values ('S12', 'V-036', '23-Mar-2017', 'Severe', 'Measles');
```

5.12 Donation Item Table

```
insert into Donation_Item (Donation_No, FS_ID, Quantity) values ('DN-007',
'FS014', 100);
insert into Donation_Item (Donation_No, FS_ID, Quantity) values ('DN-008',
'FS015', 100);
insert into Donation_Item (Donation_No, FS_ID, Quantity) values ('DN-010',
'FS016', 100);
insert into Donation_Item (Donation_No, FS_ID, Quantity) values ('DN-011',
'FS017', 100);
insert into Donation_Item (Donation_No, FS_ID, Quantity) values ('DN-012',
'FS018', 100);
insert into Donation_Item (Donation_No, FS_ID, Quantity) values ('DN-013',
'FS019', 100);
insert into Donation_Item (Donation_No, FS_ID, Quantity) values ('DN-014',
'FS020', 100);
insert into Donation_Item (Donation_No, FS_ID, Quantity) values ('DN-015',
'FS021', 100);
insert into Donation_Item (Donation_No, FS_ID, Quantity) values ('DN-017',
'FS022', 100);
insert into Donation_Item (Donation_No, FS_ID, Quantity) values ('DN-019',
'FS023', 150);
```

5.13 Workload Table

```
insert into Workload (Staff_ID, Task_ID, TaskDate, Shelter_ID, StartTime,
EndTime) values ('S36', 'T16', '02-May-2019', 'SH-05', '12:00 AM', '11:59 PM');
insert into Workload (Staff_ID, Task_ID, TaskDate, Shelter_ID, StartTime,
EndTime) values ('S36', 'T07', '02-Jan-2017', 'SH-03', '9:00 AM', '5:00 PM');
insert into Workload (Staff_ID, Task_ID, TaskDate, Shelter_ID, StartTime,
EndTime) values ('S40', 'T18', '24-Jun-2018', 'SH-08', '12:00 AM', '11:59 PM');
insert into Workload (Staff_ID, Task_ID, TaskDate, Shelter_ID, StartTime,
EndTime) values ('S08', 'T05', '01-Jul-2019', 'SH-02', '9:00 AM', '5:00 PM');
insert into Workload (Staff_ID, Task_ID, TaskDate, Shelter_ID, StartTime,
EndTime) values ('S14', 'T16', '25-Mar-2018', 'SH-03', '12:00 AM', '11:59 PM');
insert into Workload (Staff_ID, Task_ID, TaskDate, Shelter_ID, StartTime,
EndTime) values ('S31', 'T06', '01-May-2017', 'SH-02', '9:00 AM', '5:00 PM');
insert into Workload (Staff_ID, Task_ID, TaskDate, Shelter_ID, StartTime,
EndTime) values ('S35', 'T17', '11-Sep-2018', 'SH-03', '9:00 AM', '5:00 PM');
insert into Workload (Staff_ID, Task_ID, TaskDate, Shelter_ID, StartTime,
EndTime) values ('S31', 'T06', '13-Mar-2017', 'SH-01', '10:00 PM', '5:00 AM');
insert into Workload (Staff_ID, Task_ID, TaskDate, Shelter_ID, StartTime,
EndTime) values ('S09', 'T14', '01-Jun-2019', 'SH-06', '9:00 AM', '5:00 PM');
insert into Workload (Staff_ID, Task_ID, TaskDate, Shelter_ID, StartTime,
EndTime) values ('S38', 'T10', '13-Apr-2017', 'SH-01', '9:00 AM', '5:00
PM');FOREIGN KEY (Task_ID) REFERENCES Task(Task_ID),
```

Task 6: Create Queries

6.1 Chew Hwa Ern

6.1.1 Query 1 Victims' Sickness Details

Purpose:
The purpose of this query is to display Victim's health details arranged by their respective sicknesses based on user's input.

SQL Statement:

```
set pagesize 300
set linesize 300

ALTER SESSION SET NLS_DATE_FORMAT = 'DD/MON/YYYY';

ACCEPT Sickness PROMPT 'Enter sickness:';

COLUMN Sickness FORMAT A30 HEADING 'Sickness' ;
COLUMN Victim_ID FORMAT A12 HEADING 'Victim ID' ;
COLUMN Examined_Date FORMAT A14 HEADING 'Examined Date';
COLUMN Status FORMAT A12 HEADING 'Status';

BREAK ON Sickness SKIP 1 ON Victim_ID SKIP 1;

SELECT  Sickness, V.Victim_ID, Status, Examined_Date
FROM Victim V, Medical_Examination ME
WHERE V.Victim_ID = ME.Victim_ID AND Sickness IS NOT NULL AND
Sickness = '&Sickness'
ORDER BY Sickness, V.Victim_ID;

CLEAR COLUMNS;
CLEAR BREAKS;
```

Sample Output:

```
SQL> @"C:\Users\hwaer\Desktop\Query 1.txt";

Session altered.

Enter sickness:Dengue fever
old   3: WHERE V.Victim_ID = ME.Victim_ID AND Sickness IS NOT NULL AND Sickness = '&Sickness'
new   3: WHERE V.Victim_ID = ME.Victim_ID AND Sickness IS NOT NULL AND Sickness = 'Dengue fever'

Sickness                       Victim ID    Status       Examined Date
------------------------------ ------------ ------------ --------------
Dengue fever                   V-001        Moderate     16/NOV/2018

                               V-010        Moderate     21/DEC/2016

                               V-014        Severe       28/FEB/2019

                               V-015        Moderate     05/SEP/2017
                                            Severe       09/MAY/2018

                               V-020        Severe       11/JAN/2018

                               V-022        Severe       17/AUG/2018

                               V-023        Severe       28/FEB/2019

                               V-025        Moderate     08/SEP/2019

                               V-044        Moderate     01/MAY/2018
```

6.1.2 Query 2: Expired Food

Purpose:
The purpose of this query is to display Expired Food Details that have been sent out to different Shelters. This is to track the expired food to make re-collection easier.

SQL Statement:

```
set pagesize 300
set linesize 300

ALTER SESSION SET NLS_DATE_FORMAT = 'DD/MON/YYYY';

COLUMN Send_Date HEADING 'Send Date';
COLUMN Item_Desc FORMAT A20 HEADING 'Item Desc' ;
COLUMN Expiry_Date FORMAT A12 HEADING 'Expiry Date';
COLUMN FS_ID FORMAT A18 HEADING 'Food and Supply ID' ;
COLUMN Shelter_ID FORMAT A10 HEADING 'Shelter ID' ;

BREAK ON FS_ID SKIP 1 ON Item_Desc SKIP 1;
SELECT FS.FS_ID, Item_Desc, Shelter_ID, Send_Date, Expiry_Date
FROM Food_And_Supply FS, Food_And_Supply_Allocation FSA
WHERE FS.FS_ID= FSA.FS_ID AND Expiry_Date <= Send_Date
ORDER BY FS_ID, Shelter_ID;

CLEAR BREAKS;
CLEAR COLUMNS;
```

Sample Output:

```
SQL> BREAK ON FS_ID SKIP 1 ON Item_Desc SKIP 1;
SQL> SELECT FS.FS_ID, Item_Desc, Shelter_ID, Send_Date, Expiry_Date
  2  FROM Food_And_Supply FS, Food_And_Supply_Allocation FSA
  3  WHERE FS.FS_ID= FSA.FS_ID AND Expiry_Date <= Send_Date
  4  ORDER BY FS_ID, Shelter_ID;

Food and Supply ID Item Desc            Shelter ID Send Date   Expiry Date
------------------ -------------------- ---------- ----------- ------------
FS033              Eggs                 SH-09      24/SEP/2017 08/AUG/2017


FS035              Eggs                 SH-01      15/NOV/2017 12/NOV/2017
                                        SH-02      15/NOV/2017 12/NOV/2017
                                        SH-04      15/NOV/2017 12/NOV/2017
                                        SH-09      15/NOV/2017 12/NOV/2017
                                        SH-10      15/NOV/2017 12/NOV/2017


FS039              Eggs                 SH-08      29/JUL/2018 21/JUL/2018


7 rows selected.

SQL>
SQL> CLEAR BREAKS;
breaks cleared
SQL> CLEAR COLUMNS;
columns cleared
SQL>
```

6.1.3 Query 3: Shelter Availability

Purpose:
The purpose of this query is to display Shelter's availability and the number of victims occupying each shelter. This helps to keep track of the number of victims in each shelter to ease food and supply distributions or shelter allocations processes.

SQL Statement:

```
set pagesize 300
set linesize 300

COLUMN Shelter_ID FORMAT A10 HEADING 'Shelter ID' ;
COLUMN Address FORMAT A30 HEADING 'Shelter Address' ;
COLUMN No_Of_Victims FORMAT 999 HEADING 'No_Of_Victims';
COLUMN Capacity FORMAT 999 HEADING 'Capacity';

SELECT S.Shelter_ID, S.Address, Capacity, count(V.Victim_ID) AS
No_Of_Victims, Capacity-(COUNT(V.Victim_ID)) AS Availability
FROM Shelter S, Shelter_Allocation SA, Victim V
WHERE SA.Shelter_ID = S.Shelter_ID AND SA.Victim_ID = V.Victim_ID
GROUP BY S.Shelter_ID, S.Address, Capacity
ORDER BY S.Shelter_ID;

CLEAR COLUMNS;
```

Sample Output:

```
SQL> COLUMN Capacity FORMAT 999 HEADING 'Capacity';
SQL>
SQL> SELECT S.Shelter_ID, S.Address, Capacity, count(V.Victim_ID) AS No_Of_Victims,
Capacity-(COUNT(V.Victim_ID)) AS Availability
  2  FROM Shelter S, Shelter_Allocation SA, Victim V
  3  WHERE SA.Shelter_ID = S.Shelter_ID AND SA.Victim_ID = V.Victim_ID
  4  GROUP BY S.Shelter_ID, S.Address, Capacity
  5  ORDER BY S.Shelter_ID;

Shelter ID Shelter Address                Capacity No_Of_Victims AVAILABILITY
---------- ------------------------------ -------- ------------ ------------
SH-01      35 Havey Plaza                       28           18           10
SH-02      0669 Clemons Terrace                 46           14           32
SH-03      7 Porter Park                        78           23           55
SH-04      303 Truax Way                        36           27            9
SH-05      3 Clove Junction                     75           16           59
SH-06      3299 Fordem Center                   29           13           16
SH-07      9598 Shasta Lane                     32           15           17
SH-08      6 Northport Alley                    29           18           11
SH-09      07950 Brickson Park Plaza            55           17           38
SH-10      4 Lotheville Crossing                26           14           12

10 rows selected.

SQL>
SQL> CLEAR COLUMNS;
columns cleared
SQL> CLEAR BREAKS;
breaks cleared
SQL>
```

6.1.4 Report 1: Food and Supply Report

Purpose:
The purpose of this report is to show the details of Food and supply sent to each shelter. This is shown based on user input on Shelter ID. This will ensure that there is sufficient food and supply for the victims in each shelters.

SQL Statement:

```
set linesize 300
set pagesize 300

COLUMN Shelter_ID FORMAT A10 HEADING 'Shelter ID' ;
COLUMN No_Of_Victims FORMAT 999 HEADING 'No Of Victims';
COLUMN FS_ID FORMAT A5 HEADING 'FS ID' ;
COLUMN Item_Desc FORMAT A25 HEADING 'Item Desc' ;
COLUMN Sent_Quantity FORMAT 999 HEADING 'QTY';
COLUMN Receive_Date FORMAT A14 HEADING 'Receive_Date';

ACCEPT Shelter_ID PROMPT 'Please Enter Shelter ID (SH-01 to SH-10):
';

TTITLE LEFT ================================= SKIP 1 -
LEFT 'Food and Supply Report' SKIP 1-
LEFT ================================= SKIP 1-
LEFT 'Food and Supply Distribution at ' '&shelter_id' SKIP 1 -
LEFT_DATE SKIP 1-
LEFT 'Page: 'FORMAT 999 SQL.PNO SKIP 2


BREAK ON Shelter_ID SKIP 1 ON NO_OF_VICTIMS ON FS_ID SKIP 1;
COMPUTE SUM LABEL ' ' OF Sent_Quantity ON FS_ID SKIP 1;

SELECT COUNT(SA.Victim_ID) AS NO_OF_VICTIMS, SA.Shelter_ID, FS.FS_ID,
Item_Desc, Sent_Quantity, Receive_Date
FROM Food_And_Supply_Allocation FSA, Shelter_Allocation SA,
Food_And_Supply FS
WHERE SA.Shelter_ID = FSA.Shelter_ID AND FS.FS_ID = FSA.FS_ID AND
SA.Shelter_ID = '&shelter_id'
```

```
GROUP BY SA.Shelter_ID, Item_Desc, FS.FS_ID, Sent_Quantity,
Receive_Date
ORDER BY SA.Shelter_ID, FS.FS_ID, Receive_Date;

CLEAR BREAKS;
CLEAR COLUMN;
CLEAR COMPUTE;
TTITLE OFF;
```

Sample Output:

```
==================================
Food and Supply Report
==================================
Food and Supply Distribution at SH-01
20/DEC/2019
Page:    1

No Of Victims Shelter ID FS ID Item Desc                 QTY Receive_Date
------------- ---------- ----- ------------------------ ---- --------------
           18 SH-01      FS001 Toilet papers              30 28/JAN/2017
                               Toilet papers              12 24/NOV/2019
                         *****                           ----
                                                          42

                         FS003 Towels                     10 27/DEC/2018
                         *****                           ----
                                                          10

                         FS005 Bath essentials            27 07/JAN/2017
                               Bath essentials            12 20/OCT/2019
                         *****                           ----
                                                          39

                         FS007 Pads and tampons           30 27/JAN/2019
                         *****                           ----
                                                          30

                         FS008 Blankets                   25 16/JUN/2017
                               Blankets                   25 16/JUN/2018
                               Blankets                   10 16/JUN/2019
```

6.1.5 Report 2: Disaster Victim's Report

Purpose:
The purpose of this report is to show the total number of victims and victim details in each shelter based on different disaster types. This report can be used to identify which disaster brings in most victims in order for shelter to be prepared in the future.

SQL Statement:

```
set linesize 300
set pagesize 300

ALTER SESSION SET NLS_DATE_FORMAT = 'DD/MON/YYYY';

COLUMN Disaster_Type FORMAT A16 HEADING 'Disaster Type';
COLUMN Shelter_ID FORMAT A10 HEADING 'Shelter ID' ;
COLUMN Victim_ID FORMAT A12 HEADING 'Victim ID' ;
COLUMN Victim_Name HEADING 'Victim Name';
COLUMN Disaster_Date FORMAT A14 HEADING 'Disaster Date';

ACCEPT Disaster_Type PROMPT 'Enter Disaster Type (Floods, Haze,
Landslides, Forest Fires, Earthquake, Cyclonic Storms): ';

TTITLE LEFT ================================= SKIP 1 -
LEFT 'Disaster Victims Report' SKIP 1-
LEFT ================================= SKIP 1-
LEFT 'Victims of''&Disaster_Type''allocations' SKIP 1 -
LEFT_DATE SKIP 1-
LEFT 'Page: 'FORMAT 999 SQL.PNO SKIP 2

BREAK ON Disaster_Date SKIP 1 ON Disaster_Type ON Shelter_ID SKIP 1;
COMPUTE COUNT LABEL 'Total Victims' OF Victim_Name ON Disaster_Date;

SELECT Disaster_Date, Disaster_Type, SA.Shelter_ID, V.Victim_ID,
Victim_Name
FROM Disaster D, Shelter_Allocation SA, Victim V
WHERE D.Disaster_ID = SA.Disaster_ID AND SA.Victim_ID = V.Victim_ID
AND Disaster_Type='&Disaster_Type'
ORDER BY Disaster_Date, Disaster_Type, SA.Shelter_ID, V.Victim_ID;


CLEAR BREAKS;
CLEAR COLUMN;
CLEAR COMPUTE;
TTITLE OFF;
```

Sample Output:

```
================================
Disaster Victims Report
================================
Victims of'Floods'allocations
20/DEC/2019
Page:    1

Disaster Date  Disaster Type    Shelter ID Victim ID    Victim Name
-------------- ---------------- ---------- ------------ ------------------------------
13/APR/2017    Floods           SH-02      V-018        Sallee Colnett

                                SH-03      V-007        Julio Gildersleeve

                                SH-04      V-007        Julio Gildersleeve
                                           V-041        Kathi Dunnan

                                SH-05      V-015        Gawen Aylwin

                                SH-06      V-029        Jamal Croxton

                                SH-08      V-027        Eveline Bruneton
                                           V-038        Erda Branston

************** ***************** **********             ------------------------------
Total Victims                                                                      8

09/JAN/2018    Floods           SH-01      V-017        Jonah Bartolomeu
```

6.2 Hee Sze Wei

6.2.1 Query 1: Victim Details based on Disaster_Type

Purpose: The purpose of this query is to let user to input a disaster type and find the details of the victims based on the disaster.

SQL Statement:

```
COLUMN Disaster_ID FORMAT A12 HEADING 'Disaster ID' ;
COLUMN Disaster_Type FORMAT A16 HEADING 'Disaster Type';
COLUMN Victim_ID FORMAT A12 HEADING 'Victim ID' ;
COLUMN Victim_Name HEADING 'Victim Name';
COLUMN Enrol_Date FORMAT A14 HEADING 'Enrol Date';

ACCEPT D_Type VARCHAR
PROMPT 'Please enter a disaster type ('Floods', 'Haze', 'Landslides',
'Forest Fires', 'Earthquake', 'Cyclonic Stroms')

BREAK ON Disaster_Type SKIP PAGE ON Disaster_ID ON Victim_ID ON
Victim_Name SKIP 1;

SELECT D.Disaster_Type, D.Disaster_ID, V.Victim_ID, V.Victim_Name,
Enrol_Date
FROM Victim V, Shelter_Allocation SA, Disaster D
WHERE V.victim_ID = SA.victim_ID AND D.disaster_ID = SA.disaster_ID
AND disaster_type = &D_Type
ORDER BY D.Disaster_ID, V.Victim_ID, Enrol_Date;

CLEAR COLUMNS
CLEAR BREAKS
```

Sample Output:

```
SQL> @"C:\Users\Sereneee\Desktop\QUERY 1.txt";
SP2-0003: Ill-formed ACCEPT command starting as VARCHAR
'Please enter a disaster type ('Floods', 'Haze', 'Landslides', 'Forest Fires', 'Earthquake', 'Cyclonic Stroms')
Enter value for d_type: Haze
old   3: WHERE V.victim_ID = SA.victim_ID AND D.disaster_ID = SA.disaster_ID AND disaster_type = '&D_Type'
new   3: WHERE V.victim_ID = SA.victim_ID AND D.disaster_ID = SA.disaster_ID AND disaster_type = 'Haze'

Disaster Type    Disaster ID  Victim ID    Victim Name                    Enrol Date
---------------- ------------ ------------ ------------------------------ -------------
Haze             DS-14        V-006        Dave Bohden                    06/FEB/2019

                              V-022        Darrick Moubray                11/FEB/2019

                              V-023        Tabby Bartolacci               27/FEB/2019

                              V-025        Ganny Kemetz                   18/FEB/2019
                                                                          22/FEB/2019

                              V-026        Rycca Corsar                   27/FEB/2019

                              V-031        Miran Bachshell                19/FEB/2019

                              V-032        Margarete Turford              06/FEB/2019

                 DS-18        V-034        Mikel Padefield                03/MAY/2019

                              V-039        Pippo Putland                  15/MAY/2019

                              V-046        Teddi Ethelstone               03/MAY/2019


11 rows selected.
```

6.2.2 Query 2: Task details and staffs who handle the task

Purpose: The purpose of this query is to let user to input a task categoty and find what are the task description, who are the staffs that handle the task on which shelter and the task date based on the task category.

SQL Statement:

```
COLUMN Staff_ID FORMAT A14 HEADING 'Staff ID' ;
COLUMN Task_ID FORMAT A12 HEADING 'Task ID';
COLUMN Task_Desc FORMAT A34 HEADING 'Task Desc';
COLUMN Shelter_ID FORMAT A10 HEADING 'Shelter ID' ;

ACCEPT Task_Category VARCHAR
PROMPT 'Please enter task category ('MEDICAL', 'CLEANING',
'LOGISTICS', 'FOOD_SUPLY', 'SECURITY', 'DONATION')

BREAK ON Task_Category ON Task_Desc ON Task_ID ON Shelter_ID ON
Staff_ID;

SELECT T.Task_Category, T.Task_Desc, T.Task_ID, S.Staff_ID,
SH.Shelter_ID, TaskDate
FROM Staff S, Task T, Workload W, Shelter SH
WHERE S.Staff_ID = W.Staff_ID AND T.Task_ID = W.Task_ID AND
SH.Shelter_ID = W.Shelter_ID AND Task_Category = '&T_Catgy'
ORDER BY T.Task_ID, TaskDate;
CLEAR COLUMNS;
CLEAR BREAKS;
```

Sample Output:

```
SQL> @"C:\Users\Sereneee\Desktop\QUERY 2.txt";
SP2-0003: Ill-formed ACCEPT command starting as VARCHAR
'Please enter task category ('MEDICAL', 'CLEANING', 'LOGISTICS', 'FOOD_SUPLY', 'SECURITY', 'DONATION')
Enter value for t_catgy: FOOD_SUPPLY
old   3: WHERE S.Staff_ID = W.Staff_ID AND T.Task_ID = W.Task_ID AND SH.Shelter_ID = W.Shelter_ID AND Task_Category = '&T_Catgy'
new   3: WHERE S.Staff_ID = W.Staff_ID AND T.Task_ID = W.Task_ID AND SH.Shelter_ID = W.Shelter_ID AND Task_Category = 'FOOD_SUPPLY'

TASK_CATEGORY   Task Desc                          Task ID     Staff ID      Shelter ID TASKDATE
--------------- ---------------------------------- ----------- ------------- ---------- -----------
FOOD_SUPPLY     DISTRIBUTE FOOD                    T08         S37           SH-04      22/APR/2019
                                                               S22           SH-07      21/JUL/2019
                DISTRIBUTE DAILY SUPPLIES          T09         S19           SH-01      25/APR/2017
                                                               S32           SH-09      02/JUN/2018
                DISTRIBUTE WOMENS SUPPLIES         T10         S38           SH-01      13/APR/2017
                                                               S08           SH-09      04/JUN/2017
                                                               S08           SH-03      24/JAN/2018
                STOCK COUNT                        T11         S31           SH-04      18/MAY/2017
                                                               S31           SH-10      27/MAR/2018
                                                               S20           SH-01      29/MAR/2019
                                                               S37           SH-10      02/JUL/2019

11 rows selected.

SQL>
```

6.2.3 Query 3: Victims details based on the input shelter ID and year.

Purpose: The purpose of this query is to let users to input a year, a shelter_ID and list out the details of the victims, enroll date based on the year and shelter_ID.

SQL Statement:

```
COLUMN Victim_ID FORMAT A12 HEADING 'Victim ID' ;
COLUMN Shelter_ID FORMAT A10 HEADING 'Shelter ID' ;
COLUMN Victim_Name HEADING 'Victim Name';
COLUMN Enrol_Date FORMAT A14 HEADING 'Enrol Date';

ACCEPT E_Year INT
PROMPT 'Please enter year (2016 - 2019)

ACCEPT S_ID VARCHAR
PROMPT 'Please enter a shelter ID (SH-01 - SH-10)

BREAK ON Shelter_ID SKIP PAGE ON Victim_ID  SKIP 1 ON Victim_Name ;

SELECT Shelter_ID, V.Victim_ID, Victim_Name, Enrol_Date
FROM Shelter_Allocation SA, Victim V
WHERE SA.Victim_ID = V.victim_ID AND Shelter_ID = '&S_ID' AND
Enrol_Date BETWEEN '01/Jan/&E_year' AND '31/Dec/&E_year'
ORDER BY Shelter_ID, V.victim_id;
CLEAR COLUMNS;
CLEAR BREAKS;
```

Sample Output:

```
SQL> @"C:\Users\Sereneee\Desktop\query 3.txt";
SP2-0003: Ill-formed ACCEPT command starting as INT
'Please enter year (2016 - 2019)
SP2-0003: Ill-formed ACCEPT command starting as VARCHAR
'Please enter a shelter ID (SH-01 - SH-10)
Enter value for s_id: SH-02
Enter value for e_year: 2019
Enter value for e_year: 2019
old   3: WHERE SA.Victim_ID = V.victim_ID AND Shelter_ID = '&S_ID' AND Enrol_Date BETWEEN '01/Jan/&E_year' AND '31/Dec/&E_year'
new   3: WHERE SA.Victim_ID = V.victim_ID AND Shelter_ID = 'SH-02' AND Enrol_Date BETWEEN '01/Jan/2019' AND '31/Dec/2019'

Shelter ID Victim ID    Victim Name                    Enrol Date
---------- ------------ ------------------------------ --------------
SH-02      V-013        Regan Gantlett                 02/SEP/2019

           V-020        Stanly Treagus                 28/MAR/2019

           V-025        Ganny Kemetz                   18/FEB/2019

           V-050        Rurik McWilliams               14/APR/2019


SQL>
```

6.2.4 Report 1: Cash Donation Report

Purpose: The purpose of this report is to list out who donated cash and total up the cash based on the year that input by the users.

PL/SQL code:

```
TTITLE LEFT ================================= SKIP 1 -
LEFT 'CASH DONATION REPORT ' SKIP 1-
LEFT ================================= SKIP 1 -
LEFT _DATE SKIP 1-
Right 'Page: 'FORMAT 999 SQL.PNO SKIP 2

COLUMN Donation_Type FORMAT A14 HEADING 'Donation Type';
COLUMN Donation_No FORMAT A12 HEADING 'Donation No' ;
COLUMN Amount FORMAT $9999.99 HEADING 'Amount (RM)';
COLUMN Donor_ID FORMAT A12 HEADING 'Donor ID' ;
COLUMN Donation_Date FORMAT A14 HEADING 'Donation Date';

ACCEPT D_Year INT
PROMPT 'Please enter a year(2016 - 2019): '

COMPUTE SUM LABEL 'Total' OF amount ON donation_type;

BREAK ON Donation_Type ON Donor_ID SKIP 1 ON Name;
SELECT DT.Donation_Type, DN.Donor_ID, DN.Name, DT.Donation_No,
DT.Amount, DT.Donation_Date
FROM Donor DN, Donation DT
WHERE DN.Donor_ID = DT.Donor_ID AND DT.Donation_Type = 'Cash' AND
Donation_Date BETWEEN '01/Jan/&D_Year' AND '31/Dec/&D_Year'
ORDER BY Donation_Date ;

CLEAR BREAKS;
CLEAR COMPUTES;
CLEAR COLUMNS;
TTITLE OFF;
```

Sample Output:

```
SQL> @"C:\Users\Sereneee\Desktop\QUE REPORT 1.txt";
SP2-0268: linesize option not a valid number
SP2-0003: Ill-formed ACCEPT command starting as INT
'Please enter a year(2016 - 2019): '
Enter value for d_year: 2019
Enter value for d_year: 2019
old   3: WHERE DN.Donor_ID = DT.Donor_ID AND DT.Donation_Type = 'Cash' AND Donation_Date BETWEEN '01/Jan/&D_Year' AND '31/Dec/&D_Year'
new   3: WHERE DN.Donor_ID = DT.Donor_ID AND DT.Donation_Type = 'Cash' AND Donation_Date BETWEEN '01/Jan/2019' AND '31/Dec/2019'

================================
CASH DONATION REPORT
================================
20/DEC/2019

                                                        Page:    1

Donation Type  Donor ID     NAME                      Donation No  Amount (RM) Donation Date
-------------- ------------ ------------------------- ------------ ----------- --------------
Cash           D14          Maison Atheis             DN-022           $270.00 27/JAN/2019

               D23          Evaleen Stilliard         DN-024           $240.00 21/MAY/2019

               D15          Coraline Acutt            DN-027           $322.00 31/MAY/2019

               D17          Lina Beckhouse            DN-028           $225.00 06/JUN/2019

               D23          Evaleen Stilliard         DN-030           $383.00 24/JUN/2019

               D05          Alina Carreck             DN-031           $331.00 26/JUN/2019

               D26          Lorilee Sturgess          DN-033             $9.00 17/JUL/2019

               D18          Teresa Gemmell            DN-034           $191.00 18/JUL/2019

               D07          Nerta Fadian              DN-037            $94.00 05/AUG/2019

               D24          Pip Deaville              DN-039           $134.00 18/AUG/2019

               D28          Sergent Klainer           DN-040           $411.00 29/AUG/2019

               D16          Ban Maling                DN-042           $394.00 09/SEP/2019

               D12          Jerry Maddick             DN-043            $38.00 17/SEP/2019

               D25          Gabriele Iddy             DN-045            $91.00 30/OCT/2019

               D13          Rolfe Braim               DN-048           $292.00 26/NOV/2019

************** ************ ************************                -----------
Total                                                                 $3425.00

15 rows selected.

SQL>
```

6.2.5 Report 2: Victim Health Status Report

Purpose: The purpose of this report is to know the health status of victims and total up the quantity of victims in each status based on the input of year by the users.

PL/SQL code:

```
TTITLE LEFT ================================== SKIP 1 -
LEFT 'VICTIM HEALTH STATUS REPORT' SKIP 1-
LEFT ================================== SKIP 1-
LEFT_DATE SKIP 1-
Right 'Page: 'FORMAT 999 SQL.PNO SKIP 2

COLUMN Victim_ID FORMAT A12 HEADING 'Victim ID' ;
COLUMN Examined_Date FORMAT A14 HEADING 'Examined Date' ;
COLUMN Victim_Name HEADING 'Victim Name';

ACCEPT E_Year INT
PROMPT 'Please enter year(2016 - 2019): '

COMPUTE COUNT LABEL 'Total' OF Victim_ID ON status;

BREAK ON Status SKIP PAGE ON Victim_ID;
SELECT ME.Status, V.Victim_ID, Victim_Name, Sickness, Examined_Date
FROM Medical_Examination ME, Victim V
WHERE V.Victim_ID = ME.Victim_ID AND Examined_Date BETWEEN
'01/Jan/&E_Year' AND '31/Dec/&E_Year'
ORDER BY ME.Status, ME.Examined_Date, Victim_ID;

CLEAR BREAKS;
CLEAR COMPUTES;
CLEAR COLUMNS;
TTITLE OFF;
```

Sample Output:

```
SQL> @"C:\Users\Sereneee\Desktop\QUE REPORT 2.txt";
SP2-0003: Ill-formed ACCEPT command starting as INT
'Please enter year(2016 - 2019): '
Enter value for e_year: 2018
Enter value for e_year: 2018
old   3: WHERE V.Victim_ID = ME.Victim_ID AND Examined_Date BETWEEN '01/Jan/&E_Year' AND '31/Dec/&E_Year'
new   3: WHERE V.Victim_ID = ME.Victim_ID AND Examined_Date BETWEEN '01/Jan/2018' AND '31/Dec/2018'

================================
VICTIM HEALTH STATUS REPORT
================================
20/DEC/2019

                                                          Page:    1

STATUS   Victim ID    Victim Name                    SICKNESS                    Examined Date
-------- ------------ ------------------------------ --------------------------- -------------
Good     V-013        Regan Gantlett                                             08/JAN/2018
         V-024        Francis Camois                                             19/MAR/2018
         V-020        Stanly Treagus                                             10/APR/2018
         V-031        Miran Bachshell                                            23/APR/2018
         V-049        Willdon Liccardi                                           15/MAY/2018
         V-044        Enrica Ezele                                               25/MAY/2018
         V-034        Mikel Padefield                                            31/JUL/2018
         V-025        Ganny Kemetz                                               15/SEP/2018
         V-041        Kathi Dunnan                                               26/NOV/2018
         V-028        Ruddie Trigg                                               27/NOV/2018
         V-034        Mikel Padefield                                            27/NOV/2018
         V-044        Enrica Ezele                                               28/DEC/2018
******** ------------
Total              12
```

```
================================
VICTIM HEALTH STATUS REPORT
================================
20/DEC/2019

                                                          Page:    2

STATUS   Victim ID    Victim Name                    SICKNESS                    Examined Date
-------- ------------ ------------------------------ --------------------------- -------------
Moderate V-019        Carolina Reide                 Typhoid Fever               05/JAN/2018
         V-049        Willdon Liccardi               Dengue fever                26/JAN/2018
         V-034        Mikel Padefield                Measles                     30/JAN/2018
         V-008        Datha Dartan                   Malaria                     14/FEB/2018
         V-002        Jock Daudray                   Acute respiratory infection 19/FEB/2018
         V-033        Alisha Veness                  Diarrhea diseases           17/MAR/2018
         V-018        Sallee Colnett                 Diarrhea diseases           09/APR/2018
         V-044        Enrica Ezele                   Dengue fever                01/MAY/2018
```

```
        V-036        Bianka Scane                Diarrhea diseases           05/MAY/2018
        V-032        Margarete Turford           Diarrhea diseases           28/MAY/2018
        V-005        Pepito Shimon               Diarrhea diseases           09/JUN/2018
        V-035        Erda Gittins                Viral hepatitis             11/JUN/2018
        V-048        Elvis Cottesford            Viral hepatitis             12/JUN/2018
        V-039        Pippo Putland               Asthma                      19/JUN/2018
        V-013        Regan Gantlett              Typhoid Fever               01/JUL/2018
        V-050        Rurik McWilliams            Measles                     04/AUG/2018
        V-032        Margarete Turford           Measles                     28/AUG/2018
        V-009        Kearney Cameron             Asthma                      02/SEP/2018
        V-027        Eveline Bruneton            Typhoid Fever               08/OCT/2018
        V-046        Teddi Ethelstone            Diarrhea diseases           08/OCT/2018
        V-049        Willdon Liccardi            Asthma                      08/OCT/2018
        V-001        Abie Olrenshaw              Dengue fever                16/NOV/2018
        V-005        Pepito Shimon               Viral hepatitis             21/NOV/2018
        V-047        Barr Kelby                  Measles                     25/NOV/2018
        V-050        Rurik McWilliams            Acute respiratory infection 28/DEC/2018
        V-021        Blake Hassekl               Acute respiratory infection 31/DEC/2018
******** ------------
Total             26
```

```
================================
VICTIM HEALTH STATUS REPORT
================================
20/DEC/2019

                                                        Page:    3

STATUS   Victim ID    Victim Name                   SICKNESS                    Examined Date
-------- ------------ ----------------------------- --------------------------- --------------
Severe   V-045        Sandro Suermeiers             Acute respiratory infection 07/JAN/2018
         V-016        Nichols O'Luby                Measles                     09/JAN/2018
         V-020        Stanly Treagus                Dengue fever                11/JAN/2018
         V-041        Kathi Dunnan                  Asthma                      14/JAN/2018
         V-017        Jonah Bartolomeu              Viral hepatitis             15/JAN/2018
         V-018        Sallee Colnett                Typhoid Fever               03/FEB/2018
         V-001        Abie Olrenshaw                Viral hepatitis             17/MAR/2018
         V-041        Kathi Dunnan                  Typhoid Fever               17/MAR/2018
         V-031        Miran Bachshell               Asthma                      09/APR/2018
         V-011        Rossie Taberer                Chest infection             05/MAY/2018
         V-015        Gawen Aylwin                  Dengue fever                09/MAY/2018
         V-034        Mikel Padefield               Acute respiratory infection 09/MAY/2018
         V-043        Manuel Tregoning              Diarrhea diseases           09/MAY/2018
         V-013        Regan Gantlett                Diarrhea diseases           20/MAY/2018
         V-020        Stanly Treagus                Asthma                      07/JUN/2018
         V-042        Merwyn Jencken                Malaria                     14/JUN/2018
         V-031        Miran Bachshell               Asthma                      15/JUN/2018
         V-006        Dave Bohden                   Chest infection             01/JUL/2018
         V-038        Erda Branston                 Malaria                     10/JUL/2018
         V-019        Carolina Reide                Measles                     12/JUL/2018

         V-001        Abie Olrenshaw                Typhoid Fever               16/JUL/2018
                      Abie Olrenshaw                Asthma                      26/JUL/2018
         V-008        Datha Dartan                  Measles                     12/AUG/2018
         V-022        Darrick Moubray               Dengue fever                17/AUG/2018
         V-011        Rossie Taberer                Measles                     23/AUG/2018
         V-022        Darrick Moubray               Typhoid Fever               18/SEP/2018
         V-037        Florenza Alesin               Asthma                      08/OCT/2018
         V-011        Rossie Taberer                Acute respiratory infection 09/OCT/2018
         V-012        Roxanne Doohan                Diarrhea diseases           19/OCT/2018
         V-017        Jonah Bartolomeu              Measles                     28/NOV/2018
         V-022        Darrick Moubray               Acute respiratory infection 29/NOV/2018
         V-032        Margarete Turford             Acute respiratory infection 30/DEC/2018
         V-015        Gawen Aylwin                  Measles                     31/DEC/2018
******** ------------
Total             33

71 rows selected.
```

6.3 Lee Shu Ern

6.3.1 Query 1: Staff details

Purpose: The purpose of this query is to find out the staff details that carry more than 2 tasks.

SQL Statement:

```
COLUMN Staff_ID FORMAT A14 HEADING 'Staff ID' ;
COLUMN CountTask FORMAT 999 HEADING 'Count Task';

BREAK ON staff_id ON name ON position SKIP 1;

SELECT w.staff_id, name, position, count(t.task_id) AS CountTask
```

```
FROM staff s, workload w, task t
WHERE w.staff_id = s.staff_id AND t.task_id = w.task_id
GROUP BY  w.staff_id, name, position
HAVING count(t.task_id) > 2
ORDER BY w.staff_id;

CLEAR BREAKS;
CLEAR COLUMN;
```

Sample Output:

```
SQL> @"C:\Users\Sereneee\Desktop\shuern q1.txt";

Staff ID       NAME                           POSITION        Count Task
-------------- ------------------------------ --------------- ----------
S05            Jeana Drohan                   Nurse                    3

S07            Antonin Spelwood               Facilitator              4

S08            Beckie Roberti                 Staff Member             4

S13            Sibby Order                    Security                 5

S16            Itch Ackenhead                 Paramedic                6

S18            Randolf Riddock                Paramedic                8

S23            Irvine O'Donoghue              Physicians               4

S26            Edythe Padmore                 Nurse                    3

S27            Christean Metschke             Supervisor               3

S28            Kaine Pinnock                  Staff Member             3

S30            Gail Juden                     Physicians               3

S31            Ernesto Coad                   Staff Member             5

S33            Mel Janssens                   Staff Member             3

S34            Stanford Catt                  Security                 3

S35            Caritta Loftus                 Physicians               3

S36            Allix Vanstone                 Supervisor               3

16 rows selected.
```

6.3.2 Query 2: Food details

Purpose: The purpose of this query is to ensure that the food donated in each shelter has not expired.

SQL Statement:

```
COLUMN Shelter_ID FORMAT A10 HEADING 'Shelter ID' ;
COLUMN FS_ID FORMAT A18 HEADING 'Food and Supply ID' ;
COLUMN Item_Desc FORMAT A30 HEADING 'Item Desc' ;
COLUMN Sent_Quantity HEADING 'Sent Quantity';
COLUMN Expiry_Date FORMAT A12 HEADING 'Expire Date';

ACCEPT s_id VARCHAR()
 PROMPT 'Enter Shelter ID (SH-01 - SH-10):

BREAK ON shelter_id ON item_desc SKIP 1 ON fs_id;

SELECT s.shelter_id, f. fs_id, item_desc, sent_quantity, expiry_date
FROM shelter s, food_and_supply_allocation fa, food_and_supply f
WHERE s.shelter_id = fa.shelter_id AND fa.fs_id = f.fs_id AND
expiry_date > sysdate AND s.shelter_id = '&s_id'
ORDER BY s.shelter_id, f.fs_id, expiry_date;

CLEAR BREAKS;
CLEAR COLUMNS;
```

Sample Output:

```
SQL> @"C:\Users\Sereneee\Desktop\shuern q2.txt";
SP2-0003: Ill-formed ACCEPT command starting as VARCHAR()
'Enter Shelter ID (SH-01 - SH-10):
Enter value for s_id: SH-01
old   3: WHERE s.shelter_id = fa.shelter_id AND fa.fs_id = f.fs_id AND expiry_date > sysdate AND s.shelter_id = '&s_id'
new   3: WHERE s.shelter_id = fa.shelter_id AND fa.fs_id = f.fs_id AND expiry_date > sysdate AND s.shelter_id = 'SH-01'

Shelter ID Food and Supply ID Item Desc                     Sent Quantity Expire Date
---------- ------------------ ------------------------------ ------------- ------------
SH-01      FS013              Canned food                               25 25/NOV/2021

           FS032              Rice bag (10kg)                           15 10/NOV/2020

           FS053              Biscuits                                  25 31/AUG/2020

           FS055              Drinks                                    35 25/DEC/2019
           FS056                                                        25 25/DEC/2020

           FS059              Instant noodles                           25 31/DEC/2019


5 rows selected.
```

6.3.3 Query 3: Donor details

Purpose: The purpose of this query is to let user to input donor id and find the donation details based on the donor id in 2019.

SQL Statement:

```
COLUMN Donor_ID FORMAT A12 HEADING 'Donor ID' ;
COLUMN Tel_No HEADING 'Tel No';
COLUMN Donation_No FORMAT A12 HEADING 'Donation No' ;
COLUMN Donation_Type FORMAT A14 HEADING 'Donation Type';
COLUMN Donation_Date FORMAT A14 HEADING 'Donation Date';
COLUMN Amount FORMAT $9999.99 HEADING 'Amount (RM)';

ACCEPT donor_id VARCHAR()
PROMPT 'Enter donor ID (D01 - D02):';

BREAK ON donor_id  ON name SKIP 2 ON gender ON tel_no ;

SELECT d.donor_id, name, gender, tel_no, donation_no, donation_type,
amount, donation_date
FROM donor d, donation do
WHERE d.donor_id = do.donor_id AND d.donor_id = '&donor_id' AND
donation_date BETWEEN '01/Jan/2019' AND '31/Dec/2019'
ORDER BY d.donor_id, donation_no;

CLEAR BREAK;
CLEAR COLUMNS;
CLEAR COMPUTE;
```

Sample Output:

```
SQL> @"C:\Users\Sereneee\Desktop\shuern q3.txt";
SP2-0003: Ill-formed ACCEPT command starting as VARCHAR()
'Enter donor ID (D01 - D02):'
Enter value for donor_id: D03
old   3: WHERE d.donor_id = do.donor_id AND d.donor_id = '&donor_id' AND donation_date BETWEEN '01/Jan/2019' AND '31/Dec/2019'
new   3: WHERE d.donor_id = do.donor_id AND d.donor_id = 'D03' AND donation_date BETWEEN '01/Jan/2019' AND '31/Dec/2019'

Donor ID     NAME                      GENDER     Tel No                    Donation No  Donation Type  Amount (RM) Donation Date
------------ ------------------------- ---------- ------------------------- ------------ -------------- ----------- --------------
D03          Paola Langstrath          F          636-450-9825              DN-038       Food                       07/AUG/2019
```

6.3.4 Report 1: Supply Donation Report

Purpose: The purpose of this report is to list out the total task and task details that carry out in the shelter that input by the user in 2017.

PL/SQL code:

```
PROMPT 'Check for the number of task that had been done in 2017 at
selected shelter';
ACCEPT shelter_id PROMPT 'Enter shelter id (SH-01 to SH-10):';

COLUMN Shelter_ID FORMAT A10 HEADING 'Shelter ID' ;
COLUMN Task_ID FORMAT A12 HEADING 'Task ID';
COLUMN Task_Desc FORMAT A34 HEADING 'Task Desc';

TTITLE LEFT ============================== SKIP 1 -
LEFT  _DATE SKIP 1-
LEFT ============================== SKIP 1-
LEFT 'Shelter Details' SKIP 2 -
       LEFT 'Task handled at ' '&shelter_id' ' in 2017 'SKIP 1 -
LEFT 'Page:' FORMAT 999 SQL.PNO SKIP ;

BREAK ON shelter_id SKIP PAGE  ON task_desc SKIP 1;
COMPUTE COUNT LABEL 'Total' OF task_id ON shelter_id;

SELECT s.shelter_id, t.task_id, task_desc, taskdate
FROM workload w, task t, shelter s
WHERE w.shelter_id = s.shelter_id AND w.task_id = t.task_id AND
taskdate BETWEEN '01/Jan/2017' AND '31/Dec/2017' AND s.shelter_id =
'&shelter_id'
ORDER  BY shelter_id, task_id;

CLEAR BREAKS;
CLEAR COLUMN;
CLEAR COMPUTE;
TTITLE OFF;
```

Sample Output:

```
SQL> @"C:\Users\Sereneee\Desktop\shuern r1.txt";
'Check for the number of task that had been done in 2017 at selected shelter'
Enter shelter id (SH-01 to SH-10):SH-10
old   3: WHERE w.shelter_id = s.shelter_id AND w.task_id = t.task_id AND taskdate BETWEEN '01/Jan/2017' AND '31/Dec/2017' AND s.shelter_id = '&shelter_id'
new   3: WHERE w.shelter_id = s.shelter_id AND w.task_id = t.task_id AND taskdate BETWEEN '01/Jan/2017' AND '31/Dec/2017' AND s.shelter_id = 'SH-10'

==============================
20/DEC/2019
==============================
Shelter Details

Task handled at SH-10 in 2017
Page:   1
Shelter ID Task ID     Task Desc                       TASKDATE
---------- ----------- ------------------------------- -----------
SH-10      T15         VICTIM FACILITATING             03/MAR/2017

           T16         PROVIDE FIRST AID TREATMENT     14/DEC/2017

           T19         MEDICAL CHECKUP                 10/JUL/2017

********** ----------- *******************************
Total               3
```

6.3.5 Report 2: Supply Donation Report

Purpose: The purpose of this report is to list out the details of donor who donated supply and total up the type of supply and the total quantity of supply donated by donor based on the input of year by the users.

PL/SQL code:

```
TTITLE LEFT  ================================== SKIP 1 -
LEFT  _DATE SKIP 1-
LEFT  ================================== SKIP 1-
LEFT 'Donation Details' SKIP 1 -
       LEFT 'Type of donation: Supply' SKIP 0 -
LEFT 'Page:' FORMAT 999 SQL.PNO SKIP2;

COLUMN Donation_No FORMAT A12 HEADING 'Donation No' ;
COLUMN Donation_Date FORMAT A14 HEADING 'Donation Date';
COLUMN FS_ID FORMAT A18 HEADING 'Food and Supply ID' ;
COLUMN Item_Desc FORMAT A30 HEADING 'Item Desc' ;
COLUMN Quantity FORMAT 999 HEADING 'Quantity';
COLUMN Donor_ID FORMAT A12 HEADING 'Donor ID' ;

ACCEPT D_Year INT
PROMPT 'Please enter a year(2016 - 2019): ';
BREAK ON donor_id ON donation_no SKIP PAGE ON name ON donation_date;
COMPUTE COUNT LABEL 'Total Item' OF item_desc ON donation_no;
COMPUTE SUM LABEL 'Total' OF quantity ON donation_no;

SELECT do.donor_id, name, d.donation_no, donation_date, di.fs_id,
item_desc, quantity
FROM food_and_supply f, donation d, donation_item di, donor do
WHERE di.donation_no=d.donation_no AND d.donor_id = do.donor_id AND
di.fs_id = f.fs_id AND donation_type = 'Supply'
AND donation_date BETWEEN '01/Jan/&D_Year' AND '31/Dec/&D_Year'
order by d.donation_no, di.fs_id;

CLEAR BREAKS;
```

```
CLEAR COMPUTE;
CLEAR COLUMNS;
TTITLE OFF;
```

Sample Output:

```
SQL> @"C:\Users\Sereneee\Desktop\shuern r2.txt";
SP2-0003: Ill-formed ACCEPT command starting as INT
'Please enter a year(2016 - 2019): '
Enter value for d_year: 2018
Enter value for d_year: 2018
old   4: AND donation_date BETWEEN '01/Jan/&D_Year' AND '31/Dec/&D_Year'
new   4: AND donation_date BETWEEN '01/Jan/2018' AND '31/Dec/2018'

==================================
20/DEC/2019
==================================
Donation Details
Type of donation: Supply
Page:   1

Donor ID     NAME                      Donation No  Donation Date  Food and Supply ID Item Desc                      Quantity
------------ ------------------------- ------------ -------------- ------------------ ------------------------------ --------
D28          Sergent Klainer           DN-021       25/DEC/2018    FS006              First-aid items                      52
                                                                     FS007              Pads and tampons                     60
                                                                     FS008              Blankets                             40
             ************************* ************ ************** ------------------------------ --------
                                       Total Item                                                                3
                                       Total                                                                   152
```

References

ORACLE Help Center, (2019). SQL*Plus® User's Guide and Reference. Available from: https://docs.oracle.com/cd/E11882_01/server.112/e16604/ch_twelve014.htm#SQPUG035 [Accessed on 18 Dec 2019]

Oracle, (2019). SQL*Plus® User's Guide and Reference. Available from: https://docs.oracle.com/cd/B19306_01/server.102/b14357/ch6.htm [Accessed on 15 Dec 2019]

StackOverflow, (2019). ACCEPT statement in Oracle PL SQL. Available from: https://stackoverflow.com/questions/35441790/accept-statement-in-oracle-pl-sql [Accessed on 17 Dec 2019]

w3schools.com, (2019). SQL FOREIGN KEY Constraint. Available from https://www.w3schools.com/sql/sql_foreignkey.asp [Accessed on 1 Dec 2019]