



(Company No. 101067-P)

الجامعة الإسلامية العالمية ماليزيا  
INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA  
يُونَيْتِيْ اِسْلَامٌ اَنْتَارَا بَعْثًا مِّلِيْسِيَا

*Garden of Knowledge and Virtue*

KULLIYAH OF INFORMATION COMMUNICATION AND TECHNOLOGY

INFO 2103 DATABASE PROGRAMMING

SECTION 3

GROUP PROJECT ASSIGNMENT:

**ZOO TICKETING SYSTEM**

LECTURER'S NAME: DR. ZAINATUL SHIMA BINTI ABDULLAH

PREPARED BY:

NAME	MATRIC NO.
AISHAH NABILAH BINTI MUHAMAD RIDZUAN	1625084
NURUL AIN SHAHIRAH BINTI NORFRIZAT	1818540
NUR AFIFAH BINTI MOHAMAD AZAHAR	1810016
SITI NOOR NAMIRAH HASINAH BINTI NOOR RAZIFF	1816818

DUE DATE

17<sup>TH</sup> DECEMBER 2019

# CONTENT

<b>1.0 Background Project.....</b>	<b>1</b>
<b>1.2 Project Significant.....</b>	<b>1</b>
<b>1.3 Zoo Ticketing System Entity Relationship Diagram (ERD).....</b>	<b>2</b>
<b>2.0 Data dictionary for the ER Diagram – Zoo Ticketing System.....</b>	<b>3</b>
CUSTOMER.....	3
STAFF.....	3
TICKET.....	4
PROMOTION.....	4
PAYMENT.....	5
<b>2.1 Create table.....</b>	<b>6</b>
Table Customer:.....	6
Table Staff:.....	7
Table Ticket:.....	8
Table Promotion:.....	8
Table Payment:.....	9
<b>2.2 Insert data for all table.....</b>	<b>10</b>
Customer Data:.....	10
Staff Data:.....	11-12
Ticket Data:.....	13-16
Promotion Data:.....	16-17
Payment Data:.....	18-19
<b>3.0 SQL Statements (Add, Delete, Modify).....</b>	<b>20</b>
Scenario 1:.....	21
Scenario 2:.....	21
Scenario 3:.....	21
Scenario 4:.....	21-22
Scenario 5:.....	23
Scenario 6:.....	23

<b>4.0 Create two procedures:</b> .....	24
Scenario 1:.....	24
Scenario 2:.....	25-27
<b>5.0 Create two functions:</b> .....	28
Scenario 1:.....	28-29
Scenario 2:.....	30-31
Member Contribution for all tasks.....	32
<b>6.0 Web Application for Zoo Ticketing System</b> .....	<b>33</b>
WelcomePage.html.....	33
CustomerDetails.html.....	33
ChooseTicket.html.....	34
PayTicket.html.....	34
StaffUpdateDetails.html.....	35
<b>7.0 Conclusion</b> .....	<b>35</b>
<b>8.0</b>	<b>35</b>
<b>References</b> .....	<b>Error!</b>
<b>Bookmark not defined.4</b>	



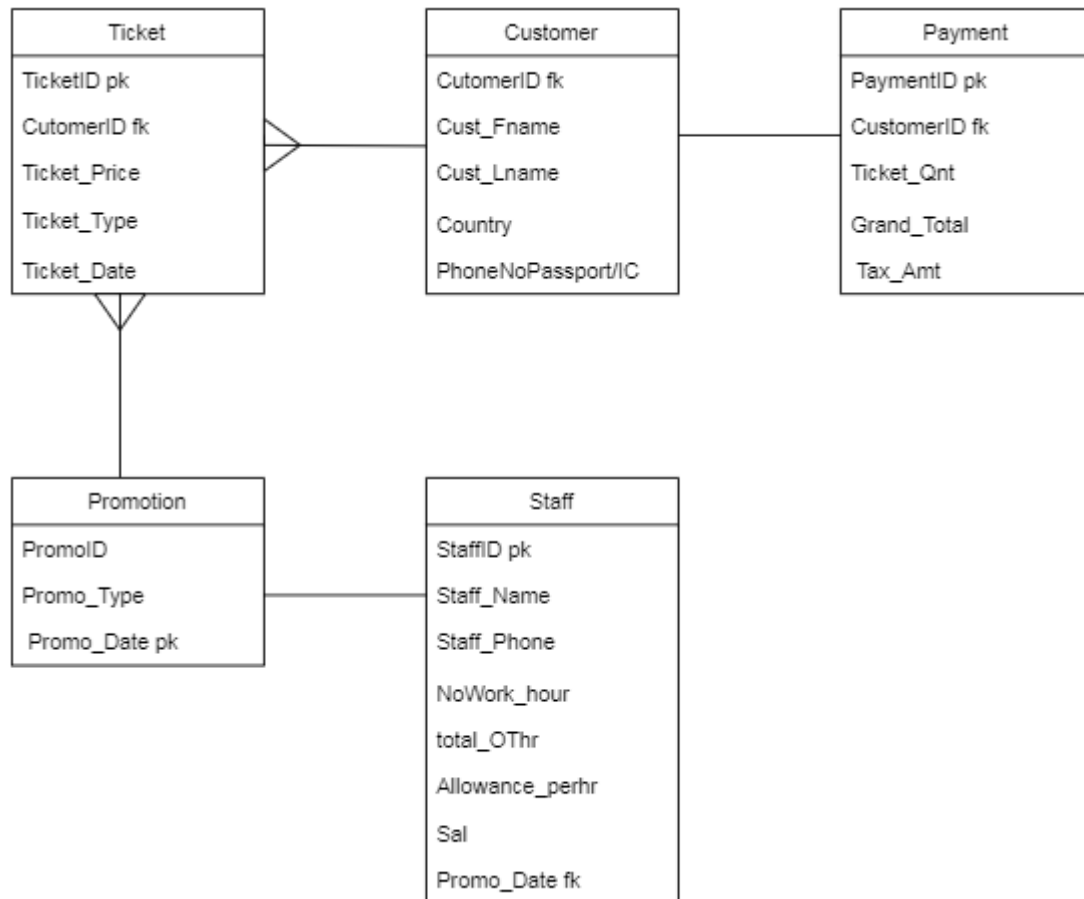
## **1.0 Background Project**

Our project title is "Zoo Ticketing System". The purpose of this database is to ease the process of storing, updating, add, calculating salary and promotions for the staff and customers that works and comes to the zoo. Everyday a lot of people come to the zoo and there are specific dates and privilege package that is offered by the zoo. For example, there is an adult zoo promotion trip for New Year where you get an adult ticket per head for only Rm26, the original price on a normal day is Rm40 and also with many more upcoming promotion. The usage of this database system also helps to minimize the wastage of paper to store a lot of data at once.

## **1.2 Project Significant**

Managing business and ensuring the business goes well requires a database management system. This database management system is important because it manages data efficiently. For example, users (admin or business owner) will be able to see their business from a different perspective. Database management system also allows users to perform multiple tasks and see unique features to improve their business. This is because, with database management system, business owners will be able to see the sales per mon that their company able to make. They also will able to observe on which product or promotion that brings more customers and more profits to the company. The database management system also stores, organizes and manages a large amount of information within a single software application.

### 1.3 Zoo Ticketing System Entity Relationship Diagram (ERD)



## 2.0 Data dictionary for the ER Diagram – Zoo Ticketing System

### CUSTOMER

ATTRIBUTE NAME	DATA TYPE	SIZE	NULL
CustomerID	Varchar2	8	No (Primary Key)
Cust_Fname	Varchar2	15	No
Cust_Lname	Varchar2	15	No
Country	Varchar2	12	No
PhoneNo	Number	11	No
custPassport_IC	Number	15	No

### STAFF

ATTRIBUTE NAME	DATA TYPE	SIZE	NULL
StaffID	Varchar2	8	No (Primary Key)
Staff_Name	Varchar2	20	No
Staff_PhoneNo	Number	11	No
Work_hour	Number	15	No
total_OThr	Number	5	No
Allowance_perhr	Number	(5,2)	No
Sal	Number	10	No

## TICKET

ATTRIBUTE NAME	DATA TYPE	SIZE	NULL
TicketID	Number	5	No (Primary Key)
CustomerID	Varchar2	8	No (Foreign Key)
Ticket_Date	Date		No
Ticket_Type	Varchar2	20	No
Ticket_Price	Number	(5,2)	No

## PROMOTION

ATTRIBUTE NAME	DATA TYPE	SIZE	NULL
PromoID	Varchar2	8	No
Promo_Type	Varchar2	30	No
Promo_Date	Date		No (Primary Key)



## **PAYMENT**

<b>ATTRIBUTE NAME</b>	<b>DATA TYPE</b>	<b>SIZE</b>	<b>NULL</b>
PaymentID	Varchar2	8	No (Primary Key)
CustomerID	Varchar2	8	No (Foreign Key)
Ticket_Qnt	Number	3	No
Tax_Amt	Number	(5,2)	No
Grand_Total	Number	(5,2)	No

## 2.1 Create table

Drop Table Customer cascade constraints;  
Drop Table Staff cascade constraints;  
Drop Table Ticket cascade constraints;  
Drop Table Promotion cascade constraints;  
Drop Table Payment cascade constraints;

### Table Customer:

```
SQL> Create Table Customer (  
2 CustomerID      Varchar2(8),  
3 Cust_Fname      Varchar2(15),  
4 Cust_Lname      Varchar2(15),  
5 Country         Varchar2(12),  
6 PhoneNo         Number(11),  
7 custPassport_IC Number(14),  
8 constraint Customer_CustomerID_PK Primary Key (CustomerID));
```

Table created.

SQL>

SQL> desc customer

Name	Null?	Type
CUSTOMERID	NOT NULL	VARCHAR2(8)
CUST_FNAME		VARCHAR2(15)
CUST_LNAME		VARCHAR2(15)
COUNTRY		VARCHAR2(12)
PHONENO		NUMBER(11)
CUSTPASSPORT_IC		NUMBER(14)

```
Create Table Customer (  
CustomerID      Varchar2(8),  
Cust_Fname      Varchar2(15),  
Cust_Lname      Varchar2(15),  
Country         Varchar2(12),  
PhoneNo         Number(11),  
custPassport_IC Number(14),  
constraint Customer_CustomerID_PK Primary Key (CustomerID));
```

## Table Staff:

```
SQL> Create Table Staff (  
2  StaffID          Number(5),  
3  Staff_Name       Varchar2(20),  
4  Staff_PhoneNo    Number(11),  
5  Promo_Date       Date,  
6  Work_hour        Number(15),  
7  total_OThr       Number(5),  
8  Allowance_perhr   Number(5,2),  
9  Sal              Number(10),  
10 constraint Staff_StaffID_PK Primary Key (StaffID),  
11 constraint Staff_Promo_Date_FK Foreign Key (Promo_Date) References Promotion);
```

Table created.

```
SQL> desc Staff
```

Name	Null?	Type
STAFFID	NOT NULL	NUMBER(5)
STAFF_NAME		VARCHAR2(20)
STAFF_PHONENO		NUMBER(11)
PROMO_DATE		DATE
WORK_HOUR		NUMBER(15)
TOTAL_OTHR		NUMBER(5)
ALLOWANCE_PERHR		NUMBER(5,2)
SAL		NUMBER(10)

Create Table Staff (

StaffID        Number(5),

Staff\_Name     Varchar2(20),

Staff\_PhoneNo Number(11),

Work\_hour      Number(15),

total\_OThr     Number(5),

Allowance\_perhr    Number(5,2),

Sal            Number(10),

Promo\_Date date,

constraint Staff\_StaffID\_PK Primary Key (StaffID),

constraint Staff\_Promo\_Date\_FK Foreign Key (Promo\_Date) References Promotion);

## Table Ticket:

```
SQL> Create Table Ticket (  
2 TicketID          Number(5),  
3 CustomerID        Varchar2(8),  
4 Ticket_Date       Date,  
5 Ticket_Type       Varchar(20),  
6 Ticket_Price      Number(5,2),  
7 constraint Ticket_TicketID_PK Primary Key (TicketID),  
8 constraint Ticket_CustomerID_FK Foreign Key (CustomerID) References Customer);
```

Table created.

```
SQL> desc Ticket
```

Name	Null?	Type
TICKETID	NOT NULL	NUMBER(5)
CUSTOMERID		VARCHAR2(8)
TICKET_DATE		DATE
TICKET_TYPE		VARCHAR2(20)
TICKET_PRICE		NUMBER(5,2)

```
Create Table Ticket (  
TicketID          Number(5),  
CustomerID        Varchar2(8),  
Ticket_Date       Date,  
Ticket_Type       Varchar(20),  
Ticket_Price      Number(5,2),  
constraint Ticket_TicketID_PK Primary Key (TicketID),  
constraint Ticket_CustomerID_FK Foreign Key (CustomerID) References Customer);
```

## Table Promotion:

```
SQL> Create Table Promotion (  
2 PromoID           Varchar2(8),  
3 Promo_Type        Varchar(30),  
4 Promo_Date        Date,  
5 constraint Promotion_Promo_Date_PK Primary Key (Promo_Date));
```

Table created.

```
SQL> desc Promotion
```

Name	Null?	Type
PROMOID		VARCHAR2(8)
PROMO_TYPE		VARCHAR2(30)
PROMO_DATE	NOT NULL	DATE

```
Create Table Promotion (  
PromoID           Varchar2(8),  
Promo_Type        Varchar(30),  
Promo_Date        Date,  
constraint Promotion_Promo_Date_PK Primary Key (Promo_Date));
```

## Table Payment:

```
SQL> Create Table Payment (  
 2 PaymentID      Varchar2(8),  
 3 CustomerID     Varchar2(8),  
 4 Ticket_Qnt     Number(3),  
 5 Tax_Amt        Number(5,2),  
 6 Grand_Total    Number(5,2),  
 7 constraint Payment_PaymentID_PK Primary Key (PaymentID),  
 8 constraint Payment_CustomerID_FK Foreign Key (CustomerID) References Customer);
```

Table created.

SQL>

SQL> desc Payment

Name	Null?	Type
PAYMENTID	NOT NULL	VARCHAR2(8)
CUSTOMERID		VARCHAR2(8)
TICKET_QNT		NUMBER(3)
TAX_AMT		NUMBER(5,2)
GRAND_TOTAL		NUMBER(5,2)

```
Create Table Payment (  
PaymentID      Varchar2(8),  
CustomerID     Varchar2(8),  
Ticket_Qnt     Number(3),  
Tax_Amt        Number(5,2),  
Grand_Total    Number(5,2),  
constraint Payment_PaymentID_PK Primary Key (PaymentID),  
constraint Payment_CustomerID_FK Foreign Key (CustomerID) References Customer);
```

## 2.2 Insert data for all table

### Customer Data:

```
insert into Customer  
values('C100', 'Tisya', 'Amani', 'Malaysia', '0123456789', '921106');
```

```
insert into Customer  
values('C200', 'Fahim', 'Najmi', 'Malaysia', '0176565410', '991203');
```

```
insert into Customer  
values('C300', 'Johan', 'Ahmad', 'Malaysia', '0122566310', '971006');
```

```
insert into Customer  
values('C400', 'Aqil', 'Ali', 'Malaysia', '0165634100', '980602');
```

```
insert into Customer  
values('C500', 'Ahmad', 'Amri', 'Malaysia', '0189090121', '900415');
```

```
insert into Customer  
values('C600', 'Sarah', 'Sue', 'Turkey', '9011234', '930403');
```

```
insert into Customer  
values('C700', 'Amirah', 'Izzati', 'Malaysia', '0142895676', '940707');
```

```
insert into Customer  
values('C800', 'Darwish', 'Ahnaf', 'Malaysia', '0156543210', '961010');
```

```
insert into Customer  
values('C900', 'Nur', 'Izzah', 'Indonesia', '6210089', '940506');
```

```
insert into Customer  
values('C110', 'Intan', 'Sofea', 'Dubai', '971050', '970118');
```

```
SQL> select * from customer;
```

CUSTOMER	CUST_FNAME	CUST_LNAME	COUNTRY	PHONENO	CUSTPASSPORT_IC
C100	Tisya	Amani	Malaysia	123456789	921106
C200	Fahim	Najmi	Malaysia	176565410	991203
C300	Johan	Ahmad	Malaysia	122566310	971006
C400	Aqil	Ali	Malaysia	165634100	980602
C500	Ahmad	Amri	Malaysia	189090121	900415
C600	Sarah	Sue	Turkey	9011234	930403
C700	Amirah	Izzati	Malaysia	142895676	940707
C800	Darwish	Ahnaf	Malaysia	156543210	961010
C900	Nur	Izzah	Indonesia	6210089	940506
C110	Intan	Sofea	Dubai	971050	970118

```
10 rows selected.
```

### Staff Data:

```
insert into Staff
```

```
values(5100, 'Amanda Eydora', '0149302784', TO_DATE('13-DEC-20'), 8, 1, 7.5,1550);
```

```
insert into Staff
```

```
values(5110, 'Fahrin Ahmad', '0127396381',TO_DATE('13-DEC-20'), 8,2, 15, 70000);
```

```
insert into Staff
```

```
values(5120, 'Uyaina Arshad', '0138740827', TO_DATE('13-DEC-20'), 8,3, 22.5, 70000);
```

```
insert into Staff
```

```
values(5130, 'Misha Omar', '0195472973', TO_DATE('13-DEC-20'), 8, 1, 7.5,1550);
```

```
insert into Staff
```

```
values(5140, 'Fatin Ameera', '0118936592', TO_DATE('13-DEC-20'), 8, 2, 15,2550);
```

```
insert into Staff
```

```
values(5150, 'Remy Ishak', '0107362857', TO_DATE('13-DEC-20'), 8,3, 22.5,3550);
```

```
insert into Staff
```

```
values(5160, 'Shaheizy Sam', '0173856728', TO_DATE('13-DEC-20'), 8,1, 7.5, 1550);
```

```
insert into Staff
```

```
values(5170, 'Hadri Zafran', '01375809547', TO_DATE('13-DEC-20'), 8,2, 15,2550);
```

```
insert into Staff
```

```
values(5180, 'John Ambrose', '0196568360', TO_DATE('13-DEC-20'), 8,3, 22.5,3550);
```

insert into Staff

values(5190, 'Kate Moss', '0148905682', TO\_DATE('13-DEC-20'), 8,1, 7.5,1550);

```
SQL> select * from staff;
```

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5100	Amanda Eydora	149302784	8	1
5110	Fahrin Ahmad	127396381	8	2
5120	Uyaina Arshad	138740827	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5130	Misha Omar	195472973	8	1
5140	Fatin Ameera	118936592	8	2
5150	Remy Ishak	107362857	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5160	Shaheizy Sam	173856728	8	1
5170	Hadri Zafran	1375809547	8	2
5180	John Ambrose	196568360	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5190	Kate Moss	148905682	8	1

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5100	Amanda Eydora	149302784	8	1
5110	Fahrin Ahmad	127396381	8	2
5120	Uyaina Arshad	138740827	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5130	Misha Omar	195472973	8	1
5140	Fatin Ameera	118936592	8	2
5150	Remy Ishak	107362857	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5160	Shaheizy Sam	173856728	8	1
5170	Hadri Zafran	1375809547	8	2
5180	John Ambrose	196568360	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5190	Kate Moss	148905682	8	1

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5100	Amanda Eydora	149302784	8	1
5110	Fahrin Ahmad	127396381	8	2
5120	Uyaina Arshad	138740827	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5130	Misha Omar	195472973	8	1
5140	Fatin Ameera	118936592	8	2
5150	Remy Ishak	107362857	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5160	Shaheizy Sam	173856728	8	1
5170	Hadri Zafran	1375809547	8	2
5180	John Ambrose	196568360	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5190	Kate Moss	148905682	8	1

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5100	Amanda Eydora	149302784	8	1
5110	Fahrin Ahmad	127396381	8	2
5120	Uyaina Arshad	138740827	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5130	Misha Omar	195472973	8	1
5140	Fatin Ameera	118936592	8	2
5150	Remy Ishak	107362857	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5160	Shaheizy Sam	173856728	8	1
5170	Hadri Zafran	1375809547	8	2
5180	John Ambrose	196568360	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5190	Kate Moss	148905682	8	1

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5100	Amanda Eydora	149302784	8	1
5110	Fahrin Ahmad	127396381	8	2
5120	Uyaina Arshad	138740827	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5130	Misha Omar	195472973	8	1
5140	Fatin Ameera	118936592	8	2
5150	Remy Ishak	107362857	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5160	Shaheizy Sam	173856728	8	1
5170	Hadri Zafran	1375809547	8	2
5180	John Ambrose	196568360	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5190	Kate Moss	148905682	8	1

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5100	Amanda Eydora	149302784	8	1
5110	Fahrin Ahmad	127396381	8	2
5120	Uyaina Arshad	138740827	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5130	Misha Omar	195472973	8	1
5140	Fatin Ameera	118936592	8	2
5150	Remy Ishak	107362857	8	3

STAFFID	STAFF_NAME	STAFF_PHONENO	WORK_HOUR	TOTAL_OTHR
5160	Shaheizy Sam	173856728	8	1
5170	Hadri Zafran	1375809547	8	2
5180	John Ambrose	196568360	8	3

10 rows selected.



### **Ticket Data:**

--2adult 1 kid

```
INSERT INTO TICKET
VALUES(10000, 'C100', TO_DATE('14-MAR-2019'), 'Adult', 40);
INSERT INTO TICKET
VALUES(10001, 'C100', TO_DATE('14-MAR-2019'), 'Adult', 40);
INSERT INTO TICKET
VALUES(10002, 'C100', TO_DATE('14-MAR-2019'), 'Kid', 20);
```

--4adult 2kid

```
INSERT INTO TICKET
VALUES(10003, 'C200', TO_DATE('4-JUN-2019'), 'Adult', 40);
INSERT INTO TICKET
VALUES(10004, 'C200', TO_DATE('4-JUN-2019'), 'Adult', 40);
INSERT INTO TICKET
VALUES(10005, 'C200', TO_DATE('4-JUN-2019'), 'Adult', 40);
INSERT INTO TICKET
VALUES(10006, 'C200', TO_DATE('4-JUN-2019'), 'Adult', 40);
INSERT INTO TICKET
VALUES(10007, 'C200', TO_DATE('4-JUN-2019'), 'Kid', 20);
INSERT INTO TICKET
VALUES(10008, 'C200', TO_DATE('4-JUN-2019'), 'Kid', 20);
```

--2adult

```
INSERT INTO TICKET
VALUES(10009, 'C300', TO_DATE('4-APR-2019'), 'Adult', 40);
INSERT INTO TICKET
VALUES(10010, 'C300', TO_DATE('4-APR-2019'), 'Adult', 40);
```

--4adult

```
INSERT INTO TICKET
VALUES(10011, 'C400', TO_DATE('14-MAY-2019'), 'Adult', 40);
INSERT INTO TICKET
VALUES(10012, 'C400', TO_DATE('14-MAY-2019'), 'Adult', 40);
INSERT INTO TICKET
VALUES(10013, 'C400', TO_DATE('14-MAY-2019'), 'Adult', 40);
INSERT INTO TICKET
VALUES(10014, 'C400', TO_DATE('14-MAY-2019'), 'Adult', 40);
```

--3 adult 2 kid

```
INSERT INTO TICKET
VALUES(10015, 'C500', TO_DATE('14-MAY-2019'), 'Adult', 40);
INSERT INTO TICKET
VALUES(10016, 'C500', TO_DATE('14-MAY-2019'), 'Adult', 40);
INSERT INTO TICKET
```

```

VALUES(10017, 'C500', TO_DATE('14-MAY-2019'), 'Adult', 40);
INSERT INTO TICKET
VALUES(10018, 'C500', TO_DATE('14-FEB-2019'), 'Kid', 20);
INSERT INTO TICKET
VALUES(10019, 'C500', TO_DATE('14-FEB-2019'), 'Kid', 20);

--1 adult
INSERT INTO TICKET
VALUES(10020, 'C600', TO_DATE('1-JUN-2019'), 'International', 60);

--2 adult
INSERT INTO TICKET
VALUES(10021, 'C700', TO_DATE('25-SEP-2019'), 'Adult', 40);
INSERT INTO TICKET
VALUES(10022, 'C700', TO_DATE('25-SEP-2019'), 'Adult', 40);

--1 adult
INSERT INTO TICKET
VALUES(10023, 'C800', TO_DATE('1-NOV-2019'), 'Adult', 40);

--10 international
INSERT INTO TICKET
VALUES(10024, 'C900', TO_DATE('14-MAR-2019'), 'International', 60);
INSERT INTO TICKET
VALUES(10025, 'C900', TO_DATE('14-MAR-2019'), 'International', 60);
INSERT INTO TICKET
VALUES(10026, 'C900', TO_DATE('14-MAR-2019'), 'International', 60);
INSERT INTO TICKET
VALUES(10027, 'C900', TO_DATE('14-MAR-2019'), 'International', 60);
INSERT INTO TICKET
VALUES(10028, 'C900', TO_DATE('14-MAR-2019'), 'International', 60);
INSERT INTO TICKET
VALUES(10029, 'C900', TO_DATE('14-MAR-2019'), 'International', 60);
INSERT INTO TICKET
VALUES(10030, 'C900', TO_DATE('14-MAR-2019'), 'International', 60);
INSERT INTO TICKET
VALUES(10031, 'C900', TO_DATE('14-MAR-2019'), 'International', 60);

INSERT INTO TICKET
VALUES(10032, 'C900', TO_DATE('14-MAR-2019'), 'International', 60);
INSERT INTO TICKET
VALUES(10033, 'C900', TO_DATE('14-MAR-2019'), 'International', 60);

```

--1 international

INSERT INTO TICKET

VALUES(10034, 'C110', TO\_DATE('1-NOV-2019'), 'International', 60);

```
SQL> select * from ticket;
```

TICKETID	CUSTOMER	TICKET_DA	TICKET_TYPE	TICKET_PRICE
10000	C100	14-MAR-19	Adult	40
10001	C100	14-MAR-19	Adult	40
10002	C100	14-MAR-19	Kid	20
10003	C200	04-JUN-19	Adult	40
10004	C200	04-JUN-19	Adult	40
10005	C200	04-JUN-19	Adult	40
10006	C200	04-JUN-19	Adult	40
10007	C200	04-JUN-19	Kid	20
10008	C200	04-JUN-19	Kid	20
10009	C300	04-APR-19	Adult	40
10010	C300	04-APR-19	Adult	40

TICKETID	CUSTOMER	TICKET_DA	TICKET_TYPE	TICKET_PRICE
10011	C400	14-MAY-19	Adult	40
10012	C400	14-MAY-19	Adult	40
10013	C400	14-MAY-19	Adult	40
10014	C400	14-MAY-19	Adult	40
10015	C500	14-MAY-19	Adult	40
10016	C500	14-MAY-19	Adult	40
10017	C500	14-MAY-19	Adult	40
10018	C500	14-FEB-19	Kid	20
10019	C500	14-FEB-19	Kid	20
10020	C600	01-JUN-19	International	60
10021	C700	25-SEP-19	Adult	40

TICKETID	CUSTOMER	TICKET_DA	TICKET_TYPE	TICKET_PRICE
10022	C700	25-SEP-19	Adult	40
10023	C800	01-NOV-19	Adult	40
10024	C900	14-MAR-19	International	60
10025	C900	14-MAR-19	International	60
10026	C900	14-MAR-19	International	60
10027	C900	14-MAR-19	International	60
10028	C900	14-MAR-19	International	60
10029	C900	14-MAR-19	International	60
10030	C900	14-MAR-19	International	60
10031	C900	14-MAR-19	International	60
10032	C900	14-MAR-19	International	60
TICKETID	CUSTOMER	TICKET_DA	TICKET_TYPE	TICKET_PRICE
10033	C900	14-MAR-19	International	60
10034	C110	01-NOV-19	International	60

35 rows selected.

### Promotion Data:

insert into Promotion

values('PR1294', 'Staff Privilege', TO\_DATE('13-DEC-20'));

insert into Promotion

values('PR1456', 'Happy New Year Promotion', TO\_DATE('01-JAN-20'));

insert into Promotion

values('PR2541', 'Chinese New Year Promotion', TO\_DATE('25-JAN-20'));

insert into Promotion

values('PR2741', 'Valentine Day Promotion', TO\_DATE('14-FEB-20'));

insert into Promotion

values('PR5784', 'Labour Day Promotion', TO\_DATE('01-MAY-19'));

insert into Promotion

values('PR5398', 'Eid Mubarak Promotion', TO\_DATE('24-MAY-19'));

insert into Promotion

values('PR5284', 'Eid Al Adha Promotion', TO\_DATE('31-JUL-19'));

insert into Promotion

values('PR8497', 'Merdeka Day Promotion', TO\_DATE('31-AUG-19'));

insert into Promotion

values('PR9143', 'Awal Muharam Promotion', TO\_DATE('01-SEP-19'));

```
insert into Promotion
values('PR9428', 'Malaysia Day Promotion', TO_DATE('16-SEP-19'));
```

```
insert into Promotion
values('PR1168', 'Deepavali Promotion', TO_DATE('14-NOV-19'));
```

```
insert into Promotion
values('PR1174', 'Childrens Day Promotion', TO_DATE('20-NOV-19'));
```

```
insert into Promotion
values('PR1194', 'Christmas Day Promotion', TO_DATE('25-DEC-19'));
```

```
SQL> select * from Promotion;
```

PROMOID	PROMO_TYPE	PROMO_DAT
PR1294	Staff Privilege	13-DEC-20
PR1456	Happy New Year Promotion	01-JAN-20
PR2541	Chinese New Year Promotion	25-JAN-20
PR2741	Valentine Day Promotion	14-FEB-20
PR5784	Labour Day Promotion	01-MAY-19
PR5398	Eid Mubarak Promotion	24-MAY-19
PR5284	Eid Al Adha Promotion	31-JUL-19
PR8497	Merdeka Day Promotion	31-AUG-19
PR9143	Awal Muharam Promotion	01-SEP-19
PR9428	Malaysia Day Promotion	16-SEP-19
PR1168	Deepavali Promotion	14-NOV-19
PR1174	Childrens Day Promotion	20-NOV-19
PR1194	Christmas Day Promotion	25-DEC-19

```
13 rows selected.
```

**Payment Data:**

Insert into Payment

values('P1101', 'C100', 3, 0.1, 100); --2Adults, 1 Kid

Insert into Payment

values('P1102', 'C200', 6, 0.1, 200); --4Adults, 2 Kids

Insert into Payment

values('P1103', 'C300', 2, 0.1, 80); --2Adults

Insert into Payment

values('P1104', 'C400', 4, 0.1, 160); --4Adults

Insert into Payment

values('P1105', 'C500', 5, 0.1, 160); --3Adults, 2 Kids

Insert into Payment

values('P1106', 'C600', 1, 0.1, 60); --1 international

Insert into Payment

values('P1107', 'C700', 2, 0.1, 80); --2Adults

Insert into Payment

values('P1108', 'C800', 1, 0.1, 40); --1Adult

Insert into Payment

values('P1109', 'C900', 10, 0.1, 600); --10 International

Insert into Payment

values('P1110', 'C110', 15, 0.1, 60); --1 international

```
SQL> select * from payment;
```

PAYMENTI	CUSTOMER	TICKET_QNT	TAX_AMT	GRAND_TOTAL
P1101	C100	3	.1	100
P1102	C200	6	.1	200
P1103	C300	2	.1	80
P1104	C400	4	.1	160
P1105	C500	5	.1	160
P1106	C600	1	.1	60
P1107	C700	2	.1	80
P1108	C800	1	.1	40
P1109	C900	10	.1	600
P1110	C110	15	.1	60

```
10 rows selected.
```

### 3.0 SQL Statements (Add, Delete, Modify)

#### Scenario 1:

- One customer data was added into the Customer database.

```
insert into Customer
values('C120', 'Jase', 'Falique', 'Malaysia', '0123434567', '970403');
```

```
SQL> insert into Customer
  2  values('C120', 'Jason', 'Falique', 'Malaysia', '0123434567', '970403');

1 row created.
```

- One customer data was deleted from the Customer database.

```
delete from customer
where customerID = 'C120';
```

```
SQL> delete from customer
  2  where customerID = 'C120';

1 row deleted.
```

- Update the customer phone number where the CustomerID = 'C500' from Customer database.

```
update Customer
set phoneno = '0122633555'
where customerid = 'C500';
```

```
select * from Customer
where customerid = 'C500';
```

```
SQL> update Customer
  2  set phoneno = '0122633555'
  3  where customerid = 'C500';

1 row updated.

SQL> select * from Customer
  2  where customerid = 'C500';
```

CUSTOMER	CUST_FNAME	CUST_LNAME	COUNTRY	PHONENO	CUSTPASSPORT_IC
C500	Ahmad	Amri	Malaysia	122633555	900415



**Scenario 2:**

Admin list out the quantity and the grand total of tickets that have been bought by the customer named 'Tisya'.

```
select c.customerid, t.ticket_qnt, t.grand_total
from customer c, ticket t
where c.customerid = t.customerid and c.cust_fname = 'Tisya';
```

```
SQL> select c.customerid, p.ticket_qnt, p.grand_total
  2  from customer c, payment p
  3  where c.customerid = p.customerid AND c.cust_fname = 'Tisya';
```

CUSTOMER	TICKET_QNT	GRAND_TOTAL
C100	3	100

**Scenario 3:**

Admin update the ticket quantity and different ticket price were generated for customer which the customerID = 'C100'.

```
update payment
set ticket_qnt = 4, grand_total = 140
where customerid = 'C100';
```

```
SQL> update payment
  2  set ticket_qnt = 4, grand_total = 140
  3  where customerID = 'C100';

1 row updated.

SQL> select customerid, ticket_qnt, grand_total
  2  from payment
  3  where customerID = 'C100';
```

CUSTOMER	TICKET_QNT	GRAND_TOTAL
C100	4	140

**Scenario 4:**

Admin list all the ticket type and ticket price that have been bought by the customer.

```
select c.customerid, c.cust_fname, c.cust_lname, t.ticket_type, t.ticket_price
from Customer c, Ticket t
where c.customerid = t.customerid;
```

```
SQL> select c.customerid, c.cust_fname, c.cust_lname, t.ticket_type, t.ticket_price
2      from Customer c, Ticket t
3      where c.customerid = t.customerid;
```

CUSTOMER	CUST_FNAME	CUST_LNAME	TICKET_TYPE	TICKET_PRICE
C100	Tisya	Amani	Adult	40
C100	Tisya	Amani	Adult	40
C100	Tisya	Amani	Kid	20
C200	Fahim	Najmi	Adult	40
C200	Fahim	Najmi	Adult	40
C200	Fahim	Najmi	Adult	40
C200	Fahim	Najmi	Adult	40
C200	Fahim	Najmi	Kid	20
C200	Fahim	Najmi	Kid	20
C300	Johan	Ahmad	Adult	40
C300	Johan	Ahmad	Adult	40

CUSTOMER	CUST_FNAME	CUST_LNAME	TICKET_TYPE	TICKET_PRICE
C400	Aqil	Ali	Adult	40
C400	Aqil	Ali	Adult	40
C400	Aqil	Ali	Adult	40
C400	Aqil	Ali	Adult	40
C500	Ahmad	Amri	Adult	40
C500	Ahmad	Amri	Adult	40
C500	Ahmad	Amri	Adult	40
C500	Ahmad	Amri	Kid	20
C500	Ahmad	Amri	Kid	20
C600	Sarah	Sue	International	60
C700	Amirah	Izzati	Adult	40

CUSTOMER	CUST_FNAME	CUST_LNAME	TICKET_TYPE	TICKET_PRICE
C700	Amirah	Izzati	Adult	40
C800	Darwish	Ahnaf	Adult	40
C900	Nur	Izzah	International	60
C900	Nur	Izzah	International	60
C900	Nur	Izzah	International	60
C900	Nur	Izzah	International	60
C900	Nur	Izzah	International	60
C900	Nur	Izzah	International	60
C900	Nur	Izzah	International	60
C900	Nur	Izzah	International	60
C900	Nur	Izzah	International	60

CUSTOMER	CUST_FNAME	CUST_LNAME	TICKET_TYPE	TICKET_PRICE
C900	Nur	Izzah	International	60
C110	Intan	Sofea	International	60

35 rows selected.

### Scenario 5:

All staff are assigned to Staff Privileged Promotion which staff need to join with the Promotion table.

```
select s.staffid, s.staff_name, p.promo_type
from staff s, promotion p
where s.promo_date = p.promo_date;
```

```
SQL> select s.staffid, s.staff_name, p.promo_type
  2  from staff s, promotion p
  3  where s.promo_date = p.promo_date;
```

STAFFID	STAFF_NAME	PROMO_TYPE
5190	Kate Moss	Staff Privilege
5180	John Ambrose	Staff Privilege
5170	Hadri Zafran	Staff Privilege
5160	Shaheizy Sam	Staff Privilege
5150	Remy Ishak	Staff Privilege
5140	Fatin Ameera	Staff Privilege
5130	Misha Omar	Staff Privilege
5120	Uyaina Arshad	Staff Privilege
5110	Fahrin Ahmad	Staff Privilege
5100	Amanda Eydora	Staff Privilege

10 rows selected.

### Scenario 6:

List all promotion date and type that available on January.

```
SQL> select promo_type, promo_date
  2  from promotion
  3  where promo_date like '%JAN%';
```

PROMO_TYPE	PROMO_DAT
Happy New Year Promotion	01-JAN-20
Chinese New Year Promotion	25-JAN-20

## 4.0 Create two procedures:

### Scenario 1:

Calculate New salary for Staff when they work overtime.

```
procedure1 - Notepad
File Edit Format View Help
create or replace procedure new_sal(v_sid IN NUMBER) is

    cursor cur_sal is
    select staff_name, staff_phoneno, work_hour, total_othr, allowance_perhr, sal
    from staff
    where staffid = v_sid;

    v_sname staff.staff_name%type;
    v_phone staff.staff_phoneno%type;
    v_workhour staff.work_hour%type;
    v_totalothr staff.total_othr%type;
    v_allowance staff.allowance_perhr%type;
    v_sal staff.sal%type;
    OT_payment number(5,2);
    newsal number(8,2);

begin
    open cur_sal;
    loop
    fetch cur_sal into v_sname, v_phone, v_workhour, v_totalothr, v_allowance, v_sal;
    OT_payment := v_totalothr * v_allowance;
    newsal := v_sal + OT_payment;
    exit when cur_sal%notfound;

    dbms_output.put_line ('The staff is: '||v_sname);
    dbms_output.put_line ('The staff contact number: '||v_phone);
    dbms_output.put_line ('The staff regular working hour: '||v_workhour);
    dbms_output.put_line ('The staff total OT per hour: '||v_totalothr);
    dbms_output.put_line ('The staff salary per month: RM '||v_sal);
    dbms_output.put_line ('The staff OT payment: RM '||OT_payment);
    dbms_output.put_line ('The new salary for staff: RM '||newsal);

    end loop;
    close cur_sal;

end;
/
```

```
SQL> @"I:\Namirah PART\procedure1.txt"

Procedure created.

SQL> execute new_sal(5110);
The staff is: Fahrin Ahmad
The staff contact number: 127396381
The staff regular working hour: 8
The staff total OT per hour: 2
The staff salary per month: RM 70000
The staff OT payment: RM 15
The new salary for staff: RM 70015

PL/SQL procedure successfully completed.

SQL> execute new_sal(5190);
The staff is: Kate Moss
The staff contact number: 148905682
The staff regular working hour: 8
The staff total OT per hour: 1
The staff salary per month: RM 1550
The staff OT payment: RM 7.5
The new salary for staff: RM 1557.5

PL/SQL procedure successfully completed.
```

### Scenario 2:

In business perspective, the management or marketing team will have to know the impact of each date. This is because, they need to know which date or promotion will bring more customers to the zoo in order to know whether their promotion is a success or not. Thus, a procedure is created to display the category and total of customers, based on their age categories(adult, kid, international) for a given date.

```
create or replace procedure show_cust
(tDate ticket.ticket_date%type)
as
tType ticket.ticket_type%type;
ex_chg EXCEPTION;
i number(2) :=1;
ad number(2) :=0;
kd number(2) :=0;
intr number(2) :=0;
cursor countcust
is
select ticket_type
from ticket
where ticket_date= tDate;

begin
open countcust;

loop

fetch countcust into tType;
IF countcust%NOTFOUND THEN
RAISE ex_chg;
END IF;

exit when countcust%notfound;
dbms_output.put_line(i ||' : '||tType);
i:= i+1;

case when tType = 'Adult' then ad:=ad+1;
when tType = 'Kid' then kd:=kd+1;
when tType = 'International' then intr:=intr+1;
end case;
end loop;
```

```

EXCEPTION
WHEN ex_chg THEN
DBMS_OUTPUT.PUT_LINE ('Oops! No more customer for that date');
dbms_output.put_line('Total kid(s) : '||kd);
dbms_output.put_line('Total adult(s) : '||ad);
dbms_output.put_line('Total international(s) : '||intr);

close countcust;

end;
/

```

Procedure created.

The procedure is called and the date to be observed is entered.

```

ACCEPT p_id PROMPT 'Enter the date to observe: '

declare
dDate Date := '&date';

begin
DBMS_OUTPUT.put_line('The customer(s) that came for date '||
dDate||' are as follows: ');

show_cust(dDate);

end;
/

```

During 25-SEP-19, only 2 adults came to the zoo. It be inferred that this may happen because 25-SEP-19 is on Wednesday where mostly people are working during weekdays

```

Enter value for date: 25-SEP-19
old 2: dDate Date := '&date';
new 2: dDate Date := '25-SEP-19';
The customer(s) that came for date 25-SEP-19 are as follows:
1 : Adult
2 : Adult
Oops! No more customer for that date
Total kid(s) : 0
Total adult(s) : 2
Total international(s) : 0

PL/SQL procedure successfully completed.

```

During 1-FEB-19, no one came to the zoo. It be inferred that this may happen because no special event happen on the day or it is on Friday where mostly people are working during weekdays.

```
Enter value for date: 1-FEB-19
old 2: dDate Date := '&date';
new 2: dDate Date := '1-FEB-19';
The customer(s) that came for date 01-FEB-19 are as follows:
Oopps! No more customer for that date
Total kid(s) : 0
Total adult(s) : 0
Total international(s) : 0

PL/SQL procedure successfully completed.
```

However, on 4-JUN-19, more people came to the zoo eventhou it is on Tuesday. This is because 4-JUN-19 is a school holiday and parents tend to bring their kid to a trip to the zoo.

```
Enter value for date: 4-JUN-19
old 2: dDate Date := '&date';
new 2: dDate Date := '4-JUN-19';
The customer(s) that came for date 04-JUN-19 are as follows:
1 : Adult
2 : Adult
3 : Adult
4 : Adult
5 : Kid
6 : Kid
Oopps! No more customer for that date
Total kid(s) : 2
Total adult(s) : 4
Total international(s) : 0

PL/SQL procedure successfully completed.
```

## 5.0 Create two functions:

### Scenario 1:

New tax has been charged for the Customer as the Zoo trip are having panda, penguin and unique camelion visit. Each new tax been charged once only for the customer grand amount.

```
tripPrice - Notepad
File Edit Format View Help

create or replace function trip_price
(v_grandAmt in payment.grand_total%type, v_tiketqnt in payment.ticket_qnt%type)
return varchar2
is
v_incrtax number(5,2);
begin
    if v_grandAmt <= 140 AND v_tiketqnt <= 4 then
        v_incrtax := 0.15;
    elsif v_grandAmt <= 250 AND v_tiketqnt >= 4 then
        v_incrtax := 0.2;
    elsif v_grandAmt <= 400 AND v_tiketqnt >= 8 then
        v_incrtax := 0.25;
    else
        v_incrtax := 0.4;
    end if;
return v_incrtax;
end trip_price;
/

SQL> @C:\Users\User\Documents\database\tripPrice.txt

Function created.
```

```
calltrip_price - Notepad
File Edit Format View Help

prompt -----welcome to Zoo Trip-----;
prompt .....*Attention*.....;
prompt 1. There's new tax charge as this zoo are having panda, penguin and unique camelion visit.;
prompt 2. Just used one customerID for one trip.;
prompt Compulsory Note: Please insert the single quotation infront and back customerID! - Eg: 'C1XX'
prompt -----Thank you and enjoy your trip-----;

Accept v_customerID prompt 'Enter the Customer ID (one ID for a group trip): '
Accept v_tiketqnt prompt 'Enter the ticket quantity: '

declare
    v_custfname customer.cust_fname%type;
    v_custlname customer.cust_lname%type;
    v_country customer.country%type;
    v_tiketqnt number(3);
    v_incrtax number(5,2);
    v_grandAmt number(5,2);
    v_tiketAmt number(5,2);

begin
    select c.cust_fname, c.cust_lname, c.country, p.ticket_qnt, p.grand_total
    into v_custfname, v_custlname, v_country, v_tiketqnt, v_grandAmt
    from customer c INNER JOIN payment p
    ON c.customerID = p.customerID
    where p.customerID = &v_customerID;

    v_incrtax := trip_price(v_grandAmt,v_tiketqnt);
    v_tiketAmt := v_grandAmt + (v_grandAmt*v_incrtax);

    dbms_output.put_line('-----NEW GRAND AMOUNT OF TICKET-----');
    dbms_output.put_line('The customer name: '||v_custfname||' '||v_custlname);
    dbms_output.put_line('The customer country: '||v_country);
    dbms_output.put_line('Total ticket quantity: '||v_tiketqnt);
    dbms_output.put_line('Ticket increment tax: '||v_incrtax*100||' percent from total price.');
```



```

SQL> @C:\Users\User\Documents\database\calltrip_price.txt
-----Welcome to Zoo Trip-----
.....*Attention*.....
1. There's additional tax implemented on panda, penguin and unique chameleons visit.
2. Just use one customerID for one trip.
Compulsory Note: Please insert the single quotation infront and back customerID! - Eg: 'C1XX'
-----Thank you and enjoy your trip-----
Enter the Customer ID (one ID for a group trip): 'C200'
Enter the ticket quantity: 6
old 16:      where p.customerID = &v_customerID;
new 16:      where p.customerID = 'C200';
-----NEW GRAND AMOUNT OF TICKET-----
The customer name: Fahim Najmi
The customer country: Malaysia
Total ticket quantity: 6
Ticket increment tax: 20 percent from total price.
Ticket grand amount: RM 240
-----

PL/SQL procedure successfully completed.

SQL> @C:\Users\User\Documents\database\calltrip_price.txt
-----Welcome to Zoo Trip-----
.....*Attention*.....
1. There's additional tax implemented on panda, penguin and unique chameleons visit.
2. Just use one customerID for one trip.
Compulsory Note: Please insert the single quotation infront and back customerID! - Eg: 'C1XX'
-----Thank you and enjoy your trip-----
Enter the Customer ID (one ID for a group trip): 'C300'
Enter the ticket quantity: 3
old 16:      where p.customerID = &v_customerID;
new 16:      where p.customerID = 'C300';
-----NEW GRAND AMOUNT OF TICKET-----
The customer name: Johan Ahmad
The customer country: Malaysia
Total ticket quantity: 2
Ticket increment tax: 15 percent from total price.
Ticket grand amount: RM 92
-----

PL/SQL procedure successfully completed.

```

## Scenario 2:

Customer may check promotion prices prior the promotion date as mentioned in the Zoo Adult Promotion Price but the purchase slot are not offered yet for all customers.

```
promoPrice - Notepad
File Edit Format View Help
create or replace function promo_price
(v_promodate in promotion.promo_date%type, v_promoid in promotion.promoid%type)
return varchar2
is
v_promoprice number(5,2);
begin

    if v_promodate = '25-DEC-19' AND v_promoid = 'PR1194' then
        v_promoprice := 26;
    elsif v_promodate = '01-JAN-20' AND v_promoid = 'PR1456' then
        v_promoprice := 20;
    elsif v_promodate = '25-JAN-20' AND v_promoid = 'PR2541' then
        v_promoprice := 18;
    elsif v_promodate = '14-FEB-20' AND v_promoid = 'PR2741' then
        v_promoprice := 14;
    else
        v_promoprice := 30;
    end if;

return v_promoprice;
end promo_price;
/
```

```
SQL> @C:\Users\User\Documents\database\promoPrice.txt
```

```
Function created.
```

```
callpromo - Notepad
File Edit Format View Help
prompt -----Zoo Adult Promotion Price-----;
prompt 1. The original ticket price for adult - RM40.00 per head.;
prompt 2. The promotion are available for selected date only: '25-DEC-19', '01-JAN-20', '25-JAN-20', '14-FEB-20';
prompt 3. Code for Christmas Day(PR1194), Happy New Year(PR2541), Chinese New Year(PR1456), Valentine Day(PR2741)
prompt 4. This system are for checking the promotion price and the purchase slot are not available yet.;
prompt Compulsory Note: Please insert single quotation for PromotionID and Date as well! - Eg: 'PR1XX'
prompt -----Enjoy your promotion month-----;
Accept v_promodate prompt 'Enter the promotion date: '
Accept v_promoid prompt 'Enter the promotion ID: '

declare

    v_promoid promotion.promoid%TYPE;
    v_promotype promotion.promo_type%type;
    v_promodate promotion.promo_date%type;
    v_promoprice number(3);

    cursor cur_promo is
    select promoid, promo_type, promo_date
    into v_promoid, v_promotype, v_promodate
    from promotion
    where promo_date = &v_promodate;

begin

    open cur_promo;
    loop
    fetch cur_promo into v_promoid, v_promotype, v_promodate;
    v_promoprice := promo_price(v_promodate,v_promoid);
    exit when cur_promo%notfound;
    dbms_output.put_line('-----ADULT PROMOTION PRICE-----');
    dbms_output.put_line('Customer promotion: '||v_promotype);
    dbms_output.put_line('Customer promotion date: '||v_promodate);
    dbms_output.put_line('Adult ticket price for the selected promotion(per head): RM '||v_promoprice);
    dbms_output.put_line('-----');
    end loop;
    close cur_promo;

end;
/
```

```

SQL> @C:\Users\User\Documents\database\callpromo.txt
-----Zoo Adult Promotion Price-----
1. The original ticket price for adult - RM40.00 per head.
2. The promotion are available for selected date only: '25-DEC-19', '01-JAN-20', '25-JAN-20', '14-FEB-20'
3. Code for Christmast Day(PR1194), Happy New Year(PR2541), Chinese New Year(PR1456), Valentine Day(PR2741)
4. This system are for checking the promotion price and the purchase slot are not available yet.
Compulsory Note: Please insert single quotation for PromotionID and Date as well! - Eg:'PR1XX'
-----Enjoy your promotion month-----
Enter the promotion date: '14-FEB-20'
Enter the promotion ID: 'PR2741'
old 12:      where promo_date = &v_promodate;
new 12:      where promo_date = '14-FEB-20';
-----ADULT PROMOTION PRICE-----
Customer promotion: Valentine Day Promotion
Customer promotion date: 14-FEB-20
Adult ticket price for the selected promotion(per head): RM 14
-----

PL/SQL procedure successfully completed.

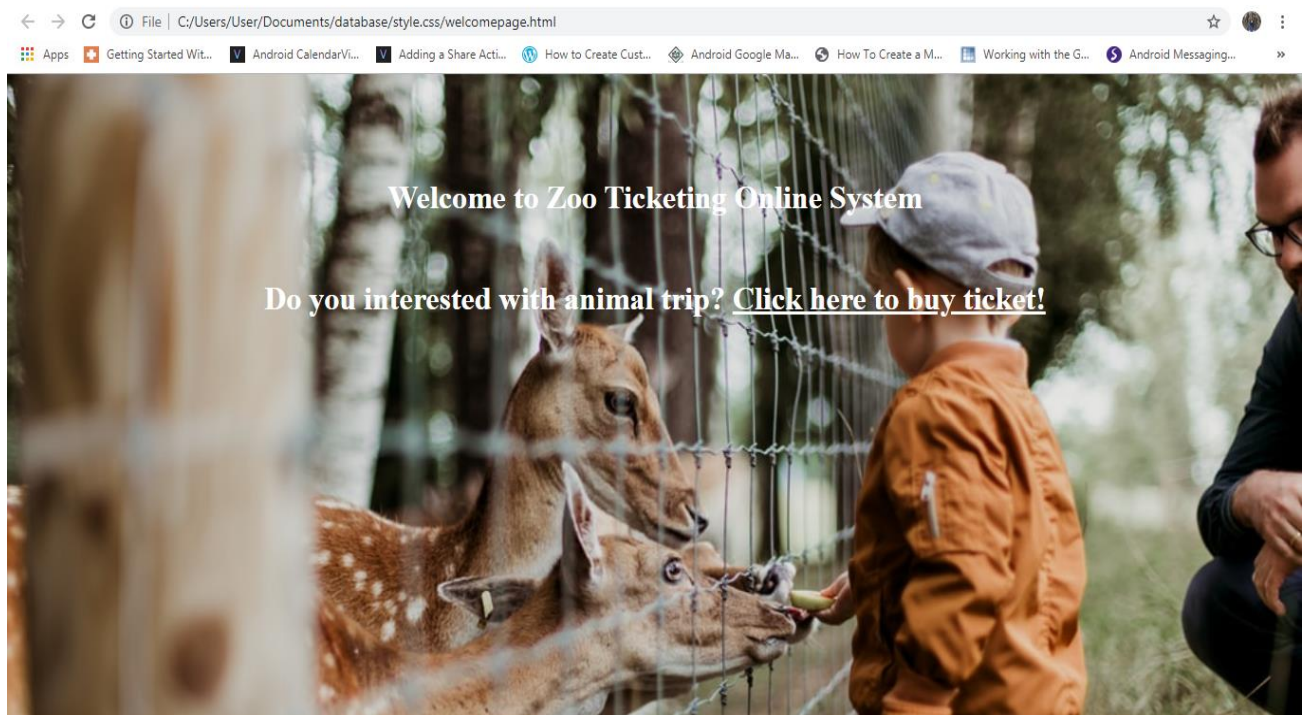
```

**Member Contribution for all tasks:**

<b>Member's Name &amp; Matric Number</b>	<b>Task</b>
Aishah Nabilah 1625084	<ul style="list-style-type: none"><li>- Project significant</li><li>- Create table ticket</li><li>- Insert data ticket</li><li>- Create data dictionary for ER Diagram ticket</li><li>- Create Zoo Ticketing System ERD</li><li>- Create procedures for Scenario 2</li></ul>
Nur Afifah 1810016	<ul style="list-style-type: none"><li>- Insert data promotion</li><li>- Create data dictionary for ER Diagram promotion</li><li>- SQL Statement (Add, Delete, Modify) for Scenario 6</li><li>- Create function for Scenario 2</li><li>- Conclusion</li><li>- References</li></ul>
Nurul Ain Shahirah 1818540	<ul style="list-style-type: none"><li>- Create table customer and promotion</li><li>- Insert data customer</li><li>- Create data dictionary for ER Diagram customer table</li><li>- SQL Statement (Add, Delete, Modify) for Scenario 1, Scenario 2, Scenario 3 and Scenario 4</li><li>- Create function for Scenario 1</li></ul>
Siti Noor Namirah Hasinah 1816818	<ul style="list-style-type: none"><li>- Background Project</li><li>- Create table payment and staff</li><li>- Insert data payment and staff</li><li>- Create data dictionary for ER Diagram payment and staff table</li><li>- SQL Statement (Add, Delete, Modify) for Scenario 5</li><li>- Create procedures for Scenario 1</li></ul>

## 6.0 Web Application for Zoo Ticketing System

### WelcomePage.html



### CustomerDetails.html

A screenshot of a web browser displaying the 'CustomerDetails.html' file. The browser's address bar shows the file path 'C:/Users/User/Documents/database/style.css/customerdetails.html'. The page has a light grey background with a subtle floral pattern. The title 'CUSTOMER DETAILS' is centered in a bold, black, serif font. Below the title is a form with the following fields: 'CustomerID:', 'First Name:', 'Last name:', 'Country:', 'Phone Number:', and 'Passport/IC Number:'. The 'Passport/IC Number:' field has two radio button options: 'Passport/Foreigner' (selected) and 'IC Number/Malaysian'. At the bottom of the form are two buttons: 'Submit' and 'Go Back'. The browser's taskbar at the bottom shows the same set of applications as the previous screenshot.

## ChooseTicket.html

← → ↻ ⓘ File | C:/Users/User/Documents/database/style.css/chooseticket.html ☆ 🌐 ⋮

📱 Apps 📌 Getting Started Wit... 📅 Android CalendarVi... 📄 Adding a Share Acti... 🌐 How to Create Cust... 📱 Android Google Ma... 📌 How To Create a M... 📄 Working with the G... 📱 Android Messaging... »

### TICKET TYPE

Ticket Type:  
☒ Adult  
☐ Kid/Senior Citizen  
☐ International

Ticket Quantity:

Date:

Promo Code:

## PayTicket.html

← → ↻ ⓘ File | C:/Users/User/Documents/database/style.css/payticket.html ☆ 🌐 ⋮

📱 Apps 📌 Getting Started Wit... 📅 Android CalendarVi... 📄 Adding a Share Acti... 🌐 How to Create Cust... 📱 Android Google Ma... 📌 How To Create a M... 📄 Working with the G... 📱 Android Messaging... »

### PAYMENT DETAILS

PaymentID:

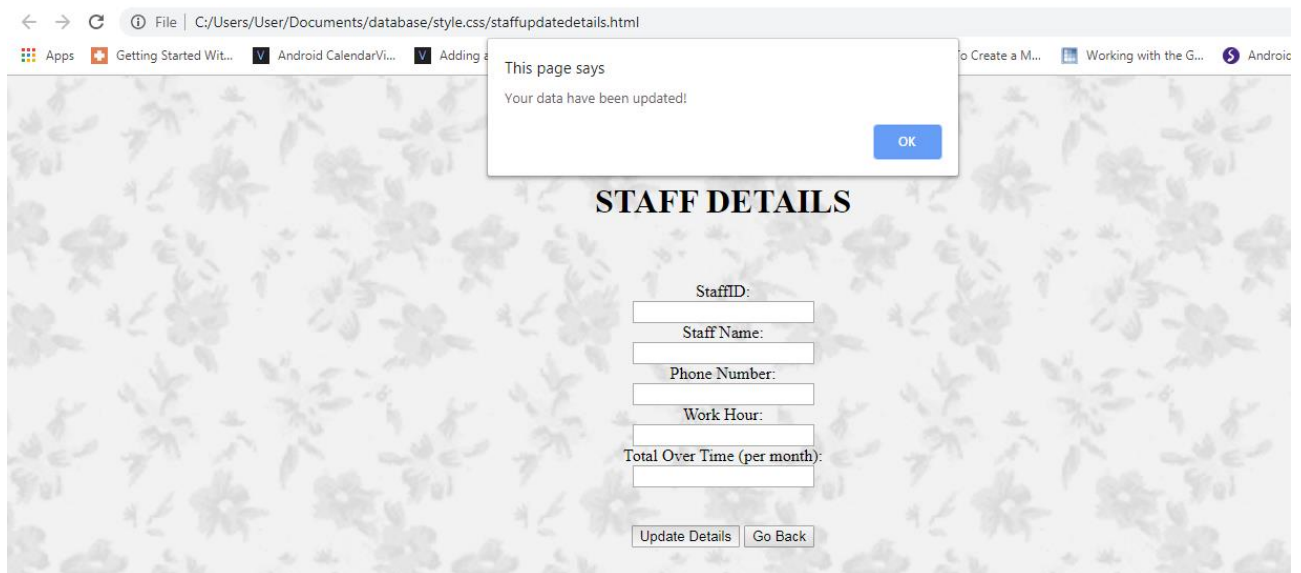
CustomerID:

Ticket Quantity:

**The tax for each grand amount are 10%, thank you!**



## StaffUpdateDetails.html



## 7.0 Conclusion

As a conclusion, completing this Group Project Assignment we have learned about how we should contribute with group members and teamwork skills while doing this assignment. We also learned how to manage our time wisely and how to divide the tasks fairly among group members. Basically, our Group Project Assignment is about Zoo Ticketing System which we allow users to buy zoo online ticket by using our system. Furthermore, we created this database is to ease the process of storing, updating, add, calculating salary and promotions for the staff and customers that works and comes to the zoo.

## 8.0 References

Buy Ticket Online - Get Zoo Negara Ticket. (n.d.).  
Retrieved from <https://ztpzoonegara.com/>.

Zoo Taiping Online Entrance Ticket Sale. (n.d.).  
Retrieved from <https://ticket.zootaiping.gov.my/selection2.aspx>