**FACULTY OF COMPUTING & INFORMATION TECHNOLOYGY**

**DEPARTMENT OF SOFTWARE ENGINEERING**

****

**Database Management System**

**TOPIC**

TRAVEL AND RESERVATION SYSTEM

**PRESENTED\_TO**

Sir Asif Sohail

**PRESENTED\_BY**

* Azka Mushtaq (BSEF21MO38)
* Aqsa Tariq(BSEF21M513)
* Asra Mushtaq (BSEF21MO41)

**TRAVEL AND RESERVATION SYSTEM**

1. **Introduction to the working of the system**

Travel Reservation System is a sophisticated backend software solution designed to streamline and automate various processes in the travel industry. It includes a robust user management system that stores comprehensive user profiles that include personal information (User Id, First Name, Last Name, Date of Birth, Account Creation Date, and Last Login Date), contact information (Email, Phone Number, and Address). The system includes secure Access Control (Username, Password) features, ensuring that only authorized users, whose identities have been verified through authentication, can make reservations or access specific functions. This adds an extra layer of protection, controlling and limiting access to sensitive actions within the system.

Reservation management is a core feature that captures detailed information about reservations, such as travel dates (reservation Id, departure date, Return Date), destinations (destination id, destination name, country, city, region, description, popular attractions, language, currency). The database must securely store payment information (credit card number, expiration date, cvv, billing address) to facilitate safe and seamless transactions.

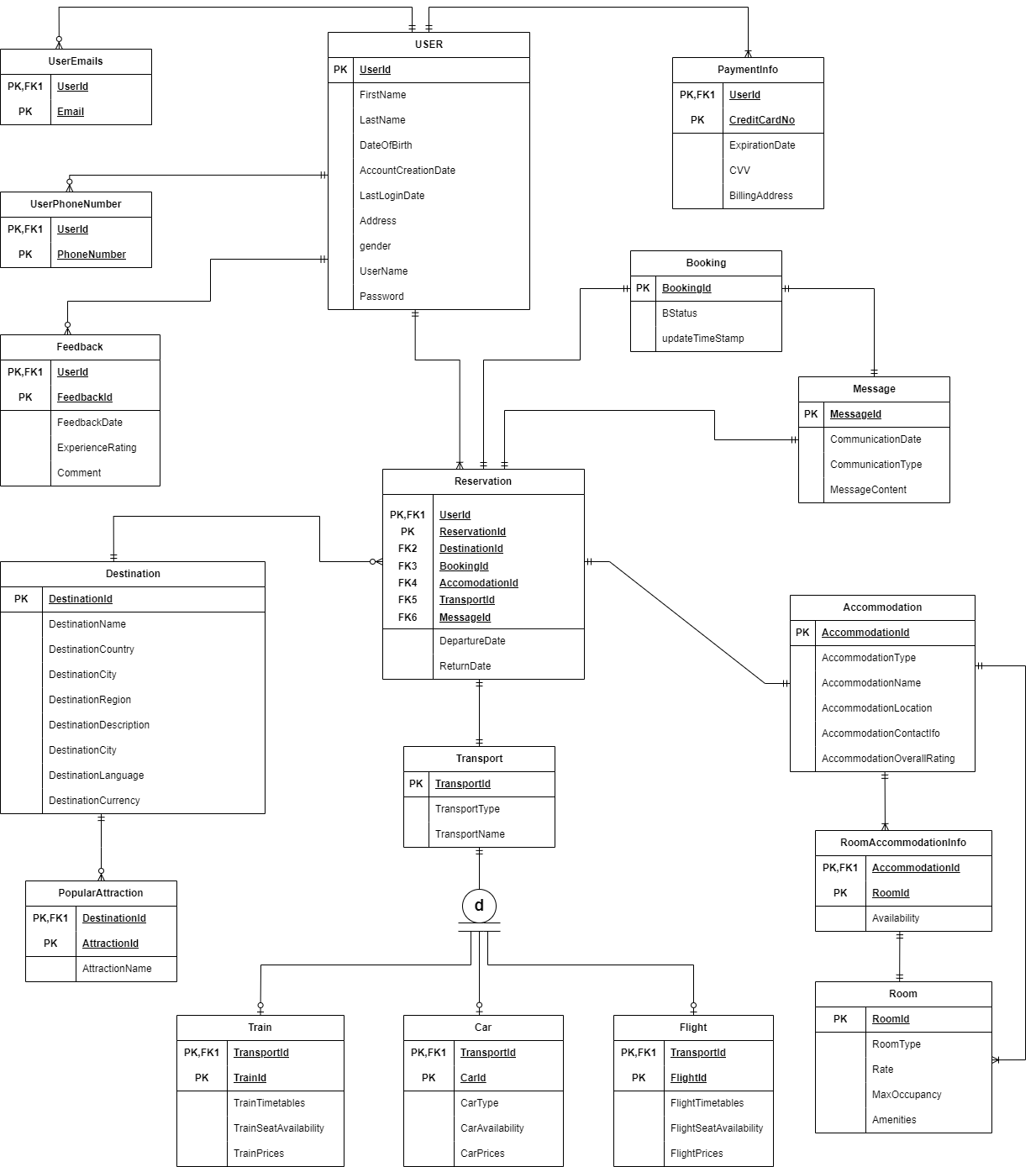
Booking status tracking is essential to keep users informed of the progress of their bookings (booking id, b\_status (e.g., pending, confirmed, cancelled), update\_timestamp). Inventory management is another key component that covers a database of available accommodations (accommodation\_id, accommodation\_type (hotel, resort), accommodation\_name, location, contact\_information, and overall\_rating), detailing room types (standard, deluxe, and suite), Price, availability, max occupancy, and amenities. Similarly, it manages transport options (Transport id, Transport Type, Transport Name), including flights (Timetables, Seat Availability, Prices), trains (Timetables, Seat Availability, Prices) or rental cars (Car Type, Availability, Prices).

The Communication History is like a message record for the Travel Reservation System. Each message or email receives a unique ID (Message ID) for easy identification, a communication date to indicate when the communication occurred, communication type to specify whether it was an email or an SMS, and stores the actual message or email content. This helps to keep track of all the back-and-forth messages between the system and users, making it easier to handle any problems and keeping everything clear.

Top of Form

The Feedback Entity is comparable to a suggestion box for the Travel Reservation System. Each feedback receives a unique ID (Feedback ID) for easy tracking. The user's ID is linked to it, allowing us to know who provided the feedback. The system records when the feedback was given (Feedback Date) and an Experience Rating to gauge the quality of the user's experience. Users can additionally leave comments to offer more details or suggestions. This aids the system in comprehending user preferences and improving the overall travel experience.

1. **ERD of the system**



1. **Construction of the Relational Schema by using both bottom-up approach and top-down approach.**

**Bottom Up Approach:**

TravelReservationSystem ( UserId, FirstName, LastName, DateOfBirth, gender, AccountCreationDate, LastLoginDate, Email, PhoneNumber, Address, Username, Password, ReservationId, DepartureDate, ReturnDate, DestinationId, DestinationName, DestinationCountry, DestinationCity, DestinationRegion, DestinationDescription, DestinationPopularAttractions, DestinationLanguage, DestinationCurrency, CreditCardNumber, ExpirationDate, CVV, BillingAddress, BookingId, BookingStatus, UpdateTimestamp, AccommodationId, AccommodationType, AccommodationName, Location, ContactInformation, OverallRating, RoomType, Price, Availability, MaxOccupancy, Amenities, TransportId, TransportType, TransportName, FlightTimetables, FlightSeatAvailability, FlightPrices, TrainTimetables, TrainSeatAvailability, TrainPrices, CarType, CarAvailability, CarPrices, MessageId, CommunicationDate, CommunicationType, MessageContent, FeedbackId, FeedbackDate, ExperienceRating, Comments )

**1NF**

User (**UserId**, FirstName, LastName, DateOfBirth, AccountCreationDate, LastLoginDate, Address, gender, Username, Password)

PaymentInfo (**UserId**, **CreditCardNumber**, ExpirationDate, CVV, BillingAddress)

ContactInfo1 (**UserId, PhoneNumber**)

ContactInfo2 (**UserId, Email**)

Feedback (**FeedbackId, UserId**, FeedbackDate, ExperienceRating, Comments)

Reservation(**ReservationId, UserId,** DepartureDate, ReturnDate, DestinationId, DestinationName, DestinationCountry, DestinationCity, DestinationRegion, DestinationDescription, DestinationPopularAttractions, DestinationLanguage, DestinationCurrency, BookingId, BStatus, AccommodationId, AccommodationType, AccommodationName, Location, ContactInformation, OverallRating, RoomTypeId, , RoomType, Price, Availability, MaxOccupancy, Amenities, TransportId, TransportType, TransportName, FlightId, Timetables, SeatAvailability, Prices, TrainId, Timetables, SeatAvailability, Prices, CarId, CarType, Availability, Prices, MessageId, CommunicationDate, CommunicationType, Content)

**2NF**

User (**UserId**, FirstName, LastName, DateOfBirth, AccountCreationDate, LastLoginDate, Address, gender, Username, Password)

PaymentInfo (**UserId**, **CreditCardNumber**, ExpirationDate, CVV, BillingAddress)

ContactInfo1 (**UserId, PhoneNumber**)

ContactInfo2 (**UserId, Email**)

Feedback (**FeedbackId, UserId**, FeedbackDate, ExperienceRating, Comments)

Reservation(**ReservationId, UserId,** DepartureDate, ReturnDate, DestinationId, DestinationName, DestinationCountry, DestinationCity, DestinationRegion, DestinationDescription, DestinationPopularAttractions, DestinationLanguage, DestinationCurrency, BookingId, BStatus, AccommodationId, AccommodationType, AccommodationName, Location, ContactInformation, OverallRating, RoomTypeId, AccommodationId, RoomType, Price, Availability, MaxOccupancy, Amenities, TransportId, TransportType, TransportName, FlightId, TransportId, Timetables, SeatAvailability, Prices, TrainId, TransportId, Timetables, SeatAvailability, Prices, CarId, TransportId, CarType, Availability, Prices, MessageId, CommunicationDate, CommunicationType, Content)

**3NF**

User (**UserId**, FirstName, LastName, DateOfBirth, AccountCreationDate, LastLoginDate, Address, gender, Username, Password)

PaymentInfo (**UserId**, **CreditCardNumber**, ExpirationDate, CVV, BillingAddress)

ContactInfo1 (**UserId, PhoneNumber**)

ContactInfo2 (**UserId, Email**)

Feedback (**FeedbackId, UserId**, FeedbackDate, ExperienceRating, Comments)

Reservation (**ReservationId, UserId,** DepartureDate, ReturnDate, DestinationId, BookingId, AccommodationId, TransportId, MessageId)

Destination (**DestinationId**, DestinationName, DestinationCountry, DestinationCity, DestinationRegion, DestinationDescriptions, DestinationLanguage, DestinationCurrency)

PopularAttractions (**DestinationId, AttractionId**, AttractionName)

Booking (**BookingId**, BStatus,updateTimeStamp)

Message (**MessageId**, CommunicationDate, CommunicationType, Content)

Accommodation (**AccommodationId**, AccommodationType, AccommodationName, Location, ContactInformation, OverallRating)

RoomAccommodationInfo (**AccommodationId, RoomId**, Availability)

Room (**RoomId**, RoomType, Price, MaxOccupancy, Amenities)

Transport (**TransportId**, TransportType, TransportName)

Flight (**TransportId, FlightId**, FlightTimetables, FlightSeatAvailability, FlightPrices)

Train (**TransportId, TrainId**, TrainTimetables, TrainSeatAvailability, TrainPrices)

Car (**TransportId**, **CarId**, CarType, CarAvailability, CarPrices)

**Top Down Approach:**

User (**UserId**, FirstName, LastName, DateOfBirth, AccountCreationDate, LastLoginDate, Address, gender, Username, Password)

ContactInfo1 (**UserId, Email)**

ContactInfo2 (**UserId, PhoneNumber**)

PaymentInfo (**UserId**, **CreditCardNumber**, ExpirationDate, CVV, BillingAddress)

Feedback (**FeedbackId, UserId**, FeedbackDate, ExperienceRating, Comments)

Reservation (**ReservationId, UserId,** DepartureDate, ReturnDate, DestinationId, BookingId, AccommodationId, TransportId, MessageId)

Booking (**BookingId**, BStatus,updateTimeStamp)

Message (**MessageId**, CommunicationDate, CommunicationType, Content)

Accommodation (**AccommodationId**, AccommodationType, AccommodationName, Location, ContactInformation, OverallRating)

Transport (**TransportId**, TransportType, TransportName)

Destination (**DestinationId**, DestinationName, DestinationCountry, DestinationCity, DestinationRegion, DestinationDescriptions, DestinationLanguage, DestinationCurrency)

PopularAttractions (**DestinationId, AttractionId**, AttractionName)

RoomAccommodationInfo (**AccommodationId, RoomId**, Availability)

Room (**RoomId**, RoomType, Price, MaxOccupancy, Amenities)

Flight (**TransportId, FlightId**, FlightTimetables, FlightSeatAvailability, FlightPrices)

Train (**TransportId, TrainId**, TrainTimetables, TrainSeatAvailability, TrainPrices)

Car (**TransportId**, **CarId**, CarType, CarAvailability, CarPrices)

1. **Description of the relations in the following format.**

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: User1 | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| UserId | Number | 4 | PK |
| FirstName | Varchar2 | 25 | Not Null |
| LastName | Varchar2 | 25 | Not Null |
| DateOfBirth | Date |  |  |
| AccountCreationDate | Date |  | Default(sysdate) |
| LastLoginDate | Date |  |  |
| Address | Varchar2 | 200 |  |
|  |  |  |  |
|  |  |  |  |
| Username | Varchar2 | 25 | Not Null |
|  |  |  |  |
| Password | Varchar2 | 8 | Not Null |
| gender | char | 6 | Can be ‘F’, ‘M’ or ‘Other’ |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: PaymentInfo | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| UserId | Number | 4 | PK,FK(user1) |
| CreditCardNo | Varchar2 | 24 | PK, Check Length in(24,16,11,10) |
| ExpirationDate | Date |  | Not Null |
| CVV | Number | 4 | Not Null |
| BillingAddress | Varchar2 | 200 | Not Null |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: UserEmails | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| UserId | Number | 4 | PK,FK(user1) |
| Email | Varchar2 | 25 | PK |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: Feedback | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| UserId | Number | 4 | PK,FK(user) |
| FeedbackId | Number | 4 | PK |
| FeedbackDate | Date |  | Default(sysdate) |
| ExperienceRating | Number | 5 | Can be 1, 2, 3, 4, 5 |
| Comment | Varchar2 | 100 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: PhoneNumber | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| UserId | Number | 4 | PK,FK(user) |
| PhoneNo | Varchar2 | 11 | PK |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: Accommodation | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| AccommodationId | Number | 4 | PK |
| AccommodationType | Varchar2 | 10 | Can be ‘Resort’ or ‘Hotel’ |
| AccommodationName | Varchar2 | 200 | Not Null |
| Location | Varchar2 | 200 | Not Null |
| ContactInformation | Varchar2 | 11 | Not Null |
| OverallRating | Number | 5 | Can be 1,2,3,4,5 |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: Room | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| RoomId | Number | 4 | PK |
| RoomType | Varchar2 | 50 | Not Null |
| Price | Timestamp | (7,1) | Check (Price >0),Not null |
| MaxOccupancy | Number | 1 | Check (MaxOccupancy >0 and <=4), Not null |
| Amenities | Varchar2 | 300 | Not null |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: RoomAccommodationInfo | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| RoomId | Number | 4 | PK,FK |
| AccommodationId | Number | 4 | PK,FK |
| Availability | Varchar2 | 3 | Check (Availability is Yes or No),  Not Null |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: Reservation | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| ReservationId | Number | 4 | PK |
| UserId | Number | 4 | PK, ,FK(User) |
| DepartureDate | Date |  |  |
| ReturnDate | Date |  |  |
| DestinationId | Number | 4 | FK(Destination) |
| BookingId | Number | 4 | FK(Booking) |
| AccommodationId | Number | 4 | FK(Accommodation) |
| MessageId | Number | 4 | FK(Message) |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: Booking | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| BookingId | Number | 4 | PK |
| Status | Varchar2 |  | Can be( pending, confirmed, cancelled) |
| updateTimeStamp | TimeStamp | 200 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: Destination | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| DestinationId | Number | 4 | PK |
| DestinationName | Varchar2 | 225 | Not null |
| DestinationCountry | Varchar2 | 50 | Not null |
| DestinationCity | VARCHAR2 | 50 | Not null |
| DestinationRegion | Varchar2 | 50 | Not null |
| DestinationDescriptions | Varchar2 | 300 |  |
| DestinationLanguage | Varchar2 | 50 | Not null |
| DestinationCurrency | Varchar2 | 10 | Not null |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: PopularAttractions | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| DestinationId | Number | 4 | PK,FK(Destination) |
| AttractionId | Number | 4 | PK |
| AttractionName | Varchar2 | 200 | Not null |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: Transport | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| TransportId | Number | 4 | PK |
| TransportType | Varchar2 | 50 | Not null |
| TransportName | Varchar2 | 200 | Not null |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: Train | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| TransportId | Number | 4 | PK,FK(Transport) |
| TrainId | Number | 4 | PK |
| TrainTimetables | Timestamp |  | Not null |
| TrainSeatAvailability | Varchar2 | 3 | Check (Yes or NO),  Not null |
| TrainPrices | number | 5 | Not null |

|  |  |
| --- | --- |
| Table Name: Car | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| TransportId | Number | 4 | PK,FK(Transport) |
| CarId | Number | 4 | PK |
| CarType | Timestamp |  | Not null |
| CarAvailability | Varchar2 | 3 | Check (Yes or NO),  Not null |
| CarPrices | number | 7 | Not null |

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name: Flight | | | |
| Attribute | **Data Types** | **Size** | **Constraints** |
| TransportId | Number | 4 | PK,FK(Transport) |
| FlightId | Number | 4 | PK |
| FlightTimetables | Timestamp |  | Not null |
| FlightSeatAvailability | Varchar2 | 3 | Check (Yes or NO),  Not null |
| FlightPrices | number | 10 | Not null |

1. **CREATE TABLE statements for all the relations of your system.**

**User:**

Create table user1

(

userid number(10),

firstName varchar2(25) not null,

lastName varchar2(25) not null,

dateofbirth date,

AccountCreationDate date default sysdate,

lastLoginDate date default sysdate,

address varchar2(200),

username varchar2(25) not null,

password varchar2(20) not null,

constraint user\_userid\_pk primary key (userid)

);

**Feedback:**

CREATE TABLE feedback

(

userid NUMBER(4),

feedbackid NUMBER(4),

feedbackdate DATE DEFAULT SYSDATE,

experiencerating NUMBER(5),

feedbackComment varchar2(100),

CONSTRAINT feed\_back\_pk PRIMARY KEY(userid, feedbackid),

CONSTRAINT feed\_back\_fk FOREIGN KEY(userid) REFERENCES user1(userid),

CONSTRAINT feed\_back\_chk CHECK(experiencerating IN(1, 2, 3, 4, 5))

);

**Message:**

Create table Message

(

MessageId number(4),

CommunicationDate date default sysdate,

CommunicationType varchar2(5),

Content varchar2(250),

Constraint message\_pk primary key(MessageId),

Constraint message\_commType\_check check (CommunicationType in('SMS', 'EMAIL' ))

);

**Destination:**

CREATE TABLE destination

(

destinationid NUMBER(4),

destinationname VARCHAR2(225) CONSTRAINT dest\_name\_nn NOT NULL,

destinationcountry VARCHAR2(50) CONSTRAINT dest\_coun\_nn NOT NULL,

destinationregion VARCHAR2(50) CONSTRAINT dest\_reg\_nn NOT NULL,

destinationcity VARCHAR2(50) CONSTRAINT dest\_city\_nn NOT NULL,

destinationdescriptions VARCHAR2(300),

destinationlanguage VARCHAR2(50) CONSTRAINT destination\_language\_nn NOT NULL,

destinationcurrency VARCHAR2(10) CONSTRAINT destination\_currency\_nn NOT NULL,

CONSTRAINT destination\_id\_pk PRIMARY KEY(destinationid)

);

**Booking:**

Create table booking

(

bookingId Number(4),

status varchar2(20),

updateTimeStamp timestamp(6),

constraint booking\_id\_pk primary key(bookingId),

constraint book\_status\_check check( status in('pending', 'confirmed','cancelled'))

);

**Reservation:**

CREATE TABLE reservation

(

reservationid NUMBER(4),

userid NUMBER(4),

departuredate DATE,

returndate DATE,

destinationid NUMBER(4),

bookingid NUMBER(4),

accommodationid NUMBER(4),

messageid NUMBER(4),

CONSTRAINT reservation\_pk PRIMARY KEY (reservationid, userid),

CONSTRAINT reservation\_fk\_user FOREIGN KEY (userid) REFERENCES user1(userid),

CONSTRAINT reservation\_fk\_destination FOREIGN KEY (destinationid) REFERENCES destination(destinationid),

CONSTRAINT reservation\_fk\_booking FOREIGN KEY (bookingid) REFERENCES booking(bookingid),

CONSTRAINT reservation\_fk\_accommodation FOREIGN KEY (accommodationid) REFERENCES accommodation(accommodationid),

CONSTRAINT reservation\_fk\_message FOREIGN KEY (messageid) REFERENCES message(messageid)

);

**Accommodation:**

create table accommodation

(

accomodationId number(4),

acc\_type varchar2(10),

acc\_name varchar2(200) not null,

location varchar2(200) not null,

contactinformation varchar2(11) not null,

overallRating number(5),

Constraint acc\_id\_pk primary key(accomodationId),

constraint acc\_type\_chk check (acc\_type in ('Resort', 'Hotel')),

constraint overallRating\_chk check (overallRating in (1, 2, 3, 4, 5))

);

**Room:**

CREATE TABLE room

(

roomid NUMBER(4) CONSTRAINT room\_id\_pk PRIMARY KEY,

roomtype VARCHAR2(50) NOT NULL,

price NUMBER(7) NOT NULL,

maxoccupancy NUMBER(1) NOT NULL,

amenities VARCHAR2(300) NOT NULL,

CONSTRAINT price\_check CHECK (price > 0),

CONSTRAINT max\_occupancy\_check CHECK (maxoccupancy > 0 AND maxoccupancy <= 4)

);

**Popular Attractions:**

CREATE TABLE popular\_attractions

(

destinationid NUMBER(4),

attractionid NUMBER(4),

attractionname VARCHAR2(200) NOT NULL,

CONSTRAINT dest\_id\_fk FOREIGN KEY (destinationid) REFERENCES destination(destinationid),

CONSTRAINT attraction\_id\_pk PRIMARY KEY (attractionid)

);

**Transport**

CREATE TABLE transport

(

transportid NUMBER(4),

transporttype VARCHAR2(50) NOT NULL,

transportname VARCHAR2(200) NOT NULL,

CONSTRAINT transport\_id\_pk PRIMARY KEY(transportid)

);

**Train**

CREATE TABLE train

(

transportid NUMBER(4),

trainid NUMBER(4),

train\_timetables TIMESTAMP NOT NULL,

trainseatavailability VARCHAR2(3) NOT NULL,

trainprices NUMBER(5) NOT NULL,

CONSTRAINT train\_seat\_chk CHECK (trainseatavailability IN ('Yes', 'No')),

CONSTRAINT train\_id\_pk PRIMARY KEY (transportid, trainid), -- Added the missing comma

CONSTRAINT transport\_id\_fk FOREIGN KEY (transportid) REFERENCES transport(transportid)

);

**Car**

CREATE TABLE car

(

transportid NUMBER(4),

carid NUMBER(4),

cartype VARCHAR2(50) NOT NULL,

caravailability VARCHAR2(3) NOT NULL,

carprices NUMBER(7) NOT NULL,

CONSTRAINT car\_transport\_id\_fk FOREIGN KEY (transportid) REFERENCES transport(transportid),

CONSTRAINT car\_id\_pk PRIMARY KEY (carid),

CONSTRAINT car\_availability\_check CHECK (caravailability IN ('Yes', 'No'))

);

**Flight**

CREATE TABLE flight

(

transportid NUMBER(4),

flightid NUMBER(4),

flighttimetables TIMESTAMP NOT NULL,

flightseatavailability VARCHAR2(3) NOT NULL,

flightprices NUMBER(10) NOT NULL,

CONSTRAINT flight\_fk FOREIGN KEY (transportid) REFERENCES transport(transportid),

CONSTRAINT flight\_id\_pk PRIMARY KEY (flightid),

CONSTRAINT flight\_check CHECK (flightseatavailability IN ('Yes', 'No'))

);

**User Emails**

create table userEmails

(

Userid number(4),

email varchar2(25),

Constraint pk\_useremails primary key (userid, email),

constraint fk\_useremails\_user foreign key (userid) references user1 (userid)

);

**User PhoneNumber**

create table phoneNumber

(

Userid number (4),

phoneNo varchar2(11),

constraint phone\_pk primary key (userid, phoneno),

constraint phone\_fk foreign key (userid) references user1(userid)

);

1. **Desing of at least two VIEWS, that you feel are the most important.**

**View 1 :**

CREATE OR REPLACE VIEW UserDetailsView AS

SELECT

u.UserId,

u.FirstName,

u.LastName,

u.DateOfBirth,

u.LastLoginDate,

u.Address,

u.Gender,

u.Username,

ci1.PhoneNo AS ContactPhoneNumber,

ci2.Email AS ContactEmail,

r.DepartureDate,

r.ReturnDate

FROM

User1 u

JOIN

phoneNumber ci1 ON u.UserId = ci1.UserId

JOIN

userEmails ci2 ON u.UserId = ci2.UserId

JOIN

Reservation r ON u.UserId = r.UserId;

**View 2 :**

CREATE OR REPLACE VIEW DestinationAttractionsView AS

SELECT

d.DestinationId,

d.DestinationName,

d.DestinationCountry,

d.DestinationCity,

d.DestinationRegion,

d.DestinationDescriptions,

d.DestinationLanguage,

d.DestinationCurrency,

pa.AttractionId,

pa.AttractionName

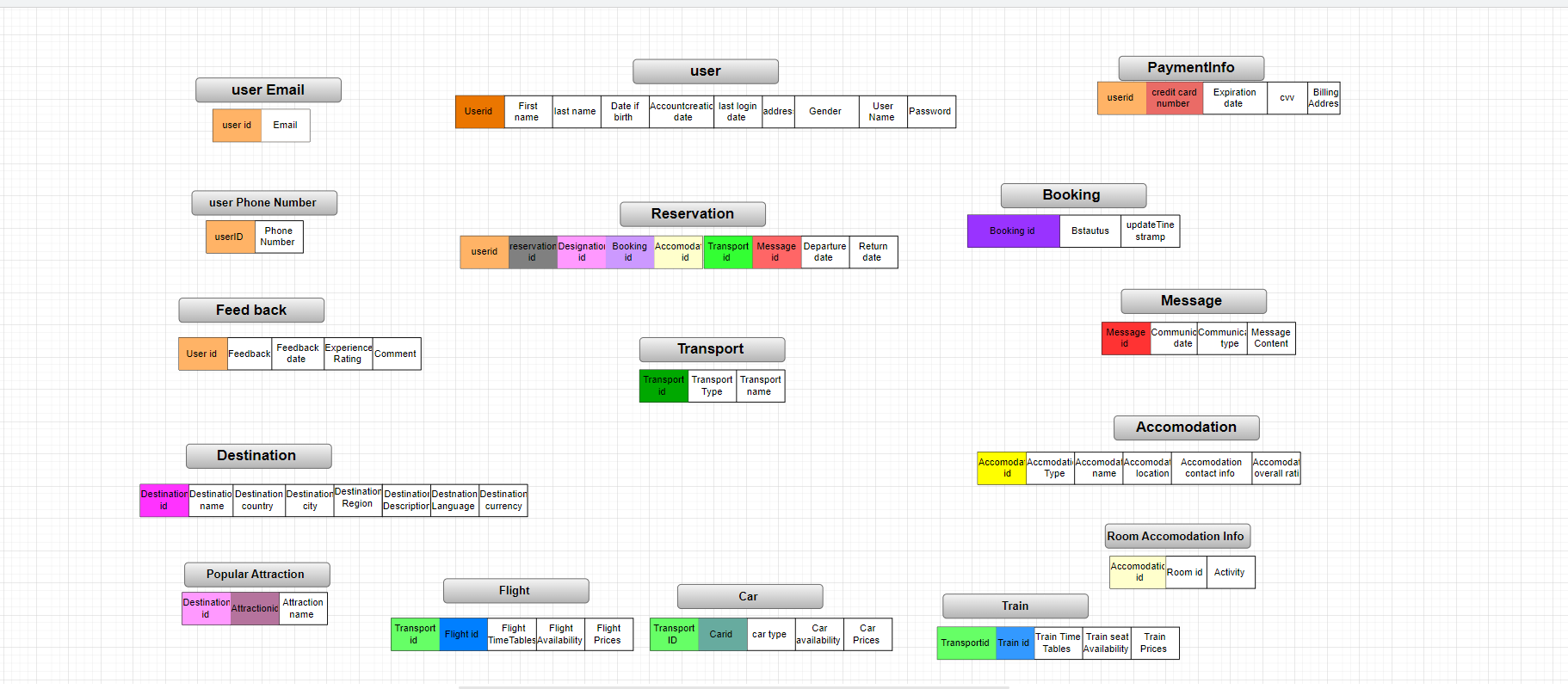
FROM

Destination d

JOIN

PopularAttractions pa ON d.DestinationId = pa.DestinationId;

1. **Relational data model showing the association among different relations of the relational schema.**



**9.Queries for the report generation :**

1. SELECT u.Userid, u.FirstName, u.LastName, r.ReservationId, r.DepartureDate, r.ReturnDate

FROM user1 u

JOIN reservation r ON u.Userid = r.userid;

2. SELECT a.AccomodationId, a.Acc\_Name, r.RoomId, r.RoomType, r.MaxOccupancy

FROM accommodation a

JOIN room r ON a.AccomodationId = r.AccommodationId

WHERE r.MaxOccupancy > 0;

3. SELECT u.Userid, u.FirstName, u.LastName, f.FeedbackId, f.ExperienceRating, f.Comment

FROM user1 u

JOIN feedback f ON u.Userid = f.userid;

4. SELECT b.BookingId, b.Status, b.UpdateTimestamp, r.ReservationId, r.DepartureDate, r.ReturnDate

FROM booking b

JOIN reservation r ON b.BookingId = r.BookingId;

5. SELECT b.BookingId, b.Status, b.UpdateTimestamp, r.ReservationId, r.DepartureDate, r.ReturnDate

FROM booking b

JOIN reservation r ON b.BookingId = r.BookingId;

1. **Demonstration of at least two functions, two stored procedures, and two database triggers.**

**FUNCTION 01:**

Create or Replace Function calculateAge(dateOfBirth user1.dateOfBirth%type)

return Number is

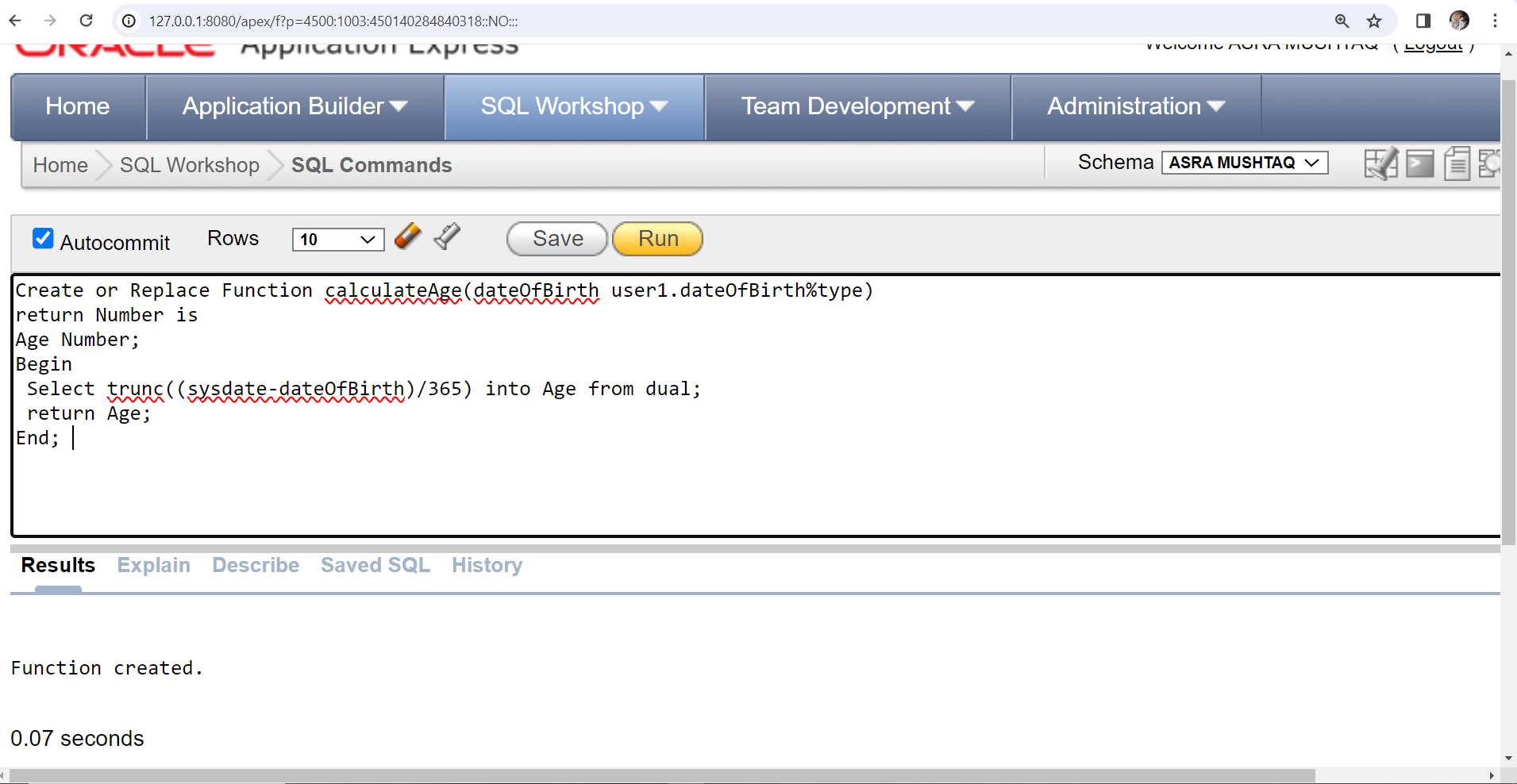
Age Number;

Begin

Select trunc((sysdate-dateOfBirth)/365) into Age from dual;

return Age;

End;



**FUNCTION 02:**

Create or Replace Function getFeedbackCountForUser(Id Number)

return Number is

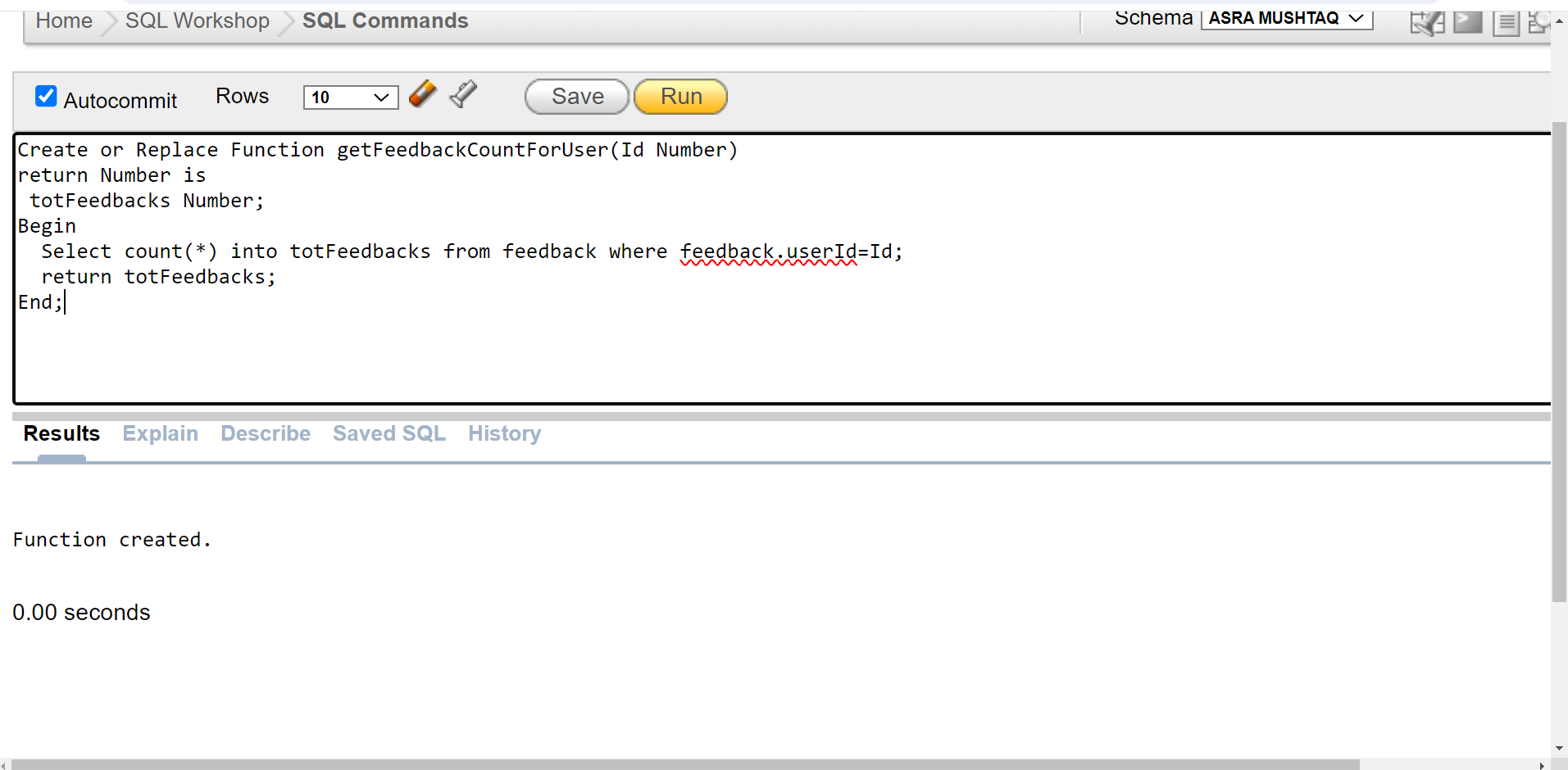
totFeedbacks Number;

Begin

Select count(\*) into totFeedbacks from feedback where feedback.userId=Id;

return totFeedbacks;

End;



**TRIGGER 01:**

CREATE OR REPLACE TRIGGER "PreventDeletionOfBooking"

BEFORE DELETE ON booking

FOR EACH ROW

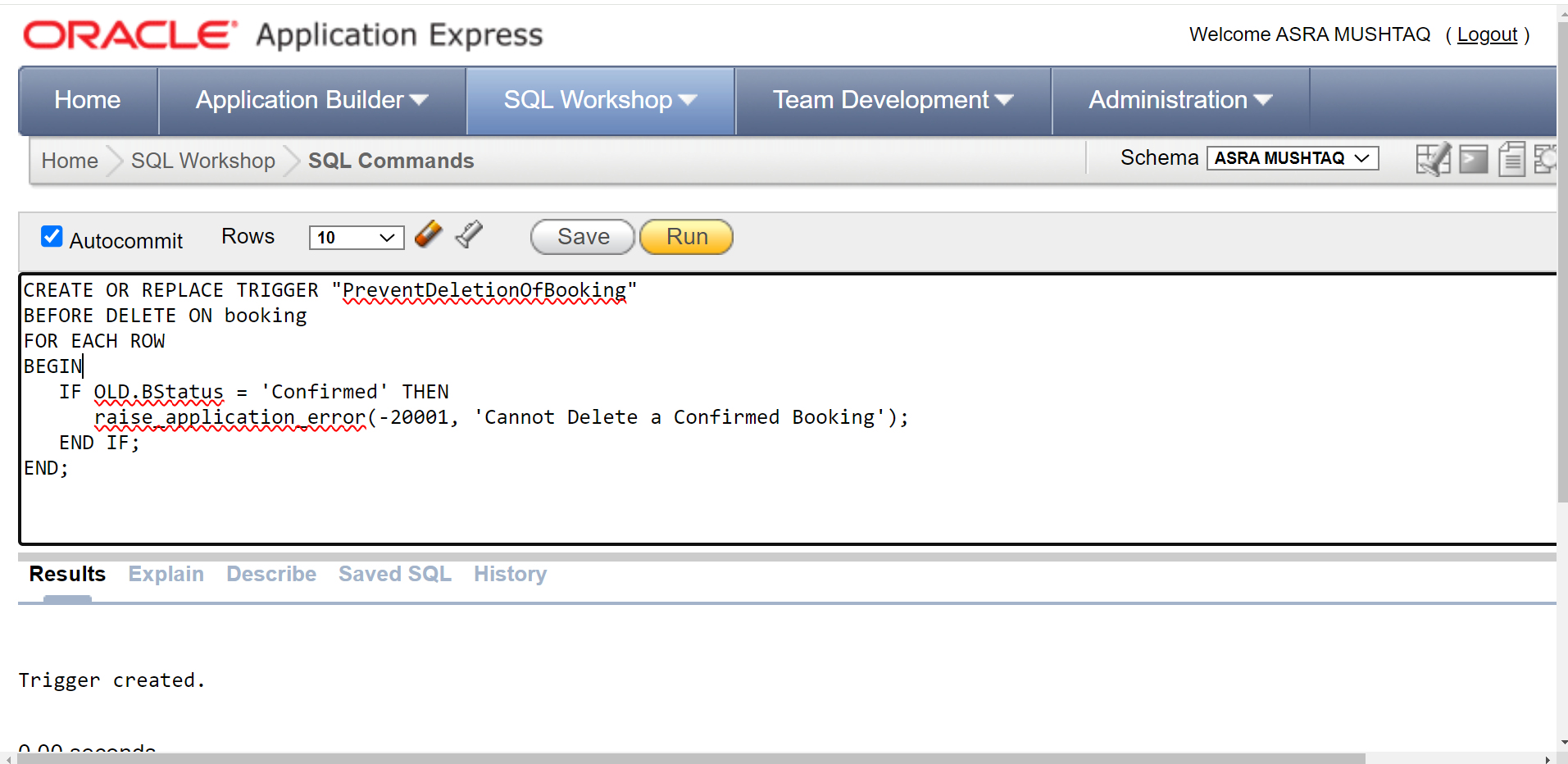
BEGIN

IF OLD.BStatus = 'Confirmed' THEN

raise\_application\_error(-20001, 'Cannot Delete a Confirmed Booking');

END IF;

END;



**TRIGGER 02:**

CREATE OR REPLACE TRIGGER reservationCancellation

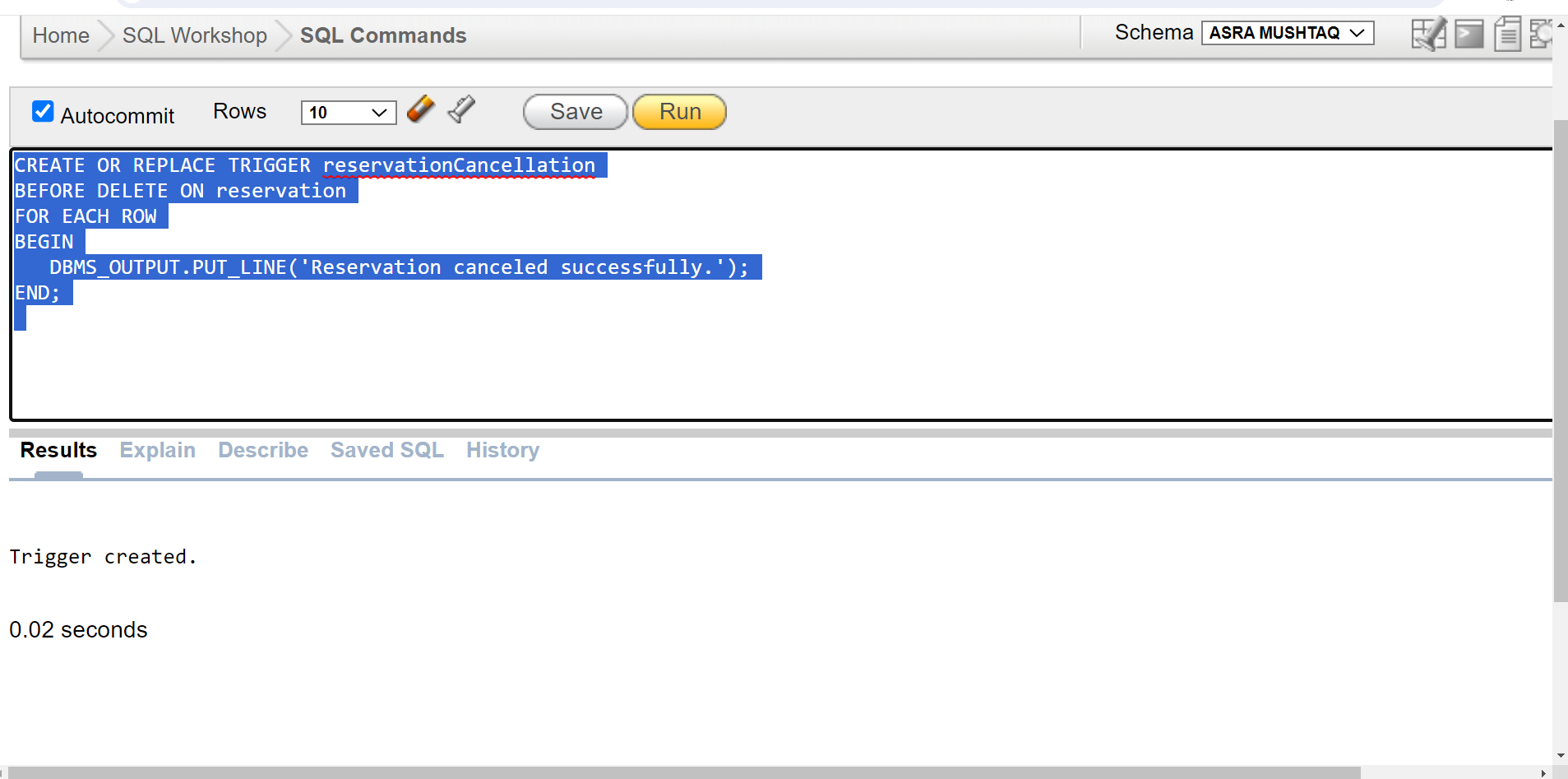
BEFORE DELETE ON reservation

FOR EACH ROW

BEGIN

DBMS\_OUTPUT.PUT\_LINE('Reservation canceled successfully.');

END;



**PROCEDURE 01:**

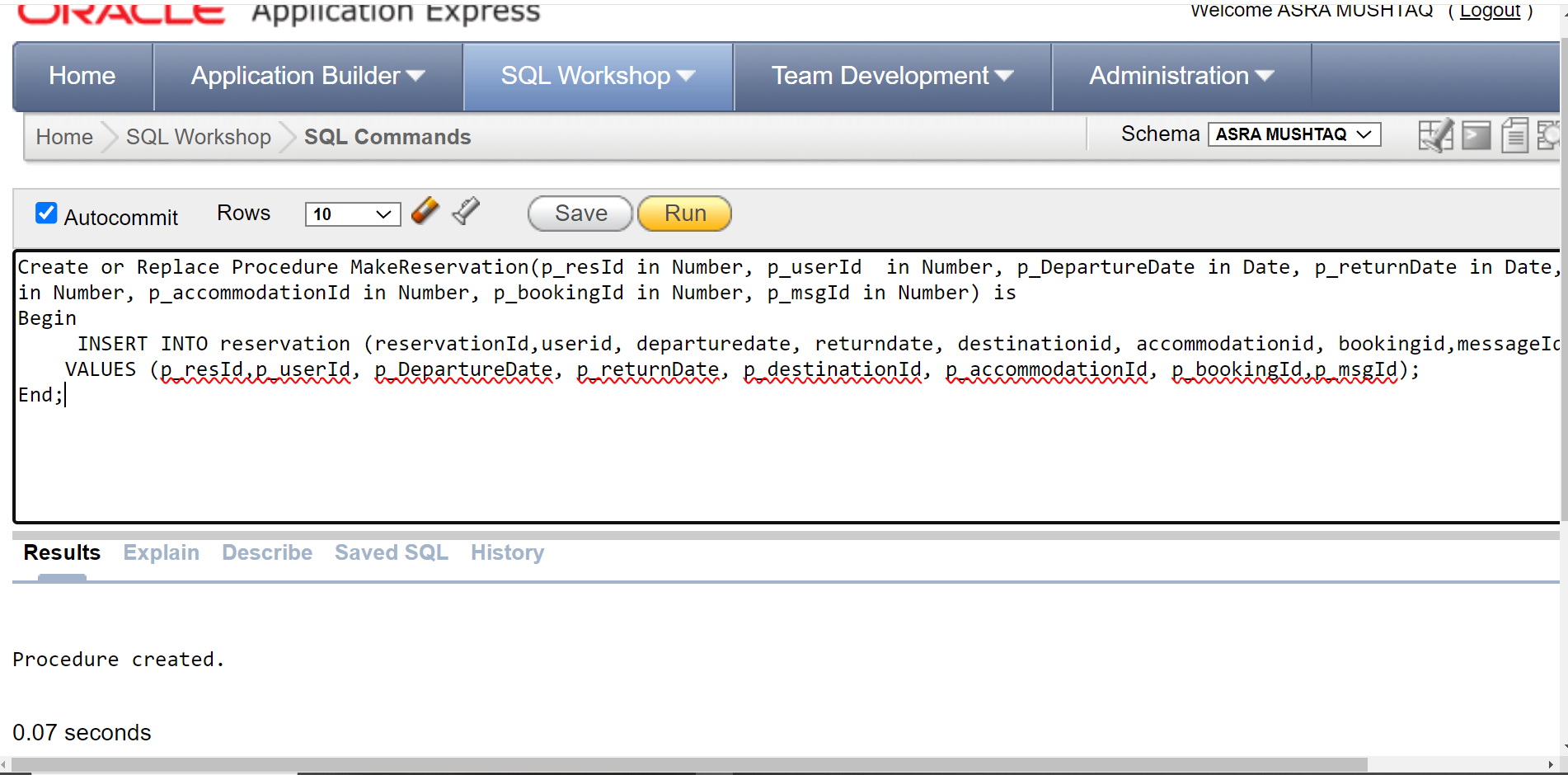
Create or Replace Procedure MakeReservation(p\_resId in Number, p\_userId in Number, p\_DepartureDate in Date, p\_returnDate in Date, p\_destinationId in Number, p\_accommodationId in Number, p\_bookingId in Number, p\_msgId in Number) is

Begin

INSERT INTO reservation (reservationId,userid, departuredate, returndate, destinationid, accommodationid, bookingid,messageId)

VALUES (p\_resId,p\_userId, p\_DepartureDate, p\_returnDate, p\_destinationId, p\_accommodationId, p\_bookingId,p\_msgId);

End;



**PROCEDURE 02:**

Create or Replace Procedure UpdateLastLogin(Id In Number)

Is

Begin

Update user1

Set lastloginDate = sysdate

where user1.userId = id;

End;

