**XML Files For all the entities:**

1. Login.xml
2. <?xml version=*"1.0"* encoding=*"UTF-8"*?>
3. <Login>
4. <loginDetails Customerid=*"001"*>
5. <CustomerTitle>Ms</CustomerTitle>
6. <CustomerFirstName>Vaishnavi</CustomerFirstName>
7. <CustomerLastName>Bhadresh</CustomerLastName>
8. <pWord>234567</pWord>
9. <userType>admin</userType>
10. <emailID>vaishubh12@gmail.com</emailID>
11. </loginDetails>
12. <loginDetails Customerid=*"002"*>
13. <CustomerTitle>Mr</CustomerTitle>
14. <CustomerFirstName>Manish</CustomerFirstName>
15. <CustomerLastName>M K</CustomerLastName>
16. <pWord>234587</pWord>
17. <userType>admin</userType>
18. <emailID>manishmukund@gmail.com</emailID>
19. </loginDetails>
20. <loginDetails Customerid=*"003"*>
21. <CustomerTitle>Ms</CustomerTitle>
22. <CustomerFirstName>Inchara</CustomerFirstName>
23. <CustomerLastName>Desai</CustomerLastName>
24. <pWord>234570</pWord>
25. <userType>customer</userType>
26. <emailID>incharadesai@gmail.com</emailID>
27. </loginDetails>
28. <loginDetails Customerid=*"004"*>
29. <CustomerTitle>Mr</CustomerTitle>
30. <CustomerFirstName>Jaswanth</CustomerFirstName>
31. <CustomerLastName>Jagdeesh</CustomerLastName>
32. <pWord>234577</pWord>
33. <userType>user</userType>
34. <emailID>jjagdessh@gmail.com</emailID>
35. </loginDetails>
37. </Login>

2. ApproveCarDetails.xml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<carDetails\_admin>

<carDetails Carid=*"01"*>

<CarName>Focus</CarName>

<CarModel>Ford</CarModel>

<Location>Chicago</Location>

</carDetails>

<carDetails Carid=*"02"*>

<CarName>Feista</CarName>

<CarModel>Ford</CarModel>

<Location>Niles</Location>

</carDetails>

<carDetails Carid=*"03"*>

<CarName>Vaishnavi</CarName>

<CarModel>Mustang</CarModel>

<Location>Naperville</Location>

</carDetails>

</carDetails\_admin>

3. AddCar.xml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<addcar>

<AddCar CarId=*"R12"*>

<CarName>Suburban</CarName>

<CarModel>Chevrolet</CarModel>

<CarNumber>JI90G2</CarNumber>

<CarColour>Silver</CarColour>

<SeatingCapacity>8</SeatingCapacity>

<DriverName>Mark</DriverName>

<DriverId>DL345</DriverId>

</AddCar>

<AddCar CarId=*"R14"*>

<CarName>Malibu</CarName>

<CarModel>Chevrolet</CarModel>

<CarNumber>J93D25</CarNumber>

<CarColour>Maroon</CarColour>

<SeatingCapacity>5</SeatingCapacity>

<DriverName>Carlson</DriverName>

<DriverId>DL355</DriverId>

</AddCar>

<AddCar CarId=*"R13"*>

<CarName>Fit</CarName>

<CarModel>Honda</CarModel>

<CarNumber>J920HU</CarNumber>

<CarColour>Green</CarColour>

<SeatingCapacity>4</SeatingCapacity>

<DriverName>Maddy</DriverName>

<DriverId>DL367</DriverId>

</AddCar>

</addcar>

4. BookingDetails.xml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<booking\_details>

<BookingDetails booking\_id=*"B777"*>

<CustomerTitle>Mr</CustomerTitle>

<CustomerFirstName>John</CustomerFirstName>

<CustomerLastName>Howell</CustomerLastName>

<CustomerID>C123</CustomerID>

<Location>Chicago</Location>

<BookingDate>05/03/2018</BookingDate>

<StartDate>06/03/2018</StartDate>

<StartTime>15:30:00</StartTime>

<EndDate>08/03/2018</EndDate>

<EndTime>18:30:00</EndTime>

<BookingDuration>Calculate1</BookingDuration>

<CarName>Focus</CarName>

<CarModel>Ford</CarModel>

<CarColour>MidnightBlack</CarColour>

<CarNumber>MB007</CarNumber>

<AmountPaid>25$</AmountPaid>

</BookingDetails>

<BookingDetails booking\_id=*"B737"*>

<CustomerTitle>Ms</CustomerTitle>

<CustomerFirstName>Sophia</CustomerFirstName>

<CustomerLastName>Danon</CustomerLastName>

<CustomerID>C234</CustomerID>

<Location>Naperville</Location>

<BookingDate>05/03/2018</BookingDate>

<StartDate>07/03/2018</StartDate>

<StartTime>14:30:00</StartTime>

<EndDate>09/03/2018</EndDate>

<EndTime>17:00:00</EndTime>

<BookingDuration>Calculate2</BookingDuration>

<CarName>Fiesta</CarName>

<CarModel>Ford</CarModel>

<CarColour>OrangePeel</CarColour>

<CarNumber>OP005</CarNumber>

<AmountPaid>25$</AmountPaid>

</BookingDetails>

<BookingDetails booking\_id=*"B757"*>

<CustomerTitle>Mrs</CustomerTitle>

<CustomerFirstName>Amanda</CustomerFirstName>

<CustomerLastName>Harris</CustomerLastName>

<CustomerID>C345</CustomerID>

<Location>Niles</Location>

<BookingDate>05/03/2018</BookingDate>

<StartDate>16/03/2018</StartDate>

<StartTime>15:30:00</StartTime>

<EndDate>19/03/2018</EndDate>

<EndTime>19:30:00</EndTime>

<BookingDuration>Calculate3</BookingDuration>

<CarName>Mustang</CarName>

<CarModel>Ford</CarModel>

<CarColour>RoyalBlue</CarColour>

<CarNumber>MRF03</CarNumber>

<AmountPaid>25$</AmountPaid>

</BookingDetails>

</booking\_details>

5. mechanicServcies\_admin.xml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<mechanicServices\_admin>

<mechanicServices Carid=*"01"*>

<typeofService>Brake</typeofService>

<serviceStatus>In Progress</serviceStatus>

</mechanicServices>

<mechanicServices Carid=*"02"*>

<typeofService>Battery</typeofService>

<serviceStatus>Completed</serviceStatus>

</mechanicServices>

<mechanicServices Carid=*"03"*>

<typeofService>Tyre</typeofService>

<serviceStatus>Pending</serviceStatus>

</mechanicServices>

</mechanicServices\_admin>

6. PaymentDetails.xml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<paymentDetails>

<PaymentDetails payment\_id=*"B777"*>

<CardNumber>27824295046739230</CardNumber>

<CardType>VISA</CardType>

<CardName>Howell</CardName>

<CardExpiry>08/20</CardExpiry>

<IssueBank>BOFA</IssueBank>

<BillingAddress>Chicago</BillingAddress>

</PaymentDetails>

<PaymentDetails payment\_id=*"B789"*>

<CardNumber>24524266399504230</CardNumber>

<CardType>MasterCard</CardType>

<CardName>David</CardName>

<CardExpiry>05/22</CardExpiry>

<IssueBank>Chase</IssueBank>

<BillingAddress>Naperville</BillingAddress>

</PaymentDetails>

<PaymentDetails payment\_id=*"B767"*>

<CardNumber>4983298718363597</CardNumber>

<CardType>MAESTRO</CardType>

<CardName>Harper</CardName>

<CardExpiry>09/24</CardExpiry>

<IssueBank>PNC</IssueBank>

<BillingAddress>Niles</BillingAddress>

</PaymentDetails>

</paymentDetails>

**DAO Classes for All Entities and JDBC Connectivity Class**

1. BookingDetailsDAO

package model;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

public class bookingDetailsDao {

public void addbookingDetails(){}

public void updatebookingDetails(){}

public void deletebookingDetails(){}

public bookingDetailsDao(Connection connect){

this.connection=connect;

if(connection!=null){

System.out.println("Established Connection successfully");

}

else{

System.err.println("Failed to establish Connection");

}

}

public void bookingDetails(){

try {

statement= connection.createStatement();

String sql="LOAD XML LOCAL INFILE 'C:/Users/vaish/itmdev/src/main/webapp"

+ "/XML/booking\_details.xml' "

+ "INTO TABLE booking\_details ROWS IDENTIFIED BY '<BookingDetails>';";

statement.execute(sql);

System.out.println("XML data is pushed into the database");

} catch (SQLException e) {

System.err.println("Failed in Updating Data to OTP Table");

}

}

public static ResultSet getResultSet() throws Exception

{

ResultSet rs = null;

try

{

statement = connection.createStatement();

String sql = "select \* from login";

rs = statement.executeQuery(sql);

}

catch(SQLException e)

{

System.out.println(e.getMessage());

}

return rs;

}

public void close(){

try {

statement.close();

connection.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

private static Statement statement = null;

private static Connection connection=null;

}

1. carDetailsDao.java

package model;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

public class carDetailsDao {

public carDetailsDao(Connection connect) {

this.connection = connect;

if (connection != null) {

System.out.println("Established Connection successfully");

} else {

System.err.println("Failed to establish Connection");

}

}

public void approveCarDetails\_insert() {

try {

statement = connection.createStatement();

String sql = "LOAD XML LOCAL INFILE 'C:/Users/vaish/itmdev/src/main/webapp" + "/XML/ApproveCarDetails.xml' "

+ "INTO TABLE carDetails\_admin ROWS IDENTIFIED BY '<carDetails>';";

statement.execute(sql);

System.out.println("XML data is pushed into the database");

} catch (SQLException e) {

System.err.println("Failed in Updating Data to OTP Table");

}

}

public static ResultSet getResultSet() throws Exception {

ResultSet rs = null;

try {

statement = connection.createStatement();

String sql = "select \* from login";

rs = statement.executeQuery(sql);

} catch (SQLException e) {

System.out.println(e.getMessage());

}

return rs;

}

public void addCar(){

try {

statement = connection.createStatement();

String sql = "LOAD XML LOCAL INFILE 'C:/Users/vaish/itmdev/src/main/webapp"

+ "/XML/addCar.xml' "

+ "INTO TABLE addCar ROWS IDENTIFIED BY '<AddCar>';";

statement.execute(sql);

System.out.println("XML data is pushed into the database");

} catch (SQLException e) {

System.err.println("Failed in Updating Data to OTP Table");

}

}

public void close() {

try {

statement.close();

connection.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

private static Statement statement = null;

private static Connection connection = null;

}

1. Connector.java

/\*\*Controller.java 10/01/2017 4:32 pm

\*Program to Database Connectivity operations.

\*Programmer: Vaishnavi Bhadresh , FileName: Connector.java , Lab Number: Lab 4

\* @author vaish

\*/

// packageName

package model;

// import statements

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

public class Connector {

// Declaring Connection Variable

public static Connection connection= null;

public Connector(){

try {

// Invoking Connection method to establish connection to Database

System.out.println("Trying to establish MySQL JDBC Driver connection");

} catch (Exception e) {

System.err.println("Failed to Connect to Database");

}

}

// Method to connect to Database

public Connection getConnection() throws SQLException, ClassNotFoundException{

Class.forName("com.mysql.jdbc.Driver");

System.out.println("MySQL JDBC Driver Registered");

return connection = DriverManager.getConnection("jdbc:mysql://127.0.0.1:3306/carserviceschicago", "vaish","root123");

}

}

1. daoModel Connection Class

/\*\*Controller.java 10/01/2017 4:32 pm

\*Program to Database CRUD operations.

\*Programmer: Vaishnavi Bhadresh , FileName: daoModel.java , Lab Number: Lab 4

\* @author vaish

\*/

// packageName

package model;

// import statements

import java.sql.Connection;

import java.sql.DatabaseMetaData;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.Scanner;

public class daoModel {

// daoModel Constructor to create Connection with Database

public daoModel() {

connector = new Connector();

connectDB();

}

private void connectDB() {

try {

connection = connector.getConnection();

System.out.println("MySQL JDBC Driver established successfully");

} catch (Exception e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

public Connection getConnection() {

return connection;

}

public void updateOTP(String otp,String id) {

String query = "Insert into login\_db (otp,id) values("+otp+","+ id+")";

try {

statement = connection.createStatement();

statement.executeUpdate(query);

} catch (Exception e) {

}

}

public void close() {

try {

connection.close();

connection = null;

} catch (Exception e) {

System.err.println("Failed to close DB Connection");

}

}

// declaring class variables

private static Connector connector = null;

private static Connection connection = null;

private static Statement statement = null;

}

1. GetConnection.java

package model;

import java.sql.Connection;

import java.sql.DriverManager;

public class GetConnection {

public GetConnection(){

System.out.println("Establishing Connection to MySQL DataBase");

}

public Connection getConnection(){

try{

Class.forName("com.mysql.jdbc.Driver");

System.out.println("MySQL JDBC Driver Registered");

connection = DriverManager.getConnection("jdbc:mysql://www.papademas.net:3306/510labs?", "db510","510");

return connection;

}

catch(Exception e){

System.err.println("Failed to establish Connection to DataBase "+e);

}

return null;

}

private static Connection connection=null;

}

1. Login.Dao

package model;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

public class LoginDao {

public LoginDao(Connection connect){

this.connection=connect;

if(connection!=null){

System.out.println("Established Connection successfully");

}

else{

System.err.println("Failed to establish Connection");

}

}

public void InsertData(){

try {

statement= connection.createStatement();

String sql="LOAD XML LOCAL INFILE 'C:/Users/vaish/itmdev/src/main/webapp"

+ "/XML/LoginDetails.xml' "

+ "INTO TABLE login ROWS IDENTIFIED BY '<loginDetails>';";

statement.execute(sql);

System.out.println("XML data is pushed into the database");

} catch (SQLException e) {

System.err.println("Failed in Updating Data to OTP Table");

}

}

public ResultSet getResultSet() throws Exception

{

ResultSet rs = null;

try

{

statement = connection.createStatement();

String sql = "select \* from login";

rs = statement.executeQuery(sql);

}

catch(SQLException e)

{

System.out.println(e.getMessage());

}

return rs;

}

public void close(){

try {

statement.close();

connection.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

private static Statement statement = null;

private static Connection connection=null;

}

1. MechanicServicesDao

package model;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

public class mechanicServicesdao {

public void addmechanicServices(){}

public void updatemechanicServices(){}

public void deletemechanicServices(){}

public mechanicServicesdao(Connection connect){

this.connection=connect;

if(connection!=null){

System.out.println("Established Connection successfully");

}

else{

System.err.println("Failed to establish Connection");

}

}

public void approveMechanicServices(){

try {

statement= connection.createStatement();

String sql="LOAD XML LOCAL INFILE 'C:/Users/vaish/itmdev/src/main/webapp"

+ "/XML/MechanicServices\_admin.xml' "

+ "INTO TABLE mechanicServices\_admin ROWS IDENTIFIED BY '<mechanicServices>';";

statement.execute(sql);

System.out.println("XML data is pushed into the database");

} catch (SQLException e) {

System.err.println("Failed in Updating Data to OTP Table");

}

}

public static ResultSet getResultSet() throws Exception

{

ResultSet rs = null;

try

{

statement = connection.createStatement();

String sql = "select \* from login";

rs = statement.executeQuery(sql);

}

catch(SQLException e)

{

System.out.println(e.getMessage());

}

return rs;

}

public void close(){

try {

statement.close();

connection.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

private static Statement statement = null;

private static Connection connection=null;

}

1. payemntDetailsDao.java

package model;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

public class paymentDetailsDao {

public paymentDetailsDao(Connection connect){

this.connection=connect;

if(connection!=null){

System.out.println("Established Connection successfully");

}

else{

System.err.println("Failed to establish Connection");

}

}

public void paymentDetails(){

try {

statement= connection.createStatement();

String sql="LOAD XML LOCAL INFILE 'C:/Users/vaish/itmdev/src/main/webapp"

+ "/XML/paymentDetails.xml' "

+ "INTO TABLE PaymentDetails ROWS IDENTIFIED BY '<PaymentDetails>';";

statement.execute(sql);

System.out.println("XML data is pushed into the database");

} catch (SQLException e) {

System.err.println("Failed in Updating Data to OTP Table");

}

}

public static ResultSet getResultSet() throws Exception

{

ResultSet rs = null;

try

{

statement = connection.createStatement();

String sql = "select \* from login";

rs = statement.executeQuery(sql);

}

catch(SQLException e)

{

System.out.println(e.getMessage());

}

return rs;

}

public void close(){

try {

statement.close();

connection.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

private static Statement statement = null;

private static Connection connection=null;

}

**Schemas of DataBase Tables**

create table Login(

Customerid varchar(50) primary key not null,

CustomerTitle varchar(100),

CustomerFirstName varchar(100),

CustomerLastName varchar(100),

pWord varchar(100),

userType varchar(100),

emailID varchar(100)

);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

create table carDetails\_admin(

Carid varchar(50) primary key not null,

CarName varchar(100),

CarModel varchar(100),

Location varchar(100)

);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

create table paymentDetails(

payment\_id varchar(50) primary key not null,

CardNumber varchar(100),

CardType varchar(100),

CardName varchar(100),

CardExpiry varchar(100),

IssueBank varchar(100),

BillingAddress varchar(100)

);

create table mechanicServices\_admin(

Carid varchar(50),

foreign key (carid) references carDetails\_admin(Carid),

typeofService varchar(100),

serviceStatus varchar(100)

);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

create table Booking\_details(

booking\_id varchar(50) primary key not null,

CustomerTitle varchar(100),

CustomerFirstName varchar(100),

CustomerLastName varchar(100),

CustomerID varchar(100),

Location varchar(100),

BookingDate varchar(100),

StartDate varchar(100),

StartTime varchar(100),

EndDate varchar(100),

EndTime varchar(100),

BookingDuration varchar(100),

CarName varchar(100),

CarModel varchar(100),

CarColour varchar(100),

CarNumber varchar(100)

);

create table addCar(

CarId varchar(100),

CarName varchar(100),

CarModel varchar(100),

CarNumber varchar(100),

CarColour varchar(100),

SeatingCapacity varchar(100),

DriverName varchar(100),

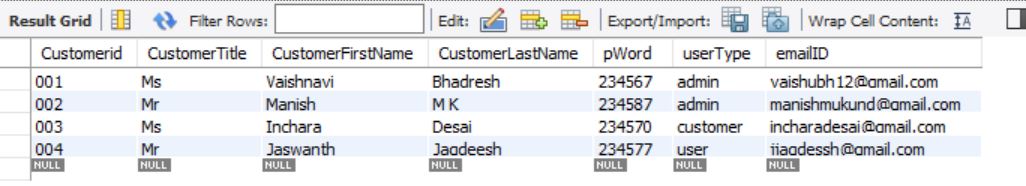
DriverId varchar(100),

foreign key(CarId) references carDetails\_admin(Carid)

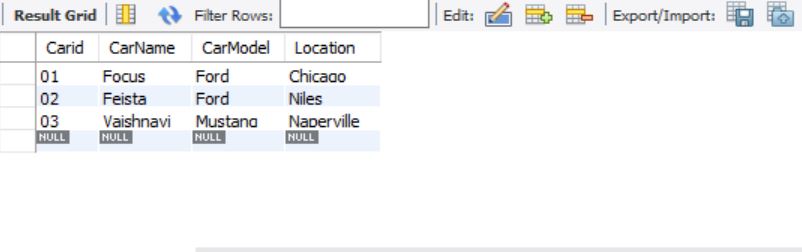
);

**Images of DataBase tables updated from XML Schema**

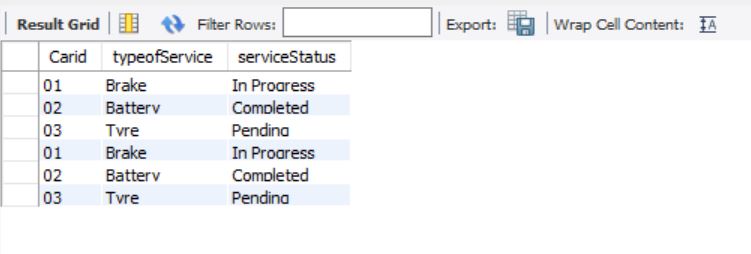
1. Login Table



1. Car Details Table



1. Mechnanic Services Table



1. Payment Details

