**package** com.dxc.daa.client;

**import** com.dxc.daa.dbcon.\*;

@SuppressWarnings("unused")

**public** **class** Main {

**public** **static** **void** main(String[] args) {

System.***out***.println("###Training Assessment APP###");

TrainApp app = **new** TrainApp();

app.launchTrainApp();

}

}

package com.dxc.daa.client;

import com.dxc.daa.dbcon.\*;

import com.dxc.daa.model.Train;

import java.util.ArrayList;

import java.util.Iterator;

import java.util.List;

import java.util.Scanner;

import com.dxc.daa.dao.\*;

@SuppressWarnings("unused")

public class TrainApp {

int choice=0;

@SuppressWarnings("resource")

public void launchTrainApp() {

// TODO Auto-generated method stub

UserDAO userDAO=new UserDAOImpl();

TrainDAO trainDAO = new TrainDAOImpl();

Scanner sc=new Scanner(System.in);

System.out.println("Username: ");

String username=sc.next();

System.out.println("Password: ");

String password=sc.next();

if(userDAO.validate(username,password)){

System.out.println("Access Granted");

while (true) {

System.out.println("M E N U ");

System.out.println("1. Display all training records : ");

System.out.println("2. Display Records one by one and update the percentage : ");

System.out.println("3. E X I T : ");

System.out.println("Please enter your choice : (1-3)");

choice = sc.nextInt();

switch (choice) {

case 1:

System.out.println(trainDAO.getAllRecords());

break;

case 2:

List<Train> record=new ArrayList<Train>();

record=trainDAO.getAllRecords();

Iterator<Train> iterator=record.iterator();

while(iterator.hasNext()) {

Train train=new Train();

train=iterator.next();

System.out.println(train.toString());

if(train.getPercentage()==0) {

System.out.println("Enter percentage: ");

int percentage=sc.nextInt();

trainDAO.updatePercentage(train.getSapId(),percentage);

}

else {

System.out.println("Percentage already exists");

}

}

break;

case 3:

System.out.println("Thanks for using my app");

System.exit(0);

break;

default:

System.out.println(" Please enter (1-3)");

}

}

}

else {

System.out.println("Invalid credentials");

System.exit(0);

}

}

}

package com.dxc.daa.dao;

import java.util.List;

import com.dxc.daa.model.Train;

public interface TrainDAO {

public List<Train> getAllRecords();

public void updatePercentage(int sapId, int percentage);

}

package com.dxc.daa.dao;

import com.dxc.daa.dbcon.\*;

import com.dxc.daa.model.Train;

import com.dxc.daa.model.User;

import java.sql.\*;

import java.util.ArrayList;

import java.util.List;

@SuppressWarnings("unused")

public class TrainDAOImpl implements TrainDAO{

Connection connection = (Connection) DBConnection.getConnection();

private static final String FETCH\_TRAIN\_RECORDS = "select \* from training";

public TrainDAOImpl(){

}

public List<Train> getAllRecords() {

// TODO Auto-generated method stub

List<Train> allRecords = new ArrayList<Train>();

try {

Statement stat = connection.createStatement();

ResultSet res = stat.executeQuery(FETCH\_TRAIN\_RECORDS);

while(res.next()) {

Train train=new Train();

train.setSapId(res.getInt(1));

train.setEmpName(res.getString(2));

train.setStream(res.getString(3));

train.setPercentage(res.getInt(4));

allRecords.add(train);

}

}catch (SQLException e) {

e.printStackTrace();

}

return allRecords;

}

public void updatePercentage(int sapId,int percentage) {

try {

PreparedStatement preparedStatement=connection.prepareStatement("update training set percentage=? where sapId=?");

preparedStatement.setInt(1,percentage);

preparedStatement.setInt(2,sapId);

preparedStatement.executeUpdate();

}catch (SQLException e) {

e.printStackTrace();

}

}

}

**package** com.dxc.daa.dao;

**import** com.dxc.daa.dbcon.\*;

@SuppressWarnings("unused")

**public** **interface** UserDAO {

**public** **boolean** validate(String username,String password);

}

package com.dxc.daa.dao;

import com.dxc.daa.dbcon.\*;

import com.dxc.daa.model.User;

import java.sql.\*;

@SuppressWarnings("unused")

public class UserDAOImpl implements UserDAO{

Connection connection = (Connection) DBConnection.getConnection();

private static final String FETCH\_USER\_DETAILS = "select \* from users where username=? AND password=?";

private static final String FETCH\_TRAIN\_RECORDS = "select \* from training";

public UserDAOImpl(){

}

@Override

public boolean validate(String username, String password) {

// TODO Auto-generated method stub

boolean exist=false;

try {

PreparedStatement statement=connection.prepareStatement(FETCH\_USER\_DETAILS);

statement.setString(1,username);

statement.setString(2,password);

ResultSet res=statement.executeQuery();

if(res.next()) {

exist=true;

}

else {

exist=false;

}

}

catch(SQLException e) {

e.printStackTrace();

}

return exist;

}

}

**package** com.dxc.daa.dbcon;

**import** java.sql.\*;

**public** **class** DBConnection {

**public** DBConnection() {

}

**public** **static** Connection getConnection() {

**try** {

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

} **catch** (ClassNotFoundException e) {

e.printStackTrace();

}

Connection connection=**null**;

**try** {

connection =DriverManager.*getConnection*("jdbc:mysql://localhost:3306/dxc", "root", "root");

} **catch** (SQLException e) {

e.printStackTrace();

}

**return** connection;

}

}

**package** com.dxc.daa.model;

**public** **class** Train {

**private** **int** SapId;

**private** String EmpName;

**private** String Stream;

**private** **int** percentage;

**public** Train() {

// **TODO** Auto-generated constructor stub

}

**public** Train(**int** sapId, String empName, String stream, **int** percentage) {

**super**();

SapId = sapId;

EmpName = empName;

Stream = stream;

**this**.percentage = percentage;

}

@Override

**public** String toString() {

**return** "Train [SapId=" + SapId + ", EmpName=" + EmpName +

", Stream=" + Stream + ", percentage=" + percentage+ "]";

}

@Override

**public** **int** hashCode() {

**final** **int** prime = 31;

**int** result = 1;

result = prime \* result + ((EmpName == **null**) ? 0 : EmpName.hashCode());

result = prime \* result + SapId;

result = prime \* result + ((Stream == **null**) ? 0 : Stream.hashCode());

result = prime \* result + percentage;

**return** result;

}

@Override

**public** **boolean** equals(Object obj) {

**if** (**this** == obj)

**return** **true**;

**if** (obj == **null**)

**return** **false**;

**if** (getClass() != obj.getClass())

**return** **false**;

Train other = (Train) obj;

**if** (EmpName == **null**) {

**if** (other.EmpName != **null**)

**return** **false**;

} **else** **if** (!EmpName.equals(other.EmpName))

**return** **false**;

**if** (SapId != other.SapId)

**return** **false**;

**if** (Stream == **null**) {

**if** (other.Stream != **null**)

**return** **false**;

} **else** **if** (!Stream.equals(other.Stream))

**return** **false**;

**if** (percentage != other.percentage)

**return** **false**;

**return** **true**;

}

**public** **int** getSapId() {

**return** SapId;

}

**public** **void** setSapId(**int** sapId) {

SapId = sapId;

}

**public** String getEmpName() {

**return** EmpName;

}

**public** **void** setEmpName(String empName) {

EmpName = empName;

}

**public** String getStream() {

**return** Stream;

}

**public** **void** setStream(String stream) {

Stream = stream;

}

**public** **int** getPercentage() {

**return** percentage;

}

**public** **void** setPercentage(**int** percentage) {

**this**.percentage = percentage;

}

}

**package** com.dxc.daa.model;

**import** com.dxc.daa.dbcon.\*;

@SuppressWarnings("unused")

**public** **class** User {

**private** String username;

**private** String password;

**public** User() {

// **TODO** Auto-generated constructor stub

}

**public** User(String username, String password) {

**super**();

**this**.username = username;

**this**.password = password;

}

@Override

**public** String toString() {

**return** "User [username=" + username + ", password=" + password + "]";

}

**public** String getUsername() {

**return** username;

}

**public** **void** setUsername(String username) {

**this**.username = username;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

@Override

**public** **int** hashCode() {

**final** **int** prime = 31;

**int** result = 1;

result = prime \* result + ((password == **null**) ? 0 : password.hashCode());

result = prime \* result + ((username == **null**) ? 0 : username.hashCode());

**return** result;

}

@Override

**public** **boolean** equals(Object obj) {

**if** (**this** == obj)

**return** **true**;

**if** (obj == **null**)

**return** **false**;

**if** (getClass() != obj.getClass())

**return** **false**;

User other = (User) obj;

**if** (password == **null**) {

**if** (other.password != **null**)

**return** **false**;

} **else** **if** (!password.equals(other.password))

**return** **false**;

**if** (username == **null**) {

**if** (other.username != **null**)

**return** **false**;

} **else** **if** (!username.equals(other.username))

**return** **false**;

**return** **true**;

}

}