COMSATS University Islamabad, Lahore Campus Department of Computer Science

**Project Smart City – Semester Spring 2020**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Course Title: | Object Oriented Programming-Lab | | | | | Course Code: | | CSC241 | Credit Hours: | 4(3,1) |
| Course Instructor/s: | Ms. Muntaha Iqbal | | | | | Programme Name: | | BCE | | |
| Semester: | 3rd | Batch: | FA19-BCE | | Session: | A | | | | |
| Total Marks: | **40** | Obtained Marks: | |  | | Date: | December 23, 2020 | | | |
| Student’s Name: | **Hassaan Saleem** | | | | | Reg. No. | FA19-BCE-024 | | | |
| **Important Instruction:**   * Student is himself/herself responsible for successful submission. * Submission deadline: January 02, 2021 at 11:59 pm | | | | | | | | | | |

**Name:**

Hassaan Saleem FA19-BCE-024

**PROJECT IDEA:**

**SMART CITY CLASS:**

The Smart City project idea is taken from the concept of new Colonies or cities application built to store all the some details of a city. In which due to time I only covered a part of it .

1. The first part of the code Is Based on Adding no of users In Student Block, Faculty Block, Employee Block.
2. The Second part is to print of Every Class
3. Last but not least, it is used to show Result of Student

**Submission Guideline:**

* You have to submit a pdf file (project idea 40-50 words, UML diagram, your code of all classes and screenshot of output) on cu-online.
* Upload you project folder in zip form on MS Teams.
* Submission deadline: January 02, 2021 at 11:59 pm
* Late submission will not be accepted

**AdminTest Class:**

package **com**;

import **com.ict\_abstracts\_parents\_interfaces.Address**; import **com.ict\_abstracts\_parents\_interfaces.Date**; import **com.ict\_education.faculty.ArtBlock**;

import **com.ict\_education.faculty.BBA\_Block**; import **com.ict\_education.faculty.FacultyBlock**; import **com.ict\_education.faculty.ITBlock**; import **com.ict\_education.students\_block.**\*; import **com.ict\_employees.**\*;

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

import **java.util.ArrayList**; import **java.util.Scanner**;

import static **com.ict\_employees.EmployeeStatus**.*PieceWorkers*; public class **AdminTest** {

public static void main( **String**[] args ) { **Scanner** integer = new Scanner(**System**.*in*); **Scanner** string = new Scanner(**System**.*in*);

**System**.*out*.print("Enter" +

"**\n\t**'1' for Entering Data" + "**\n\t**'2' for Printing Information" +

"**\n\t**'3' for Printing Student's result" + "**\n**: ");

int allInput = integer.nextInt();

if ( allInput == 1 ) {

for (int i = 0; i < 3; i++) {

**System**.*out*.print("**\n**Student Block**\n**Enter " + "**\n\t**'1' for Primary Section" + "**\n\t**'2' for Secondary Section" + "**\n\t**'3' for Higher Section" +

"**\n**: ");

*addStudent*(integer.nextInt());

}

**System**.*out*.println();

for (int i = 0; i < 3; i++) {

**System**.*out*.printf("**\n**Faculty Block**\n**" + "Enter" +

"**\n\t**'1' for Art Block" + "**\n\t**'2' for BBA\_Block" + "**\n\t**'3' for ITBlock" + "**\n**: ");

*addFaculty*(integer.nextInt());

}

for (int i = 0; i < 5; i++) {

**System**.*out*.printf("**\n**Employee Block**\n**" + "Enter" +

"**\n\t**'1' for Hourly Employee" + "**\n\t**'2' for Piece Workers " + "**\n\t**'3' for Salaried Employee" + "**\n\t**'4' for Commission Employee" +

"**\n\t**'5' for Base Plus Commission Employee" + "**\n**: ");

*addEmployees*(integer.nextInt());

}

} else if ( allInput == 2 ) { for (int i = 0; i < 5; i++) {

**System**.*out*.printf("**\n**Enter" + "**\n\t**'1' for Faculty Block" + "**\n\t**'2' for Student Block " + "**\n\t**'3' for Employee Block" + "**\n**: ");

*printInformation*(integer.nextInt());

}

} else if ( allInput == 3 ) {

**System**.*out*.print("Enter Department Name" + "**\n\t**'ARTS Block' for Arts Block" + "**\n\t**'BBA Block' for BBA Block" + "**\n\t**'IT Block' for IT Block" + "**\n**: ");

*studentResult*(string.nextLine());

}

}

public static void addStudent( int userInput ) {

**Connection** connection = null; **Statement** statement = null; int num1, num11;

**String** url = "jdbc:mysql://localhost/ict\_city";

**String** root = "root";

**String** password = "B796096b";

**StudentBlock**[] primaryBlock = new Primary[5]; **StudentBlock**[] middleBlock = new Middle[5]; **StudentBlock**[] higherBlock = new Higher[5];

if ( userInput == 1 ) {

**GPACalculation**[] gpaCalculations = new GPACalculation[primaryBlock.length]; primaryBlock[0] = new Primary(new Date(12, 7, 2000),

new Address("Lahore", "Punjab", "Pakistan"),

1, "Bilal Amjad", ["mee.mirzas@gmail.com"](mailto:mee.mirzas@gmail.com), 117000, 12000,

"PS5", 0);

gpaCalculations[0] = new GPACalculation(primaryBlock[0].getName(), primaryBlock[0].getId(), "ARTS Block"); gpaCalculations[0].calculateMarks(98, 98, 99, 90, 97);

primaryBlock[1] = new Primary(new Date(2, 7, 2001), new Address("Lahore", "Punjab", "Pakistan"),

2, "Hassaan Saleem", ["hassaansaleem@gmail.com"](mailto:hassaansaleem@gmail.com), 117020, 11000,

"X-BOX", 0);

gpaCalculations[1] = new GPACalculation(primaryBlock[1].getName(), primaryBlock[1].getId(), "ARTS Block"); gpaCalculations[1].calculateMarks(90, 30, 71, 74, 87);

primaryBlock[2] = new Primary(new Date(1, 7, 2010), new Address("Lahore", "Punjab", "Pakistan"),

3, "Fouzan Shakeel", ["fouzanshakeel@gmail.com"](mailto:fouzanshakeel@gmail.com), 117100, 13000,

"PS1", 1);

gpaCalculations[2] = new GPACalculation(primaryBlock[2].getName(), primaryBlock[2].getId(), "ARTS Block"); gpaCalculations[2].calculateMarks(18, 20, 31, 44, 97);

primaryBlock[3] = new Primary(new Date(12, 7, 200), new Address("Lahore", "Punjab", "Pakistan"),

4, "Sahal Abdullah", ["sahalabdullah@gmail.com"](mailto:sahalabdullah@gmail.com), 212222, 11020,

"PS2", 8);

gpaCalculations[3] = new GPACalculation(primaryBlock[3].getName(), primaryBlock[3].getId(), "ARTS Block"); gpaCalculations[3].calculateMarks(18, 20, 51, 96, 97);

primaryBlock[4] = new Primary(new Date(12, 7, 200), new Address("Lahore", "Punjab", "Pakistan"),

5, "Abdul Hadi", ["abdulhadi@gmail.com"](mailto:abdulhadi@gmail.com), 117120, 1000,

"PS12", 4);

gpaCalculations[4] = new GPACalculation(primaryBlock[4].getName(), primaryBlock[4].getId(), "ARTS Block"); gpaCalculations[4].calculateMarks(90, 91, 96, 94, 97);

for (int i = 0; i < gpaCalculations.length; i++) {

if ( gpaCalculations[i].percentage() >= 90 ) { gpaCalculations[i].setGrades('A'); gpaCalculations[i].setCGPA(4.0);

} else if ( gpaCalculations[i].percentage() < 90 && gpaCalculations[i].percentage() >= 80 ) { gpaCalculations[i].setGrades('B');

gpaCalculations[i].setCGPA(3.5);

} else if ( gpaCalculations[i].percentage() < 80 && gpaCalculations[i].percentage() >= 70 ) { gpaCalculations[i].setGrades('C');

gpaCalculations[i].setCGPA(3.0);

} else if ( gpaCalculations[i].percentage() < 70 && gpaCalculations[i].percentage() >= 60 ) { gpaCalculations[i].setGrades('D');

gpaCalculations[i].setCGPA(2.5);

} else if ( gpaCalculations[i].percentage() < 60 && gpaCalculations[i].percentage() >= 50 ) { gpaCalculations[i].setGrades('E');

gpaCalculations[i].setCGPA(2.0);

} else if ( gpaCalculations[i].percentage() < 50 && gpaCalculations[i].percentage() >= 0 ) { gpaCalculations[i].setGrades('F');

gpaCalculations[i].setCGPA(1.5);

}

}

try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement(); **System**.*out*.println("Primary Block: ");

int counter = 0;

for (**StudentBlock** primarySection : primaryBlock) {

num1 = statement.executeUpdate("" +

"INSERT INTO primaryBlock(id, name, email, fee, " + "pocketMoney, playingKit, absentDays, dateOfBirth, address) " +

"VALUES(" + primarySection.getId() + ", '" + primarySection.getName()

+ "', '" + primarySection.getEmail() + "'," + primarySection.calculateFee()

+ ", " + primarySection.getPocket\_money() + ", '" + primarySection.getPlaying\_kit()

+ "', " + primarySection.getAbsent\_days() + ", '" + primarySection.getDate()

+ "', '" + primarySection.getAddress() + "')");

num11 = statement.executeUpdate("INSERT INTO student\_result(id, name, department, marks, grades, CGPA)" + "VALUES(" + primarySection.getId() + ", '"

+ primarySection.getName() + "', '"

+ gpaCalculations[counter].getDepartment() + "', "

+ gpaCalculations[counter].getTotalMarks() + ", '"

+ gpaCalculations[counter].getGrades() + "', "

+ gpaCalculations[counter].getCGPA() + ")"); counter++;

}

**System**.*out*.println("Information Added !");

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( userInput == 2 ) {

**GPACalculation**[] gpaCalculations = new GPACalculation[middleBlock.length]; middleBlock[0] = new Middle(new Date(1, 2, 2000),

new Address("Narowal", "Punjab", "Pakistan"), primaryBlock.length + 1,

"Hammad Kurshid", ["hammadkurshid@gmail.com"](mailto:hammadkurshid@gmail.com), 117000, 12000,

"PS5", 0);

gpaCalculations[0] = new GPACalculation(middleBlock[0].getName(), middleBlock[0].getId(), "BBA Block");

gpaCalculations[0].calculateMarks(21, 32, 43, 54, 65);

middleBlock[1] = new Middle(new Date(2, 3, 2001),

new Address("Shakar Ghar", "Punjab", "Pakistan"), primaryBlock.length + 2,

"Hamza Shehraz", ["hamzashehraz@gmail.com"](mailto:hamzashehraz@gmail.com), 117020, 11000,

"X-BOX", 0);

gpaCalculations[1] = new GPACalculation(middleBlock[1].getName(), middleBlock[1].getId(), "BBA Block");

gpaCalculations[1].calculateMarks(31, 41, 51, 61, 71);

middleBlock[2] = new Middle(new Date(3, 4, 2000),

new Address("Raheem Yaar Khan", "KPK", "Pakistan"), primaryBlock.length + 3,

"Ahaz Ali", ["ahazali@gmail.com"](mailto:ahazali@gmail.com), 117100, 13000, "PS1", 1);

gpaCalculations[2] = new GPACalculation(middleBlock[2].getName(), middleBlock[2].getId(), "BBA Block");

gpaCalculations[2].calculateMarks(92, 88, 56, 68, 97);

middleBlock[3] = new Middle(new Date(4, 5, 2001),

new Address("Sadiq Abad", "Punjab", "Pakistan"), primaryBlock.length + 4,

"Osama Ali", ["osamaali@gmail.com"](mailto:osamaali@gmail.com), 212222, 11020,

"PS2", 8);

gpaCalculations[3] = new GPACalculation(middleBlock[3].getName(), middleBlock[3].getId(), "BBA Block");

gpaCalculations[3].calculateMarks(98, 98, 99, 90, 97);

middleBlock[4] = new Middle(new Date(5, 6, 2002), new Address("Lahore", "Punjab", "Pakistan"), primaryBlock.length + 5,

"Hafiz Osama", ["hafizosama@gmail.com"](mailto:hafizosama@gmail.com), 117120, 1000,

"PS12", 4);

gpaCalculations[4] = new GPACalculation(middleBlock[4].getName(), middleBlock[4].getId(), "BBA Block");

gpaCalculations[4].calculateMarks(98, 98, 99, 90, 97);

for (int i = 0; i < gpaCalculations.length; i++) { if ( gpaCalculations[i].percentage() >= 90 ) {

gpaCalculations[i].setGrades('A'); gpaCalculations[i].setCGPA(4.0);

} else if ( gpaCalculations[i].percentage() < 90

&& gpaCalculations[i].percentage() >= 80 ) { gpaCalculations[i].setGrades('B'); gpaCalculations[i].setCGPA(3.5);

} else if ( gpaCalculations[i].percentage() < 80

&& gpaCalculations[i].percentage() >= 70 ) { gpaCalculations[i].setGrades('C'); gpaCalculations[i].setCGPA(3.0);

} else if ( gpaCalculations[i].percentage() < 70

&& gpaCalculations[i].percentage() >= 60 ) { gpaCalculations[i].setGrades('D'); gpaCalculations[i].setCGPA(2.5);

} else if ( gpaCalculations[i].percentage() < 60

&& gpaCalculations[i].percentage() >= 50 ) { gpaCalculations[i].setGrades('E'); gpaCalculations[i].setCGPA(2.0);

} else if ( gpaCalculations[i].percentage() < 50

&& gpaCalculations[i].percentage() >= 0 ) { gpaCalculations[i].setGrades('F'); gpaCalculations[i].setCGPA(1.5);

}

}

try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement(); **System**.*out*.println("Middle Block: ");

int counter = 0;

for (**StudentBlock** middleSection : middleBlock) {

num1 = statement.executeUpdate("" +

"INSERT INTO middleBlock(id, name, email, fee, " + "pocketMoney, playingKit, absentDays, dateOfBirth, address) " +

"VALUES(" + middleSection.getId() + ", '" + middleSection.getName()

+ "', '" + middleSection.getEmail() + "'," + middleSection.calculateFee()

+ ", " + middleSection.getPocket\_money() + ", '" + middleSection.getPlaying\_kit()

+ "', " + middleSection.getAbsent\_days() + ", '" + middleSection.getDate()

+ "', '" + middleSection.getAddress() + "')");

num11 = statement.executeUpdate("" +

"INSERT INTO student\_result(id, name, " + "department, marks, grades, CGPA)" + "VALUES(" + middleSection.getId() + ", '"

+ middleSection.getName() + "', '"

+ gpaCalculations[counter].getDepartment() + "', "

+ gpaCalculations[counter].getTotalMarks() + ", '"

+ gpaCalculations[counter].getGrades() + "', "

+ gpaCalculations[counter].getCGPA() + ")"); counter++;

}

**System**.*out*.println("Information Added !");

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( userInput == 3 ) {

**GPACalculation**[] gpaCalculations = new GPACalculation[higherBlock.length]; higherBlock[0] = new Higher(new Date(1, 1, 2000),

new Address("Narowal", "Punjab", "Pakistan"), primaryBlock.length + middleBlock.length + 1, "Zeshan Ali", ["zeshanali@gmail.co](mailto:zeshanali@gmail.com)m", 117000, 12000,

"PS5", 0);

gpaCalculations[0] = new GPACalculation(higherBlock[0].getName(), higherBlock[0].getId(), "IT Block");

gpaCalculations[0].calculateMarks(91, 80, 26, 60, 76);

higherBlock[1] = new Higher(new Date(2, 2, 2002), new Address("Narowal", "Punjab", "Pakistan"), primaryBlock.length + middleBlock.length + 2,

"Hassan Zaheer", ["hassanzaheer@gmail.com"](mailto:hassanzaheer@gmail.com), 117020, 11000,

"X-BOX", 0);

gpaCalculations[1] = new GPACalculation(higherBlock[1].getName(), higherBlock[1].getId(), "IT Block");

gpaCalculations[1].calculateMarks(12, 56, 76, 67, 97);

higherBlock[2] = new Higher(new Date(3, 3, 2003), new Address("Multan", "Punjab", "Pakistan"), primaryBlock.length + middleBlock.length + 3,

"Muneeb ur Rehman", ["muneeb\_rehman@gmail.com"](mailto:muneeb_rehman@gmail.com), 117100, 13000,

"PS1", 1);

gpaCalculations[2] = new GPACalculation(higherBlock[2].getName(), higherBlock[2].getId(), "IT Block");

gpaCalculations[2].calculateMarks(12, 23, 34, 45, 56);

higherBlock[3] = new Higher(new Date(4, 4, 2004), new Address("Karachi", "Sindh", "Pakistan"), primaryBlock.length + middleBlock.length + 4,

"Sohail Arshad", ["sohailarshad@gmail.com"](mailto:sohailarshad@gmail.com), 212222, 11020,

"PS2", 8);

gpaCalculations[3] = new GPACalculation(higherBlock[3].getName(), higherBlock[3].getId(), "IT Block");

gpaCalculations[3].calculateMarks(13, 24, 35, 46, 57);

higherBlock[4] = new Higher(new Date(5, 5, 2005),

new Address("Haidar a bad", "Punjab", "Pakistan"), primaryBlock.length + middleBlock.length + 5, "Zain Basra", ["zainbasra@gmail.com"](mailto:zainbasra@gmail.com), 117120, 1000,

"PS12", 4);

gpaCalculations[4] = new GPACalculation(higherBlock[4].getName(), higherBlock[4].getId(), "IT Block");

gpaCalculations[4].calculateMarks(14, 25, 36, 58, 69); for (int i = 0; i < gpaCalculations.length; i++) {

if ( gpaCalculations[i].percentage() >= 90 ) { gpaCalculations[i].setGrades('A'); gpaCalculations[i].setCGPA(4.0);

} else if ( gpaCalculations[i].percentage() < 90

&& gpaCalculations[i].percentage() >= 80 ) { gpaCalculations[i].setGrades('B'); gpaCalculations[i].setCGPA(3.5);

} else if ( gpaCalculations[i].percentage() < 80

&& gpaCalculations[i].percentage() >= 70 ) { gpaCalculations[i].setGrades('C'); gpaCalculations[i].setCGPA(3.0);

} else if ( gpaCalculations[i].percentage() < 70

&& gpaCalculations[i].percentage() >= 60 ) { gpaCalculations[i].setGrades('D'); gpaCalculations[i].setCGPA(2.5);

} else if ( gpaCalculations[i].percentage() < 60

&& gpaCalculations[i].percentage() >= 50 ) { gpaCalculations[i].setGrades('E'); gpaCalculations[i].setCGPA(2.0);

} else if ( gpaCalculations[i].percentage() < 50

&& gpaCalculations[i].percentage() >= 0 ) { gpaCalculations[i].setGrades('F'); gpaCalculations[i].setCGPA(1.5);

}

}

try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement(); **System**.*out*.println("Higher Block: ");

int counter = 0;

for (**StudentBlock** higherSection : higherBlock) {

num1 = statement.executeUpdate("" +

"INSERT INTO higherBlock(id, name, email, fee, " + "pocketMoney, playingKit, absentDays, dateOfBirth, address) " +

"VALUES(" + higherSection.getId() + ", '" + higherSection.getName()

+ "', '" + higherSection.getEmail() + "'," + higherSection.calculateFee()

+ ", " + higherSection.getPocket\_money() + ", '" + higherSection.getPlaying\_kit()

+ "', " + higherSection.getAbsent\_days() + ", '" + higherSection.getDate()

+ "', '" + higherSection.getAddress() + "')");

num11 = statement.executeUpdate("" +

"INSERT INTO student\_result(id, name, " + "department, marks, grades, CGPA)" + "VALUES(" + higherSection.getId() + ", '"

+ higherSection.getName() + "', '"

+ gpaCalculations[counter].getDepartment() + "', "

+ gpaCalculations[counter].getTotalMarks() + ", '"

+ gpaCalculations[counter].getGrades() + "', "

+ gpaCalculations[counter].getCGPA() + ")"); counter++;

}

**System**.*out*.println("Information Added !");

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

}

}

public static void addFaculty( int input ) {

**Connection** connection = null; **Statement** statement = null; int num1;

**String** url = "jdbc:mysql://localhost/ict\_city";

**String** root = "root";

**String** password = "B796096b"; int num2;

**FacultyBlock**[] artsBlock = new ArtBlock[9]; **FacultyBlock**[] bbaBlocks = new BBA\_Block[9]; **FacultyBlock**[] itBlock = new ITBlock[9];

if ( input == 1 ) {

**System**.*out*.println("Arts Block Information"); artsBlock[0] = new ArtBlock(new Date(1, 1, 2021),

new Address("Islamabad", "Capital Territory", "Pakistan")

, 1, "Shafiq", ["shafiq786@gmail.com"](mailto:shafiq786@gmail.com), 12, 201200);

artsBlock[1] = new ArtBlock(new Date(2, 3, 2004),

new Address("London", "metropolitan county", "England")

, 2, "Ubaid Rajpoot", ["rana\_rajpoot112@Yahoo.com"](mailto:rana_rajpoot112@Yahoo.com), 22, 365000);

artsBlock[2] = new ArtBlock(new Date(5, 9, 2010), new Address("Khushab", "Punjab", "Pakistan")

, 3, "Shafqat Mehboob", ["shafqat999@outlook.com"](mailto:shafqat999@outlook.com), 66, 789000);

artsBlock[3] = new ArtBlock(new Date(3, 5, 2009),

new Address("Melbourne", "Victoria", "Australia")

, 4, "Malika Rania", ["ra\_malika@hotmail.com"](mailto:ra_malika@hotmail.com), 23, 129900);

artsBlock[4] = new ArtBlock(new Date(9, 12, 2005),

new Address("Rawalpindi", "Capital Territory", "Pakistan")

, 5, "Moazzam Ali Murtaza", ["murtazaali790@gmail.com"](mailto:murtazaali790@gmail.com), 77, 134900);

artsBlock[5] = new ArtBlock(new Date(7, 2, 2003),

new Address("Quetta", "Balochistan", "Pakistan")

, 6, "Sawera Rafi", ["rafii420@outmail.com"](mailto:rafii420@outmail.com), 24, 789000);

artsBlock[6] = new ArtBlock(new Date(4, 9, 2002),

new Address("Los Angeles", "California", "America")

, 7, "Bilal Ahmad", ["ahmad\_bilal786@gmail.com"](mailto:ahmad_bilal786@gmail.com), 87, 287000);

artsBlock[7] = new ArtBlock(new Date(9, 4, 2007), new Address("Multan", "Punjab", "Pakistan")

, 8, "Saim Iqbal", ["iqbal](mailto:iqbal-578@yahoo.com)-[578@yahoo.com"](mailto:iqbal-578@yahoo.com),

33, 207000);

artsBlock[8] = new ArtBlock(new Date(6, 9, 2006), new Address("Hyderabad", "Sindh", "Pakistan")

, 9, "Rana Arshad", ["rana](mailto:rana-tawana@outlook.com)-[tawana@outlook.com"](mailto:rana-tawana@outlook.com), 10, 213000);

try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

for (**FacultyBlock** artSection : artsBlock) { num1 = statement.executeUpdate("" +

"INSERT INTO art\_block " +

"(id, name, email, overTimeHours, " + "monthlySalary, dateOfBirth, address)" + "VALUES("

+ artSection.getId() + ", '"

+ artSection.getName() + "', '"

+ artSection.getEmail() + "', "

+ artSection.getOvertimeHours() + ", "

+ artSection.calculateSalary() + ", '"

+ artSection.getDate() + "', '"

+ artSection.getAddress() + "')");

}

**System**.*out*.println("Information Added!");

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( input == 2 ) {

**System**.*out*.println("BBA Block Information");

bbaBlocks[0] = new BBA\_Block(new Date(1, 1, 2021),

new Address("Islamabad", "Capital Territory", "Pakistan")

, artsBlock.length + 1, "Shafiq", ["shafiq786@gmail.com"](mailto:shafiq786@gmail.com), 12, 201200);

bbaBlocks[1] = new BBA\_Block(new Date(2, 3, 2004),

new Address("London", "metropolitan county", "England")

, artsBlock.length + 2, "Ubaid Rajpoot", ["rana\_rajpoot112@Yahoo.com"](mailto:rana_rajpoot112@Yahoo.com), 22, 365000);

bbaBlocks[2] = new BBA\_Block(new Date(5, 9, 2010), new Address("Khushab", "Punjab", "Pakistan")

, artsBlock.length + 3, "Shafqat Mehboob", ["shafqat999@outlook.com"](mailto:shafqat999@outlook.com), 66, 789000);

bbaBlocks[3] = new BBA\_Block(new Date(3, 5, 2009),

new Address("Melbourne", "Victoria", "Australia")

, artsBlock.length + 4, "Malika Rania", ["ra\_malika@hotmail.com"](mailto:ra_malika@hotmail.com), 23, 129900);

bbaBlocks[4] = new BBA\_Block(new Date(9, 12, 2005),

new Address("Rawalpindi", "Capital Territory", "Pakistan")

, artsBlock.length + 5, "Moazzam Ali Murtaza", ["murtazaali790@gmail.com"](mailto:murtazaali790@gmail.com), 77, 134900);

bbaBlocks[5] = new BBA\_Block(new Date(7, 2, 2003),

new Address("Quetta", "Balochistan", "Pakistan")

, artsBlock.length + 6, "Sawaira Rafi", ["rafii420@outmail.com"](mailto:rafii420@outmail.com), 24, 789000);

bbaBlocks[6] = new BBA\_Block(new Date(4, 9, 2002),

new Address("Los Angeles", "California", "America")

, artsBlock.length + 7, "Bilal Ahmad", ["ahmad\_bilal786@gmail.com"](mailto:ahmad_bilal786@gmail.com), 87, 287000);

bbaBlocks[7] = new BBA\_Block(new Date(9, 4, 2007), new Address("Multan", "Punjab", "Pakistan")

, artsBlock.length + 8, "Saim Iqbal", ["iqbal](mailto:iqbal-578@yahoo.com)-[578@yahoo.com"](mailto:iqbal-578@yahoo.com), 33, 207000);

bbaBlocks[8] = new BBA\_Block(new Date(6, 9, 2006), new Address("Hyderabad", "Sindh", "Pakistan")

, artsBlock.length + 9, "Rana Arshad", ["rana](mailto:rana-tawana@outlook.com)-[tawana@outlook.com"](mailto:rana-tawana@outlook.com), 10, 213000);

try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

for (**FacultyBlock** bbaSections : bbaBlocks) { num1 = statement.executeUpdate("" +

"INSERT INTO bba\_block " +

"(id, name, email, overTimeHours, " + "monthlySalary, dateOfBirth, address)" + "VALUES("

+ bbaSections.getId() + ", '"

+ bbaSections.getName() + "', '"

+ bbaSections.getEmail() + "', "

+ bbaSections.getOvertimeHours() + ", "

+ bbaSections.calculateSalary() + ", '"

+ bbaSections.getDate() + "', '"

+ bbaSections.getAddress() + "')");

}

**System**.*out*.println("Information Added!");

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( input == 3 ) {

**System**.*out*.println("IT Block Information"); itBlock[0] = new ITBlock(new Date(1, 1, 2021),

new Address("Islamabad", "Capital Territory", "Pakistan")

, artsBlock.length + bbaBlocks.length + 1, "Shafiq", ["shafiq786@gmail.com"](mailto:shafiq786@gmail.com), 12, 201200);

itBlock[1] = new ITBlock(new Date(2, 3, 2004),

new Address("London", "metropolitan county", "England")

, artsBlock.length + bbaBlocks.length + 2, "Ubaid Rajpoot", ["rana\_rajpoot112@Yahoo.com"](mailto:rana_rajpoot112@Yahoo.com), 22, 365000);

itBlock[2] = new ITBlock(new Date(5, 9, 2010),

new Address("Khushab", "Punjab", "Pakistan")

, artsBlock.length + bbaBlocks.length + 3, "Shafqat Mehboob", ["shafqat999@outlook.com"](mailto:shafqat999@outlook.com), 66, 789000);

itBlock[3] = new ITBlock(new Date(3, 5, 2009),

new Address("Melbourne", "Victoria", "Australia")

, artsBlock.length + bbaBlocks.length + 4, "Malika Rania", ["ra\_malika@hotmail.com"](mailto:ra_malika@hotmail.com), 23, 129900);

itBlock[4] = new ITBlock(new Date(9, 12, 2005),

new Address("Rawalpindi", "Capital Territory", "Pakistan")

, artsBlock.length + bbaBlocks.length + 5, "Moazzam Ali Murtaza", ["murtazaali790@gmail.com"](mailto:murtazaali790@gmail.com), 77, 134900);

itBlock[5] = new ITBlock(new Date(7, 2, 2003),

new Address("Quetta", "Balochistan", "Pakistan")

, artsBlock.length + bbaBlocks.length + 6, "Sawaira Rafi", ["rafii420@outmail.com"](mailto:rafii420@outmail.com), 24, 789000);

itBlock[6] = new ITBlock(new Date(4, 9, 2002),

new Address("Los Angeles", "California", "America")

, artsBlock.length + bbaBlocks.length + 7, "Bilal Ahmad", ["ahmad\_bilal786@gmail.com"](mailto:ahmad_bilal786@gmail.com), 87, 287000);

itBlock[7] = new ITBlock(new Date(9, 4, 2007),

new Address("Multan", "Punjab", "Pakistan")

, artsBlock.length + bbaBlocks.length + 8,

"Saim Iqbal", ["iqbal\_578@yahoo.com"](mailto:iqbal_578@yahoo.com), 33, 207000);

itBlock[8] = new ITBlock(new Date(6, 9, 2006),

new Address("Hyderabad", "Sindh", "Pakistan")

, artsBlock.length + bbaBlocks.length + 9, "Rana Arshad", ["rana\_tawana@outlook.com"](mailto:rana_tawana@outlook.com), 10, 213000);

try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

for (**FacultyBlock** itSections : itBlock) { num1 = statement.executeUpdate("" +

"INSERT INTO it\_block " +

"(id, name, email, overTimeHours, " + "monthlySalary, dateOfBirth, address)" + "VALUES("

+ itSections.getId() + ", '"

+ itSections.getName() + "', '"

+ itSections.getEmail() + "', "

+ itSections.getOvertimeHours() + ", "

+ itSections.calculateSalary() + ", '"

+ itSections.getDate() + "', '"

+ itSections.getAddress() + "')");

}

**System**.*out*.println("Information Added!");

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

}

}

public static void addEmployees( int userInput ) { **Scanner** integer = new Scanner(**System**.*in*); **ArrayList**<**Employee**> arrayList = new ArrayList<>(); **EmployeeStatus** employeeStatus = null;

**Connection** connection = null; **Statement** statement = null; int sum1, sum2;

**String** url = "jdbc:mysql://localhost/ict\_city";

**String** root = "root";

**String** password = "B796096b";

if ( userInput == 1 ) {

**System**.*out*.println("Hourly Employee"); arrayList.add(new HourlyEmployee(new Date(9, 8, 2004),

new Address("Calgary", "Alberta", "Canada"), "Mario", "Pussa", 1, 12000, 10));

arrayList.add(new HourlyEmployee(new Date(12, 12, 2012), new Address("Winnipeg", "Manitoba", "Canada"), "Soloman", "Richard", 2, 15600, 20));

arrayList.add(new HourlyEmployee(new Date(20, 11, 2000), new Address("Toronto", "Ontario", "Canada"), "Daniel", "Ryan", 3, 123000, 50));

arrayList.add(new HourlyEmployee(new Date(14, 12, 1980), new Address("Quebec City", "Quebec", "Canada"), "Danny", "Gates", 4, 67000, 98));

arrayList.add(new HourlyEmployee(new Date(6, 1, 1979),

new Address("Lunenberg", "Nova Scotia", "Canada"), "Hayashi", "Jhones", 5, 98000, 33));

try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

for (**Employee** arrayEmployees : arrayList) {

if ( arrayEmployees instanceof **HourlyEmployee** ) { sum1 = statement.executeUpdate("" +

"INSERT INTO hourly\_employee(SSN, firstName, lastName," + "dateOfBirth, address, wages, hours, earnings, employeeType)" + "VALUES("

+ arrayEmployees.getSocialSecurityNumber() + ", '"

+ arrayEmployees.getFirstName() + "', '"

+ arrayEmployees.getLastName() + "', '"

+ arrayEmployees.getDate() + "', '"

+ arrayEmployees.getAddress() + "', "

+ ((**HourlyEmployee**) arrayEmployees).getWages() + ", "

+ ((**HourlyEmployee**) arrayEmployees).getHours() + ", "

+ arrayEmployees.earnings() + ", '"

+ **EmployeeStatus**.*HourlyEmployee* + "')");

}

}

**System**.*out*.println("Information Added");

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( userInput == 2 ) { **System**.*out*.println("Piece Workers"); arrayList.add(new PieceWorkers(new Date(1, 3, 2003),

new Address("Sahiwal", "Punjab", "Pakistan"), "Zubaida", "Hashmi", 6, 76, 2));

arrayList.add(new PieceWorkers(new Date(9, 12, 1998), new Address("Sakhur", "Sindh", "Pakistan"), "Ahmed", "Meer", 7, 65, 3));

arrayList.add(new PieceWorkers(new Date(9, 12, 1998), new Address("Sakhur", "Sindh", "Pakistan"), "Zain", "Khan", 8, 64, 6));

arrayList.add(new PieceWorkers(new Date(4, 5, 1997), new Address("Quetta", "KPK", "Pakistan"), "Hamid", "Noor", 9, 12, 10));

arrayList.add(new PieceWorkers(new Date(10, 10, 1978),

new Address("Sheikhupura", " Punjab", " Pakistan"), "Naila", "Aamir", 10, 6, 42));

try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

for (**Employee** arrayEmployees : arrayList) {

if ( arrayEmployees instanceof **PieceWorkers** ) { sum1 = statement.executeUpdate("" +

"INSERT INTO piece\_workers(SSN, firstName, lastName, " + "dateOfBirth, address, noOfPieces, wages, earnings, employeeType)" + "VALUES("

+ arrayEmployees.getSocialSecurityNumber() + ", '"

+ arrayEmployees.getFirstName() + "', '"

+ arrayEmployees.getLastName() + "', '"

+ arrayEmployees.getDate() + "', '"

+ arrayEmployees.getAddress() + "', "

+ ((**PieceWorkers**) arrayEmployees).getNumberOfPieces() + ", "

+ ((**PieceWorkers**) arrayEmployees).getWages() + ", "

+ arrayEmployees.earnings() + ", '"

+ *PieceWorkers* + "')");

}

}

**System**.*out*.println("Information Added");

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( userInput == 3 ) {

**System**.*out*.println("Salaried Employee"); arrayList.add(new SalariedEmployee(new Date(22, 7, 2001),

new Address("Cuiaba", "Mato Grroso", "Brazil"), "Burhana", "lekimosto", 11, 1200));

arrayList.add(new SalariedEmployee(new Date(23, 5, 1975), new Address("Ouro Preto", "Minas Gerais", "Brazil"),

"Aleiko", "khastan", 12, 200));

arrayList.add(new SalariedEmployee(new Date(24, 9, 1895), new Address("Salvador", "State of Bahia", "Brazil"), "Amber", "Hertz", 13, 900));

arrayList.add(new SalariedEmployee(new Date(11, 4, 1780), new Address("Curituba", "Parana", "Brazil"), "Hayazi", "Brittes", 14, 4500));

arrayList.add(new SalariedEmployee(new Date(24, 8, 1988), new Address("Goiania", "Goias", "Brazil"), "Zavchi", "Zillia", 15, 7800));

try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

for (**Employee** arrayEmployees : arrayList) {

if ( arrayEmployees instanceof **SalariedEmployee** ) { sum1 = statement.executeUpdate("" +

"INSERT INTO salaried\_employee(SSN, firstName, lastName," +

" dateOfBirth, address, weeklySalary, earnings, employeeType)" + "VALUES("

+ arrayEmployees.getSocialSecurityNumber() + ", '"

+ arrayEmployees.getFirstName() + "', '"

+ arrayEmployees.getLastName() + "', '"

+ arrayEmployees.getDate() + "', '"

+ arrayEmployees.getAddress() + "', "

+ ((**SalariedEmployee**) arrayEmployees).getWeeklySalary() + ", "

+ arrayEmployees.earnings() + ", '"

+ **EmployeeStatus**.*SalariedEmployee* + "')");

}

}

**System**.*out*.println("Information Added");

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( userInput == 4 ) {

**System**.*out*.println("Commission Employee"); arrayList.add(new CommissionEmployee(new Date(27, 9, 1965),

new Address("New York", "North Eastern", "America"), "George", "Bush", 16, 23, 5000));

arrayList.add(new CommissionEmployee(new Date(25, 12, 1991), new Address("San Fransisco", "California", "America"), "Bill", "Beard", 17, 13, 23000));

arrayList.add(new CommissionEmployee(new Date(19, 2, 1995), new Address("Seattle", "Washington", "America"), "Jhon", "Simmons", 18, 18, 10000));

arrayList.add(new CommissionEmployee(new Date(29, 11, 1899), new Address("Detroit", "Michigan", "America"), "Amelia", "Clarke", 19, 20, 7600));

arrayList.add(new CommissionEmployee(new Date(30, 10, 1898), new Address("Las Vegas", "Nevada", "America"), "Aizart", "Dunk", 20, 50, 67000));

try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

for (**Employee** arrayEmployees : arrayList) {

if ( arrayEmployees instanceof **CommissionEmployee** ) { sum1 = statement.executeUpdate("" +

"INSERT INTO commission\_employee(SSN, firstName, lastName, dateOfBirth," + " address, commissionRate, grossSales, earnings, employeeType)" + "VALUES("

+ arrayEmployees.getSocialSecurityNumber() + ", '"

+ arrayEmployees.getFirstName() + "', '"

+ arrayEmployees.getLastName() + "', '"

+ arrayEmployees.getDate() + "', '"

+ arrayEmployees.getAddress() + "', "

+ ((**CommissionEmployee**) arrayEmployees).getCommissionRate() + ", "

+ ((**CommissionEmployee**) arrayEmployees).getGrossSales() + ", "

+ arrayEmployees.earnings() + ", '"

+ **EmployeeStatus**.*CommissionEmployee* + "')");

}

}

**System**.*out*.println("Information Added");

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( userInput == 5 ) {

**System**.*out*.println("BasePlusCommissionEmployee");

arrayList.add(new BasePlusCommissionEmployee(new Date(21, 12, 1970), new Address("Hiran Minar", "Punjab", "Pakistan"),

"Zain", "Saeed", 21, 2,

12000, 121000));

arrayList.add(new BasePlusCommissionEmployee(new Date(27, 9, 1976), new Address("Gujrat", "Punjab", "Pakistan"),

"Kashif", "Ali", 22, 12,

1400, 365000));

arrayList.add(new BasePlusCommissionEmployee(new Date(31, 1, 1981), new Address("Chakwal", "Punjab", "Pakistan"),

"Ali", "Ahmad", 23, 40,

15400, 679000));

arrayList.add(new BasePlusCommissionEmployee(new Date(20, 12, 1975), new Address("Sargodha", "Punjab", "Pakistan"),

"Ashar", "Muhammad", 24, 32,

16000, 45000));

arrayList.add(new BasePlusCommissionEmployee(new Date(18, 4, 1977), new Address("Kot Momin", "Punjab", "Pakistan"), "Shahzeib", "Aliyaar", 25, 12,

160, 3200));

try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

for (**Employee** arrayEmployees : arrayList) {

if ( arrayEmployees instanceof **BasePlusCommissionEmployee** ) { sum1 = statement.executeUpdate("" +

"INSERT INTO baseplus\_commission\_employee(SSN, firstName, lastName," +

" dateOfBirth, address, commissionRate, grossSales, basicSalary, earnings, employeeType)" + "VALUES("

+ arrayEmployees.getSocialSecurityNumber() + ", '"

+ arrayEmployees.getFirstName() + "', '"

+ arrayEmployees.getLastName() + "', '"

+ arrayEmployees.getDate() + "', '"

+ arrayEmployees.getAddress() + "', "

+ ((**BasePlusCommissionEmployee**) arrayEmployees).getCommissionRate() + ", "

+ ((**BasePlusCommissionEmployee**) arrayEmployees).getGrossSales() + ", "

+ ((**BasePlusCommissionEmployee**) arrayEmployees).getBasicSalary() + ", "

+ arrayEmployees.earnings() + ", '"

+ **EmployeeStatus**.*BasePlusCommissionEmployee* + "')");

}

}

**System**.*out*.println("Information Added");

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

}

}

public static void printInformation( int userInput ) { **Scanner** integer = new Scanner(**System**.*in*); **Connection** connection = null;

**Statement** statement = null;

**ResultSet** resultSet = null;

**String** url = "jdbc:mysql://localhost/ict\_city";

**String** root = "root";

**String** password = "B796096b";

switch (userInput) { case 1: {

**System**.*out*.println("You have Entered FACULTY BLOCK");

**System**.*out*.print("Enter " + "**\n\t**1 for Arts Block" + "**\n\t**2 for BBA Block" + "**\n\t**3 for IT Block**\n**: ");

int myInput = integer.nextInt(); if ( myInput == 1 ) {

**System**.*out*.println("|> Arts Block! <|**\n**"); try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

resultSet = statement.executeQuery("SELECT \* FROM art\_block"); while (resultSet.next()) {

**System**.*out*.println("||> ID: " + resultSet.getInt("id")); **System**.*out*.println("||> Name: " + resultSet.getString("name")); **System**.*out*.println("||> Overtime Hours: " + resultSet.getInt("overTimeHours"));

**System**.*out*.println("||> Monthly Salary: " + resultSet.getDouble("monthlySalary")); **System**.*out*.println("||> Date Of Birth: " + resultSet.getString("dateOfBirth")); **System**.*out*.println("||> Address: " + resultSet.getString("address")); **System**.*out*.println();

}

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( myInput == 2 ) { **System**.*out*.println("|> BBA Block <|**\n**"); try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

resultSet = statement.executeQuery("SELECT \* FROM bba\_block"); while (resultSet.next()) {

**System**.*out*.println("||> ID: " + resultSet.getInt("id")); **System**.*out*.println("||> Name: " + resultSet.getString("name")); **System**.*out*.println("||> Overtime Hours: " + resultSet.getInt("overTimeHours"));

**System**.*out*.println("||> Monthly Salary: " + resultSet.getDouble("monthlySalary")); **System**.*out*.println("||> Date Of Birth: " + resultSet.getString("dateOfBirth")); **System**.*out*.println("||> Address: " + resultSet.getString("address")); **System**.*out*.println();

}

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( myInput == 3 ) { **System**.*out*.println("|> IT Block <|**\n**"); try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

resultSet = statement.executeQuery("SELECT \* FROM it\_block"); while (resultSet.next()) {

**System**.*out*.println("||> ID: " + resultSet.getInt("id")); **System**.*out*.println("||> Name: " + resultSet.getString("name")); **System**.*out*.println("||> Overtime Hours: " + resultSet.getInt("overTimeHours"));

**System**.*out*.println("||> Monthly Salary: " + resultSet.getDouble("monthlySalary")); **System**.*out*.println("||> Date Of Birth: " + resultSet.getString("dateOfBirth")); **System**.*out*.println("||> Address: " + resultSet.getString("address")); **System**.*out*.println();

}

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

}

break;

}

case 2: {

**System**.*out*.println("You have Entered STUDENT BLOCK");

**System**.*out*.print("Enter " +

"**\n\t**1 for Primary Block" + "**\n\t**2 for Middle Block" + "**\n\t**3 for Higher Block**\n**: ");

int myInput = integer.nextInt(); if ( myInput == 1 ) {

**System**.*out*.println("|> PRIMARY BLOCK! <| **\n**"); try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

resultSet = statement.executeQuery("SELECT \* FROM primaryblock");

while (resultSet.next()) {

**System**.*out*.println("||> ID: " + resultSet.getInt("id")); **System**.*out*.println("||> Name: " + resultSet.getString("name")); **System**.*out*.println("||> E-mail: " + resultSet.getString("email")); **System**.*out*.println("||> Date Of Birth: " + resultSet.getString("dateOfBirth")); **System**.*out*.println("||> Address: " + resultSet.getString("address")); **System**.*out*.println("||> Playing Kit: " + resultSet.getString("playingKit")); **System**.*out*.println("||> Pocket Money: " + resultSet.getDouble("pocketMoney")); **System**.*out*.println("||> Absent Days: " + resultSet.getInt("absentDays")); **System**.*out*.println("||> Fee: " + resultSet.getDouble("fee")); **System**.*out*.println();

}

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( myInput == 2 ) { **System**.*out*.println("|> Middle BLOCK! <| **\n**"); try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

resultSet = statement.executeQuery("SELECT \* FROM middleblock");

while (resultSet.next()) {

**System**.*out*.println("||> ID: " + resultSet.getInt("id")); **System**.*out*.println("||> Name: " + resultSet.getString("name")); **System**.*out*.println("||> E-mail: " + resultSet.getString("email")); **System**.*out*.println("||> Date Of Birth: " + resultSet.getString("dateOfBirth")); **System**.*out*.println("||> Address: " + resultSet.getString("address")); **System**.*out*.println("||> Playing Kit: " + resultSet.getString("playingKit")); **System**.*out*.println("||> Pocket Money: " + resultSet.getDouble("pocketMoney")); **System**.*out*.println("||> Absent Days: " + resultSet.getInt("absentDays")); **System**.*out*.println("||> Fee: " + resultSet.getDouble("fee")); **System**.*out*.println();

}

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( myInput == 3 ) { **System**.*out*.println("|> Higher BLOCK! <| **\n**"); try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

resultSet = statement.executeQuery("SELECT \* FROM higherblock");

while (resultSet.next()) {

**System**.*out*.println("||> ID: " + resultSet.getInt("id")); **System**.*out*.println("||> Name: " + resultSet.getString("name")); **System**.*out*.println("||> E-mail: " + resultSet.getString("email"));

**System**.*out*.println("||> Date Of Birth: " + resultSet.getString("dateOfBirth")); **System**.*out*.println("||> Address: " + resultSet.getString("address")); **System**.*out*.println("||> Playing Kit: " + resultSet.getString("playingKit")); **System**.*out*.println("||> Pocket Money: " + resultSet.getDouble("pocketMoney")); **System**.*out*.println("||> Absent Days: " + resultSet.getInt("absentDays")); **System**.*out*.println("||> Fee: " + resultSet.getDouble("fee")); **System**.*out*.println();

}

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

}

break;

}

case 3: {

**System**.*out*.println("You have Entered EMPLOYEE BLOCK");

**System**.*out*.print("Enter " +

"**\n\t**'1' for Hourly Employee" + "**\n\t**'2' for Salaried Employee" + "**\n\t**'3' for Piece Workers" + "**\n\t**'4' for Commission Employee" +

"**\n\t**'5' for Base Plus Commission Employee" + "**\n**: ");

int myInput = integer.nextInt(); if ( myInput == 1 ) {

**System**.*out*.println("|> Hourly Employee <|**\n**"); try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

resultSet = statement.executeQuery("SELECT \* FROM hourly\_employee"); while (resultSet.next()) {

**System**.*out*.println("||> SSN: " + resultSet.getInt("SSN")); **System**.*out*.println("||> First Name: " + resultSet.getString("firstName")); **System**.*out*.println("||> Last Name: " + resultSet.getString("lastName")); **System**.*out*.println("||> Date of Birth: " + resultSet.getString("dateOfBirth")); **System**.*out*.println("||> Address: " + resultSet.getString("address")); **System**.*out*.println("||> Wages: " + resultSet.getDouble("wages")); **System**.*out*.println("||> Hours: " + resultSet.getDouble("hours")); **System**.*out*.println("||> Earnings: " + resultSet.getDouble("earnings")); **System**.*out*.println("||> Employee Type: " + resultSet.getString("employeeType")); **System**.*out*.println();

}

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( myInput == 2 ) {

**System**.*out*.println("|> Salaried Employee <| **\n**"); try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

resultSet = statement.executeQuery("SELECT \* FROM salaried\_employee"); while (resultSet.next()) {

**System**.*out*.println("||> SSN: " + resultSet.getInt("SSN")); **System**.*out*.println("||> First Name: " + resultSet.getString("firstName")); **System**.*out*.println("||> Last Name: " + resultSet.getString("lastName")); **System**.*out*.println("||> Date of Birth: " + resultSet.getString("dateOfBirth")); **System**.*out*.println("||> Address: " + resultSet.getString("address")); **System**.*out*.println("||> Weekly Salary: " + resultSet.getDouble("weeklySalary")); **System**.*out*.println("||> Earnings: " + resultSet.getDouble("earnings")); **System**.*out*.println("||> Employee Type: " + resultSet.getString("employeeType")); **System**.*out*.println();

}

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( myInput == 3 ) { **System**.*out*.println("|> Piece Workers <| **\n**"); try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

resultSet = statement.executeQuery("SELECT \* FROM piece\_workers"); while (resultSet.next()) {

**System**.*out*.println("||> SSN: " + resultSet.getInt("SSN")); **System**.*out*.println("||> First Name: " + resultSet.getString("firstName")); **System**.*out*.println("||> Last Name: " + resultSet.getString("lastName")); **System**.*out*.println("||> Date of Birth: " + resultSet.getString("dateOfBirth")); **System**.*out*.println("||> Address: " + resultSet.getString("address")); **System**.*out*.println("||> No of Pieces: " + resultSet.getDouble("noOfPieces")); **System**.*out*.println("||> Wages: " + resultSet.getDouble("wages")); **System**.*out*.println("||> Earnings: " + resultSet.getDouble("earnings")); **System**.*out*.println("||> Employee Type: " + resultSet.getString("employeeType")); **System**.*out*.println();

}

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( myInput == 4 ) {

**System**.*out*.println("|> Commission Employee <|**\n** "); try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

resultSet = statement.executeQuery("SELECT \* FROM commission\_employee"); while (resultSet.next()) {

**System**.*out*.println("||> SSN: " + resultSet.getInt("SSN")); **System**.*out*.println("||> First Name: " + resultSet.getString("firstName")); **System**.*out*.println("||> Last Name: " + resultSet.getString("lastName")); **System**.*out*.println("||> Date of Birth: " + resultSet.getString("dateOfBirth")); **System**.*out*.println("||> Address: " + resultSet.getString("address"));

**System**.*out*.println("||> Commission Rate: " + resultSet.getDouble("commissionRate")); **System**.*out*.println("||> Gross Sales: " + resultSet.getDouble("grossSales")); **System**.*out*.println("||> Earnings: " + resultSet.getDouble("earnings")); **System**.*out*.println("||> Employee Type: " + resultSet.getString("employeeType")); **System**.*out*.println();

}

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

} else if ( myInput == 5 ) {

**System**.*out*.println("|> Base Plus Commission Employee <| **\n**"); try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

resultSet = statement.executeQuery("SELECT \* FROM baseplus\_commission\_employee"); while (resultSet.next()) {

**System**.*out*.println("||> SSN: " + resultSet.getInt("SSN")); **System**.*out*.println("||> First Name: " + resultSet.getString("firstName")); **System**.*out*.println("||> Last Name: " + resultSet.getString("lastName")); **System**.*out*.println("||> Date of Birth: " + resultSet.getString("dateOfBirth")); **System**.*out*.println("||> Address: " + resultSet.getString("address"));

**System**.*out*.println("||> Commission Rate: " + resultSet.getDouble("commissionRate")); **System**.*out*.println("||> Gross Sales: " + resultSet.getDouble("grossSales")); **System**.*out*.println("||> Basic Salary: " + resultSet.getDouble("basicSalary")); **System**.*out*.println("||> Earnings: " + resultSet.getDouble("earnings")); **System**.*out*.println("||> Employee Type: " + resultSet.getString("employeeType")); **System**.*out*.println();

}

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

}

break;

}

}

}

public static void studentResult( **String** userInput ) {

**Connection** connection = null; **Statement** statement = null; **ResultSet** resultSet = null;

**String** url = "jdbc:mysql://localhost/ict\_city";

**String** root = "root";

**String** password = "B796096b"; try {

connection = **DriverManager**.*getConnection*(url, root, password); statement = connection.createStatement();

resultSet = statement.executeQuery("SELECT \* FROM student\_result");

**System**.*out*.println("| ID **\t**| Name**\t\t\t** | Department**\t\t\t** | Marks **\t** | Grades |CGPA**\t** |"); while (resultSet.next()) {

if ( resultSet.getString("department").equals(userInput) ) {

if ( resultSet.getInt("id") == 1 || resultSet.getInt("id") == 5 )

**System**.*out*.println(" " + resultSet.getInt("id") + " **\t** "

+ resultSet.getString("name") + " **\t\t** "

+ resultSet.getString("department") + " **\t \t\t**"

+ resultSet.getDouble("marks") + "**\t\t**"

+ resultSet.getString("grades") + " **\t\t** "

+ resultSet.getDouble("CGPA"));

else if ( resultSet.getInt("id") > 1 && resultSet.getInt("id") < 5 )

**System**.*out*.println(" " + resultSet.getInt("id") + " **\t** "

+ resultSet.getString("name") + " **\t** "

+ resultSet.getString("department") + " **\t \t\t**"

+ resultSet.getDouble("marks") + "**\t\t**"

+ resultSet.getString("grades") + " **\t\t** "

+ resultSet.getDouble("CGPA"));

else if ( resultSet.getInt("id") == 6 || resultSet.getInt("id") == 7 )

**System**.*out*.println(" " + resultSet.getInt("id") + " **\t** "

+ resultSet.getString("name") + " **\t** "

+ resultSet.getString("department") + " **\t \t\t**"

+ resultSet.getDouble("marks") + "**\t\t**"

+ resultSet.getString("grades") + " **\t\t** "

+ resultSet.getDouble("CGPA"));

else if ( resultSet.getInt("id") > 7 && resultSet.getInt("id") < 11 )

**System**.*out*.println(" " + resultSet.getInt("id") + " **\t** "

+ resultSet.getString("name") + " **\t\t** "

+ resultSet.getString("department") + " **\t \t\t**"

+ resultSet.getDouble("marks") + "**\t\t**"

+ resultSet.getString("grades") + " **\t\t** "

+ resultSet.getDouble("CGPA"));

else if ( resultSet.getInt("id") == 11 || resultSet.getInt("id") == 15 )

**System**.*out*.println(" " + resultSet.getInt("id") + " **\t** "

+ resultSet.getString("name") + " **\t\t** "

+ resultSet.getString("department") + " **\t \t\t\t**"

+ resultSet.getDouble("marks") + "**\t\t**"

+ resultSet.getString("grades") + " **\t\t** "

+ resultSet.getDouble("CGPA"));

else if ( resultSet.getInt("id") == 12 || resultSet.getInt("id") == 14 )

**System**.*out*.println(" " + resultSet.getInt("id") + " **\t** "

+ resultSet.getString("name") + " **\t** "

+ resultSet.getString("department") + " **\t \t\t\t**"

+ resultSet.getDouble("marks") + "**\t\t**"

+ resultSet.getString("grades") + " **\t\t** "

+ resultSet.getDouble("CGPA")); else if ( resultSet.getInt("id") == 13 )

**System**.*out*.println(" " + resultSet.getInt("id") + " **\t** "

+ resultSet.getString("name") + " "

+ resultSet.getString("department") + " **\t \t\t\t**"

+ resultSet.getDouble("marks") + "**\t\t**"

+ resultSet.getString("grades") + " **\t\t** "

+ resultSet.getDouble("CGPA"));

}

}

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

} finally { try {

connection.close(); statement.close();

} catch (**SQLException** sqlException) { sqlException.printStackTrace();

}

}

}

}

**Administration Class:**

package **com.ict\_abstracts\_parents\_interfaces**;

public abstract class **Administration** { private **Date** date;

private **Address** address; private int id;

private **String** name; private **String** email;

public Administration( **Date** date, **Address** address ) { this.date = date;

this.address = address;

}

public Administration( **Date** date, **Address** address, int id, **String** name,

**String** email ) {

setId(id); setName(name); setEmail(email); setAddress(address); setDate(date);

}

public **Date** getDate() { return date;

}

public void setDate( **Date** date ) { this.date = date;

}

public **Address** getAddress() { return address;

}

public void setAddress( **Address** address ) { this.address = address;

}

public int getId() { return id;

}

public void setId( int id ) { this.id = id;

}

public **String** getName() { return name;

}

public void setName( **String** name ) { this.name = name;

}

public **String** getEmail() { return email;

}

public void setEmail( **String** email ) { this.email = email;

}

}

**Address class:**

package **com.ict\_abstracts\_parents\_interfaces**;

public class **Address** { private **String** city; private **String** province; private **String** country;

public Address( **String** city, **String** province, **String** country ) { setCity(city);

setProvince(province); setCountry(country);

}

public **String** getCity() { return city;

}

public void setCity( **String** city ) { this.city = city;

}

public **String** getProvince() { return province;

}

public void setProvince( **String** province ) { this.province = province;

}

public **String** getCountry() { return country;

}

public void setCountry( **String** country ) { this.country = country;

}

**@Override**

public **String** toString() {

return " Address: " + getCity() + ", " + getProvince() + ", " + getCountry();

}

}

**Date Class:**

package **com.ict\_abstracts\_parents\_interfaces**;

public class **Date** { private int date; private int month; private int year;

public Date( int date, int month, int year ) { setYear(year);

setMonth(month); setDate(date);

}

public int getDate() { return date;

}

public void setDate( int date ) { this.date = date;

}

public int getMonth() { return month;

}

public void setMonth( int month ) { this.month = month;

}

public int getYear() { return year;

}

public void setYear( int year ) { this.year = year;

}

**@Override**

public **String** toString() {

return "Date: " + getDate() + ", " + getMonth() + ", " + getYear();

}

}

**Faculty class:**

package **com.ict\_education.faculty**;

import **com.ict\_abstracts\_parents\_interfaces.Address**; import **com.ict\_abstracts\_parents\_interfaces.Calculation**; import **com.ict\_abstracts\_parents\_interfaces.Date**;

import **com.ict\_education.EducationalBlock**;

public class **FacultyBlock** extends **EducationalBlock** implements **Calculation** { private int overtimeHours;

private double monthlySalary;

public FacultyBlock( **Date** date, **Address** address, int id,

**String** name, **String** email,

int overtimeHours, double monthlySalary ) { super(date, address, id, name, email); setOvertimeHours(overtimeHours); setMonthlySalary(monthlySalary);

}

public int getOvertimeHours() { return overtimeHours;

}

public void setOvertimeHours( int overtimeHours ) { this.overtimeHours = overtimeHours;

}

public double getMonthlySalary() { return monthlySalary;

}

public void setMonthlySalary( double monthlySalary ) { this.monthlySalary = monthlySalary;

}

**@Override**

public double calculateSalary() { if ( getOvertimeHours() == 0 )

return getMonthlySalary(); else if ( getOvertimeHours() > 0 )

return (1200 \* getOvertimeHours()) + getMonthlySalary(); return calculateSalary();

}

}

**ARTS Block Code:**

package **com.ict\_education.faculty**;

import **com.ict\_abstracts\_parents\_interfaces.Address**; import **com.ict\_abstracts\_parents\_interfaces.Date**;

public class **ArtBlock** extends **FacultyBlock** {

public ArtBlock( **Date** date, **Address** address, int id,

**String** name, **String** email, int overtimeHours, double monthlySalary ) {

super(date, address, id, name, email, overtimeHours, monthlySalary);

}

**@Override**

public double calculateSalary() { return super.calculateSalary();

}

}

**BBA\_Block**:

package **com.ict\_education.faculty**;

import **com.ict\_abstracts\_parents\_interfaces.Address**; import **com.ict\_abstracts\_parents\_interfaces.Date**;

public class **BBA\_Block** extends **FacultyBlock** {

public BBA\_Block( **Date** date, **Address** address,

int id, **String** name, **String** email,

int overtimeHours, double monthlySalary ) { super(date, address, id, name, email, overtimeHours, monthlySalary);

}

**@Override**

public double calculateSalary() { return super.calculateSalary();

}

}

**IT Block Class:**

package **com.ict\_education.faculty**;

import **com.ict\_abstracts\_parents\_interfaces.Address**; import **com.ict\_abstracts\_parents\_interfaces.Date**;

public class **ITBlock** extends **FacultyBlock** {

public ITBlock( **Date** date, **Address** address, int id,

**String** name, **String** email, int overtimeHours, double monthlySalary ) {

super(date, address, id, name, email, overtimeHours, monthlySalary);

}

**@Override**

public double calculateSalary() { return super.calculateSalary();

}

}

**Student class:**

package **com.ict\_education.students\_block**;

import **com.ict\_abstracts\_parents\_interfaces.Address**; import **com.ict\_abstracts\_parents\_interfaces.Date**; import **com.ict\_education.EducationalBlock**;

public class **StudentBlock** extends **EducationalBlock** {

//////// we have to insert fines yet /////////////// private double fee;

private double pocket\_money; private **String** playing\_kit; private int absent\_days;

public StudentBlock( **Date** date, **Address** address, int id,

**String** name, **String** email, double fee, double pocket\_money, **String** playing\_kit, int absent\_days ) {

super(date, address, id, name, email); setAbsent\_days(absent\_days); setPlaying\_kit(playing\_kit); setPocket\_money(pocket\_money); setFee(fee);

}

public StudentBlock( **Date** date, **Address** address,

int id, **String** name, **String** email ) { super(date, address, id, name, email);

}

public double getFee() { return fee;

}

public void setFee( double fee ) { this.fee = fee;

}

public double getPocket\_money() { return pocket\_money;

}

public void setPocket\_money( double pocket\_money ) { this.pocket\_money = pocket\_money;

}

public **String** getPlaying\_kit() { return playing\_kit;

}

public void setPlaying\_kit( **String** playing\_kit ) { this.playing\_kit = playing\_kit;

}

public int getAbsent\_days() { return absent\_days;

}

public void setAbsent\_days( int absent\_days ) { this.absent\_days = absent\_days;

}

public double calculateFee() {

if ( getAbsent\_days() == 0 ) { return getFee();

} else if ( getAbsent\_days() > 0 ) { double instantFine = 100;

return getFee() + (getAbsent\_days() \* instantFine);

}

return calculateFee();

}

}

**Primary Class:**

package **com.ict\_education.students\_block**;

import **com.ict\_abstracts\_parents\_interfaces.Address**; import **com.ict\_abstracts\_parents\_interfaces.Date**;

public class **Primary** extends **StudentBlock**{

public Primary( **Date** date, **Address** address, int id,

**String** name, **String** email, double fee, double pocket\_money, **String** playing\_kit, int absent\_days ) {

super(date, address, id, name, email, fee, pocket\_money, playing\_kit, absent\_days);

}

}

**Middle class:**

package **com.ict\_education.students\_block**;

import **com.ict\_abstracts\_parents\_interfaces.Address**; import **com.ict\_abstracts\_parents\_interfaces.Date**;

public class **Middle** extends **StudentBlock** {

public Middle( **Date** date, **Address** address, int id, **String** name, **String** email, double fee, double pocket\_money, **String** playing\_kit, int absent\_days ) {

super(date, address, id, name, email, fee, pocket\_money, playing\_kit, absent\_days);

}

}

**Higher class:**

package **com.ict\_education.students\_block**;

import **com.ict\_abstracts\_parents\_interfaces.Address**; import **com.ict\_abstracts\_parents\_interfaces.Date**;

public class **Higher** extends **StudentBlock** {

public Higher( **Date** date, **Address** address, int id, **String** name, **String** email, double fee, double pocket\_money, **String** playing\_kit, int absent\_days ) {

super(date, address, id, name, email, fee, pocket\_money, playing\_kit, absent\_days);

}

}

**GPACalculation class:**

package **com.ict\_education.students\_block**;

import **com.ict\_abstracts\_parents\_interfaces.Calculation**; public class **GPACalculation** implements **Calculation** {

private **String** name; private int id;

private **String** department; private double totalMarks; private double CGPA;

private **Character** grades = 'B'; private int noOfSubjects;

public GPACalculation( **String** name, int id, **String** department ) { setDepartment(department);

setName(name); setId(id);

}

## public int getNoOfSubjects() { return noOfSubjects;

}

## public void setNoOfSubjects( int noOfSubjects ) { this.noOfSubjects = noOfSubjects;

}

## public **String** getName() { return name;

}

## public void setName( **String** name ) { this.name = name;

}

## public int getId() { return id;

}

## public void setId( int id ) { this.id = id;

}

## public **String** getDepartment() { return department;

}

## public void setDepartment( **String** department ) { this.department = department;

}

## public double getTotalMarks() { return totalMarks;

}

## public void setTotalMarks( double totalMarks ) { this.totalMarks = totalMarks;

}

## public double getCGPA() { return CGPA;

}

## public void setCGPA( double CGPA ) { this.CGPA = CGPA;

}

public **Character** getGrades() { return grades;

}

## public void setGrades( **Character** grades ) { this.grades = grades;

}

# @Override

## public double calculateMarks( double... marks ) { double sum = 0; setNoOfSubjects(marks.length);

for (int i = 0; i < marks.length; i++) { sum += marks[i];

}

setTotalMarks(sum); return getTotalMarks();

}

public double percentage() { double percent = 0;

percent = (getTotalMarks() \* 100) / (getNoOfSubjects() \* 100); return percent;

}

}

**EDUCATIONAL Block class**

package **com.ict\_education**;

import **com.ict\_abstracts\_parents\_interfaces.Address**;

import **com.ict\_abstracts\_parents\_interfaces.Administration**; import **com.ict\_abstracts\_parents\_interfaces.Date**;

public class **EducationalBlock** extends **Administration** { public EducationalBlock( **Date** date, **Address** address,

int id, **String** name, **String** email ) { super(date, address, id, name, email);

}

}

**Employee class:**

package **com.ict\_employees**;

import **com.ict\_abstracts\_parents\_interfaces.Address**;

import **com.ict\_abstracts\_parents\_interfaces.Administration**; import **com.ict\_abstracts\_parents\_interfaces.Calculation**; import **com.ict\_abstracts\_parents\_interfaces.Date**;

public abstract class **Employee** extends **Administration** implements **Calculation** { private **String** firstName;

private **String** lastName;

private int socialSecurityNumber;

public Employee( **Date** date, **Address** address,

**String** firstName, **String** lastName, int socialSecurityNumber ) {

super(date, address); this.firstName = firstName; this.lastName = lastName;

this.socialSecurityNumber = socialSecurityNumber;

}

public **String** getFirstName() { return firstName;

}

public void setFirstName( **String** firstName ) { this.firstName = firstName;

}

public **String** getLastName() { return lastName;

}

public void setLastName( **String** lastName ) { this.lastName = lastName;

}

public int getSocialSecurityNumber() { return socialSecurityNumber;

}

public void setSocialSecurityNumber( int socialSecurityNumber ) { this.socialSecurityNumber = socialSecurityNumber;

}

**@Override**

public double earnings() { return 0;

}

**@Override**

public **String** toString() { return " Employee{ " +

"firstName: '" + getFirstName() + '**\'**' + ", lastName: '" + getLastName() + '**\'**' +

", socialSecurityNumber: " + getSocialSecurityNumber() + " }";

}

}

**Base Plus Commission Employee Class:**

package **com.ict\_employees**;

import **com.ict\_abstracts\_parents\_interfaces.Address**; import **com.ict\_abstracts\_parents\_interfaces.Date**;

public class **BasePlusCommissionEmployee** extends **CommissionEmployee** { private double basicSalary;

public BasePlusCommissionEmployee( **Date** date,

**Address** address,

**String** firstName,

**String** lastName, int socialSecurityNumber, double commissionRate, double grossSales, double

basicSalary ) {

super(date, address, firstName, lastName, socialSecurityNumber, commissionRate, grossSales);

this.basicSalary = basicSalary;

}

public double getBasicSalary() { return basicSalary;

}

public void setBasicSalary(double basicSalary) { this.basicSalary = basicSalary;

}

//// Methods

**@Override**

public double earnings() {

return getBasicSalary() + (getCommissionRate() \* getGrossSales());

}

**@Override**

public **String** toString() {

return "" + super.toString() + "**\n\t** BasePlusCommissionEmployee{ " + "basicSalary: " + getBasicSalary() +

" }";

}

}

**Commission Employee Class:**

package **com.ict\_employees**;

import **com.ict\_abstracts\_parents\_interfaces.Address**; import **com.ict\_abstracts\_parents\_interfaces.Date**;

public class **CommissionEmployee** extends **Employee** { private double commissionRate;

private double grossSales;

public CommissionEmployee( **Date** date, **Address** address,

**String** firstName, **String** lastName, int socialSecurityNumber,

double commissionRate, double grossSales ) { super(date, address, firstName, lastName, socialSecurityNumber); this.commissionRate = commissionRate;

this.grossSales = grossSales;

}

public double getCommissionRate() { return commissionRate;

}

public void setCommissionRate(double commissionRate) { this.commissionRate = commissionRate;

}

public double getGrossSales() { return grossSales;

}

public void setGrossSales(double grossSales) { this.grossSales = grossSales;

}

////Methods

**@Override**

public double earnings() {

return getCommissionRate() \* getGrossSales();

}

**@Override**

public **String** toString() {

return "Commission Employee Information: **\n\t**" + super.toString() + " CommissionEmployee{ " +

"commissionRate: " + getCommissionRate() + ", grossSales: " + getGrossSales() +

" }";

}

}

**HourlyEmployee Class:**

package **com.ict\_employees**;

import **com.ict\_abstracts\_parents\_interfaces.Address**; import **com.ict\_abstracts\_parents\_interfaces.Date**;

public class **HourlyEmployee** extends **Employee** { private double wages;

private double hours;

public HourlyEmployee( **Date** date, **Address** address,

**String** firstName, **String** lastName, int socialSecurityNumber,

double wages, double hours ) {

super(date, address, firstName, lastName, socialSecurityNumber); this.wages = wages;

this.hours = hours;

}

public double getWages() { return wages;

}

public void setWages( double wages ) { this.wages = wages;

}

public double getHours() { return hours;

}

public void setHours(double hours) { this.hours = hours;

}

//methods

**@Override**

public double earnings() { double salary = 0;

if ( hours <= 40 ) {

salary = getWages() \* getHours();

} else if ( hours > 40 ) {

salary = 40 \* getWages() + (getHours() - 40) \* getWages() \* 1.5;

}

return salary;

}

**@Override**

public **String** toString() {

return "Hourly Employee Information: **\n\t**" + super.toString() + " HourlyEmployee{ " + "wage: " + getWages() +

", hours: " + getHours() + " }";

}

}

**Salaried Employee:**

package **com.ict\_employees**;

import **com.ict\_abstracts\_parents\_interfaces.Address**; import **com.ict\_abstracts\_parents\_interfaces.Date**;

public class **SalariedEmployee** extends **Employee** {

private double weeklySalary;

public SalariedEmployee( **Date** date, **Address** address,

**String** firstName, **String** lastName, int socialSecurityNumber,

double weeklySalary ) {

super(date, address, firstName, lastName, socialSecurityNumber); this.weeklySalary = weeklySalary;

}

public double getWeeklySalary() { return weeklySalary;

}

public void setWeeklySalary(double weeklySalary) { this.weeklySalary = weeklySalary;

}

//// Methods

**@Override**

public double earnings() { return getWeeklySalary();

}

**@Override**

public **String** toString() {

return "Salaried Employee Information: **\n\t**" + super.toString()

+ " SalariedEmployee{ weeklySalary: " + getWeeklySalary() + " }";

}

}

**Piece Workers Class**

package **com.ict\_employees**;

import **com.ict\_abstracts\_parents\_interfaces.Address**; import **com.ict\_abstracts\_parents\_interfaces.Date**;

public class **PieceWorkers** extends **Employee** { private double numberOfPieces;

private double wages;

public PieceWorkers( **Date** date, **Address** address, **String** firstName,

**String** lastName, int socialSecurityNumber, double numberOfPieces, double wages ) {

super(date, address, firstName, lastName, socialSecurityNumber); this.numberOfPieces = numberOfPieces;

this.wages = wages;

}

public double getNumberOfPieces() { return numberOfPieces;

}

public void setNumberOfPieces( double numberOfPieces ) { this.numberOfPieces = numberOfPieces;

}

public double getWages() { return wages;

}

public void setWages( double wages ) { this.wages = wages;

}

public double earnings() {

return numberOfPieces \* wages;

}

**@Override**

public **String** toString() {

return "Piece Work Information: **\n\t**"+super.toString() + " PieceWorkers{" + "numberOfPieces: " + numberOfPieces +

", wage: " + wages + '}';

}

}

**Employee Status Class:**

package **com.ict\_employees**;

public enum **EmployeeStatus** {

*CommissionEmployee*, *BasePlusCommissionEmployee*, *HourlyEmployee*, *SalariedEmployee*, *PieceWorkers*;

}





