**Important Instructions:**

1. **Please read the document thoroughly before you code.**
2. **Import the given skeleton code into your Eclipse.**
3. **Use Java 8 for solving the coding challenge.**
4. **You have to test the code and ensure there are no compilation errors before submission**
5. **Business Scenario 1:**

**HR App**

Create an abstract class **Employee** with the following private member fields.

* Int emp\_id
* String name
* double salary

Provide getters, setters and a default constructor appropriately. Also add following constructor:-

Employee(int id, String name, double salary)

Employee(int id,String name)

Add an abstract method printSalary() that prints his total monthly salary upto 2 places of decimal.

Add a method called displayDetails() that prints all the details of an employee. Details include Id ,name and total monthly salary calculated by printSalary().

There are 2 types of employees:-**FullTimeEmployee** and **PartTimeEmployee**. Data members and member functions in these classes are follows respectively.

**FullTimeEmployee**

private Int emp\_id

private String name

private double salary

private int overTimeHrs

Provide getters, setters and default constructor appropriately.Over time hours cannot exceed 20 hrs. Also add the following constructosr:-

FullTimeEmployee(int id,String name,double salary,int overTimeHrs)

Implement printSalary method and display the total amount upto 2 places of decimal using the given formula considering charge per hour as 1000/-.

Salary +overTimeHrs\*charge

**PartTimeEmployee**

private int emp\_id

private String name

private double salary

private int noOfDays

private int chargePerDay

Provide getters, setters and default constructor appropriately. No. of days cannot exceed 30.Add the following constructor:-

PartTimeEmployee(int id,String name,int noOfDays,int chargePerDay)

Implement printSalary method and display the total amount upto 2 places of decimal using the given formula : noOfDays\*chargePerDay

Create a class named Employee with the following public methods.

|  |  |  |
| --- | --- | --- |
| Sr No | Name | Description |
| 1. | void printSalary() | This method is used to print all the salary of an employee |
| 2. | void displayDetails() | This method displays all the data of an employee. |

Create another class **Main** and write the main method to test these classes. Implement **dynamic polymorphism** and accept and print details of an Full Time Employee or Part Time Employee.

**Sample Input and Output 1:**

Enter Employee Details

Enter Employee Id

100

Enter Employee Name

**Geeta**

Enter Type of Employee: 1. FullTimeEmployee 2. PartTimeEmployee

1

Enter Employee salary

30000

Enter Employee Over Time Hours

10

Total Employee Details:-

Employee Id: 100

Employee Name :Geeta

Employee Total Salary : 40000.00

**Sample Input and Output 2:**

Enter Employee Details

Enter Employee Id

100

Enter Employee Name

**Geeta**

Enter Type of Employee: 1. FullTimeEmployee 2. PartTimeEmployee

1

Enter Employee salary

30000

Enter Employee Over Time Hours

25

Over Time cannot exceed 20Hrs.Over Time will be considered as 0hrs.

Total Employee Details:-

Employee Id: 100

Employee Name: Geeta

Employee Total Salary : 30000.00

**Sample Input and Output 3:**

Enter Employee Details

Enter Employee Id

101

Enter Employee Name

**Ashok**

Enter Type of Employee : 1. FullTimeEmployee 2. PartTimeEmployee

2

Enter Number Of Days Worked

20

Enter Charge Per Day

**7000**

Total Employee Details:-

Employee Id :101

Employee Name :Ashok

Employee Total Salary :140000

**Sample Input and Output 4:**

Enter Employee Details

Enter Employee Id

101

Enter Employee Name

**Ashok**

Enter Type of Employee : 1. FullTimeEmployee 2. PartTimeEmployee

**3**

Invalid Type Of Employee.Try again

1. **Requirement Specification:**

|  |  |
| --- | --- |
| <Req #> | <Req Name> |
| **Functional Requirements** | |
|  | |
| **Trigger** – <How to get there?>. | |
| **Post Conditions** – <State of the system after the action is performed> | |
| **Success End Condition** – <Happy Path Specification> | |
| **Failed End Condition** – <Unhappy Path Specification, along with boundary conditions> | |
| **Steps and Actions**   1. <Flow of Events> | |
| **Business Rules and Validations**   1. <Validations and rules> | |

1. **Technical Specifications:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Class Name** | **Method Name** | **Input Parameters** | **Sample Input** | **Output Parameters** |
| Employee | printSalary |  |  | **String**  Employee Total  salary : \_\_\_\_\_ |
| Employee | displayDetails |  |  | **String**  Employee Id : \_\_\_  Employee Name :\_\_\_  Employee Total  Salary **: \_\_\_\_\_** |
| FullTimeEmployee | printSalary |  |  | **String**  Employee Total  salary : \_\_\_\_\_ |
| PartTimeEmployee | printSalary |  |  | **String**  Employee Total  salary : \_\_\_\_\_ |

1. **Skeleton File for Development:**

Import the below attached skeleton code into your eclipse project and implement the required functionalities

**Employee.java**

**package** college.files;

**package** pack1;

**public** **class** Employee {

**private** int id;

**private** String name;

**private** double salary;

**public** abstract void printSalary();

**public** void displayDetails(){

//code

}

}

**FullTimeEmployee.java**

**package** pack1;

**public** **class** FullTimeEmployee

{

**private** int overTimeHrs;

**public** void printSalary(){

//code

}

}

**PartTimeEmployee.java**

**package** pack1;

**public** **class** PartTimeEmployee

{ **private** int noOfDays;

**private** int chargePerHr;

**public** void printSalary(){

//code

}

}

**Main.java**

**package** challange1;

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

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

// code

}

}

1. **Solution:**

**Employee.java**

**package pack1;**

**public** **abstract** **class** Employee {

**private** **int** id=1;

**private** String name="";

**private** **double** salary=0.0;

**public** Employee()

{id=1; name="";salary=0.0;

}

**public** Employee(**int** id,String name)

{**this**.id=id; **this.name**=name;

}

**public** Employee(**int** id,String name,**double** salary)

{**this**.id=id; **this.name**=name; **this.salary**=salary;

}

**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** **double** getSalary() {

**return** salary;

}

**public** **void** setSalary(**double** salary) {

**this**.salary = salary;

}

**public** **abstract** **void** printSalary();

**public** **void** displayDetails()

{System.***out***.println ("Employee Id :"+id);

System.***out***.println ("Employee Name :"+name);

printSalary ();

}

} }

**FullTimeEmployee.java**

**package pack1**;

**public** **class** FullTimeEmployee **extends** Employee{

**private** **int** overTimeHrs=0;

**public** FullTimeEmployee()

{overTimeHrs=0;}

**public** FullTimeEmployee(**int** id,String name,**double** salary,**int** overTimeHrs)

{**super**(id,name,salary);

**if**(overTimeHrs<=20)

**this**.overTimeHrs = overTimeHrs;

**else**

System.***out***.println ("Over Time cannot exceed 20Hrs.It will be considered as 0");

}

**public** **int** getOverTimeHrs() {

**return** overTimeHrs;

}

**public** **void** setOverTimeHrs(**int** overTimeHrs) {

**if**(overTimeHrs<=20)

**this**.overTimeHrs = overTimeHrs;

}

**public** **void** printSalary()

{ **double** salary=getSalary();

System.***out***.printf("%s %7.2f%n","Employee Total Salary : ",(salary+overTimeHrs\*1000));

}

}

**PartTimeEmployee.java**

**package pack1;**

**public** **class** PartTimeEmployee **extends** Employee{

**private** **int** noOfDays;

**private** **int** chargePerHr;

**public** PartTimeEmployee()

{noOfDays=0;

chargePerHr=0;

}

**public** PartTimeEmployee(**int** id,String name,**int** noOfDays,**int** chargePerHr)

{**super** (id,name);

**this**.noOfDays=noOfDays;

**this**.chargePerHr=chargePerHr;

}

**public** **int** getNoOfDays() {

**return** noOfDays;

}

**public** **void** setNoOfDays(**int** noOfDays) {

**this**.noOfDays = noOfDays;

}

**public** **int** getChargePerHr() {

**return** chargePerHr;

}

**public** **void** setChargePerHr(**int** chargePerHr) {

**this**.chargePerHr = chargePerHr;

}

**public** **void** printSalary()

{

System.***out***.printf ("%s%d%n","Employee Total Salary :",noOfDays\*chargePerHr);

}

}

**Main.java**

**package** challange1;

**import** java.util.Scanner;

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

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

**int** empid, overTimeHrs,noOfDays,chargePerDay,typeOfEmp;String name; **double** salary;

Employee emp=**null**;

System.***out***.println ("Enter Employee Details");

Scanner sc=**new** Scanner(System.***in***);

System.***out***.println ("Enter Employee Id");

empid=sc.nextInt();

System.***out***.println ("Enter Employee Name");

name=sc.next();

System.***out***.println ("Enter Type of Employee: 1. FullTimeEmployee 2. PartTimeEmployee");

typeOfEmp=sc.nextInt();

**switch** (typeOfEmp)

{ **case** 1:

System.***out***.println ("Enter Employee salary");

salary=sc.nextDouble ();

System.***out***.println ("Enter Employee Over Time Hours");

OverTimeHrs=sc.nextInt ();

emp=**new** FullTimeEmployee (empid, name, salary, overTimeHrs);

emp.displayDetails ();

**break**;

**case** 2:

System.***out***.println ("Enter Number Of Days Worked");

noOfDays=sc.nextInt();

System.***out***.println ("Enter Charge Per Day");

chargePerDay=sc.nextInt();

emp=**new** PartTimeEmployee(empid,name,noOfDays,chargePerDay);

emp.displayDetails ();

**break**;

// **TODO** Auto-generated method stub

**default**:

System.***out***.println ("Invalid Type. Try again");

}

sc.close();

}}

1. **Business Scenario 2:**

Create 2 text files containing information about Staffs and Students respectively.

Create a package called college.files with a class called ConnectFile .The class has a static method called readFile() as follows:-

public static Reader readFile(String filename)

This method creates a FileReader object with the given file and returns the same.

Create a package called college.staff with 2 classes called Staff and StaffDetails. The data members of Staff class is as follows:

private int emp\_id; private String name; private String deptName;

It has following constructors:-

public staff(){}

public staff(int emp\_id,String name,String deptName)

Override **toString()** in the format given in the sample output.

The StaffDetails class contains main() which creates a reader object with Staffs.txt file using static method of ConnectFile class,converts each line into a Staff object, stores it in the collection and prints them on the screen.

Create another package called college.student with 2 classes called Student and StudentDetails. The data members of Student class is as follows:-

private int student\_id; private String name; private String deptName; private double score;

It has following constructors:-

public Student();

public Student(int student\_id,String name,String deptName,double score)

Override **toString()** in the format given in the sample output.

The StudentDetails class contains main() which creates a reader object with Students.txt file using static method of ConnectFile class,converts each line into a Staff object, stores it in the collection and prints them on the screen.

Create a class named ConnectFile with the following public methods.

|  |  |  |
| --- | --- | --- |
| Sr No | Name | Description |
| 1. | Reader readFile() | This method is used to print all the salary of an employee |

**Sample Input and Output 1:**

Enter Employee Details

Enter Employee Id

100

Enter Employee Name

**Geeta**

Enter Type of Employee: 1. FullTimeEmployee 2. PartTimeEmployee

1

Enter Employee salary

30000

Enter Employee Over Time Hours

10

Total Employee Details:-

Employee Id: 100

Employee Name :Geeta

Employee Total Salary : 40000.00

**Sample Input and Output 2:**

Enter Employee Details

Enter Employee Id

100

Enter Employee Name

**Geeta**

Enter Type of Employee: 1. FullTimeEmployee 2. PartTimeEmployee

1

Enter Employee salary

30000

Enter Employee Over Time Hours

25

Over Time cannot exceed 20Hrs.Over Time will be considered as 0hrs.

Total Employee Details:-

Employee Id: 100

Employee Name: Geeta

Employee Total Salary : 30000.00

1. **Requirement Specification:**

|  |  |
| --- | --- |
| <Req #> | <Req Name> |
| **Functional Requirements** | |
|  | |
| **Trigger** – <How to get there?>. | |
| **Post Conditions** – <State of the system after the action is performed> | |
| **Success End Condition** – <Happy Path Specification> | |
| **Failed End Condition** – <Unhappy Path Specification, along with boundary conditions> | |
| **Steps and Actions**   1. <Flow of Events> | |
| **Business Rules and Validations**   1. <Validations and rules> | |

1. **Technical Specifications:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Class Name** | **Method Name** | **Input Parameters** | **Sample Input** | **Output Parameters** |
| ConnectFile | readFile | **FileName** |  |  |

1. **Skeleton File for Development:**

Import the below attached skeleton code into your eclipse project and implement the required functionalities

**ConnectFile.java**

**package** college.files;

**public** **class** ConnectFile {

**public** static Reader readFile(String filename);

}

**Staffs.java**

**package** college.staff;

**public** **class** Staff{

**private** int emp\_id;

**private** String name;

**private** String deptName;

**public** Staff(){

//code

}

**public** Staff(int emp\_id,String name,String deptName){

//code

}

}

**StaffDetails.java**

**package** college.staff;

**public** **class** StaffDetails

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

//code

}

**Student.java**

**package** college.student

**class** Student

{

**private** int student\_id;

**private** String name;

**private** String deptName;

**private** double score;

**public** Student(){

//code

}

**public** Student(int student\_id ,String name, String deptName, double score){

//code

}

**StudentDetails.java**

**package** college.student

**class** StudentDetails

{

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

}

}

1. **Solution:**

**ConnectFile.java**

**package college.files;**

**import** java.io.\*;

**public** **class** ConnectFile {

**public** **static** BufferedReader readFile (String fname)

{**try**

{

FileReader fr=**new** FileReader (fname);

BufferedReader br=**new** BufferedReader (fr);

**return** br;

}**catch**(IOException ioe)

{System.***out***.println (ioe.toString ());

**return** **null**;

}

}

}

**Staff.java**

**package college.staff;**

**class** Staff {

**private** **int** empId=1;

**private** String name="";

**private** String deptName="";

Staff()

{empId=1;name="";deptName="";

}

Staff(**int** empId, String name, String deptName)

{**this**.empId=empId; **this**.name=name; **this**.deptName=deptName;

}

**public** String toString()

{**return** String.*format*("%12s %15s %13s",empId, name,deptName);}

}

**StaffDetails.java**

**package college.staff;**

**import** java.io.\*;

**import** college.files.\*;

**import** java.util.\*;

**public** **class** StaffDetails

{

**public** **static** **void** getStaffDetails (String fname) **throws** IOException

{ **int** empid;String name;String deptName;

ArrayList<Staff> staffList=**new** ArrayList<Staff>();

Staff staff=**null**;

String[] staffDetails=**null**;

String line=**null**;

BufferedReader br=ConnectFile.*readFile* (fname);

**while**((line=br.readLine())!=**null**)

{staffDetails=line.split(" ");

empid=Integer.*parseInt* (staffDetails[0]);

name=staffDetails[1];

deptName=staffDetails[2];

staff=**new** Staff(empid, name,deptName);

staffList.add (staff);

}

System.***out***.printf("%12s %15s %15s%n","Employee Id","Name","Department");

**for**(Staff stf:staffList)

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

}}

**Student.java**

**package college.student**;

**public** **class** Student {

**private** **int** studentId=1;

**private** String name="";

**private** String deptName="";

**private** **double** score=0.0;

Student()

{studentId=1;name="";deptName="";score=0.0; }

Student(**int** studentId,String name,String deptName,**double** score)

{**this**.studentId=studentId;**this**.name=name;**this**.deptName=deptName;**this**.score=score;

}

**public** String toString ()

{**return** String.*format* ("%12s %15s %13s %15.2f", studentId,name,deptName,score);}

}

**StudentDetails.java**

**package college.student;**

**import** java.io.\*;

**import** college.files.\*;

**import** java.util.\*;

**public** **class** StudentDetails

{

**public** **static** **void** getStudentDetails(String fname) **throws** IOException

{

**int** studentId; String name;String deptName; **double** score;

ArrayList<Student> studentList=**new** ArrayList<Student>();

Student student=**null**;

String[] studentDetails=**null**;

String line=**null**;

BufferedReader br=ConnectFile.*readFile* (fname);

**while**((line=br.readLine ())!=**null**)

{studentDetails=line.split (" ");

studentId=Integer.*parseInt* (studentDetails[0]);

name=studentDetails[1];

deptName=studentDetails[2];

score=Double.*parseDouble* (studentDetails[3]);

student=**new** Student (studentId, name, deptName, score);

studentList.add (student);

}

System.***out***.printf("%12s %15s %15s %12s%n","Student Id","Name","Department","Score");

**for**(Student std:studentList)

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

}

}

**CollegeInfo.java**

**package college;**

**import** java.util.\*;

**import** java.io.\*;

**import** college.staff. StaffDetails;

**import** college.student. StudentDetails;

**public** **class** CollegeInfo {

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

// **TODO** Auto-generated method stub

**int** choice;

Scanner sc=**new** Scanner (System.***in***);

System.***out***.println ("Welcome to College Database");

System.***out***.println ("Please Enter Whose Information You Want? 1.Staff 2.Student");

choice=sc.nextInt ();

**switch** (choice)

{**case** 1:

StaffDetails.*getStaffDetails* ("Staff.txt");

**break**;

**case** 2:

StudentDetails.*getStudentDetails* ("Student.txt");

**break**;

**default**:

System.***out***.println ("Invalid Choice!!!Try Again");

}

}

}