FlexJobs(---RETIRED---)

Grade settings: Maximum grade: 100
Disable external file upload, paste and drop external content: Yes
Based on: FlexJobs(---RETIRED---)
Run: Yes **Evaluate**: Yes
Automatic grade: Yes

FlexJobs is one of the most reputable organizations. They have chosen to develop an application for freelance workers that would gather their information and determine their pay. The management has approached you to develop software that will determine how much has to be paid.

Component Specification: EmployeeInfo

Type (Class)	Attributes	Methods
EmployeeInfo	String employeeId String employeeName String jobName int yearsOfExperience int workedHours	Necessary getters, setters and five argument constructor are provided as part of the code skeleton.

Functional Requirement 1: Extract the employeeDetails and create an object of the EmployeeInfo class.

Type (Class)	Methods	Responsibilities
UserInterface	public static EmployeeInfo **extractDetails**(String employeeDetails)	This method accepts employeeDetails separated by colon as an argument and should extract the properties of the EmployeeInfo from the argument by parsing the employeeDetails. Set these values to

		the EmployeeInfo object and return this object.

Functional Requirement 2: Calculate the salary based on jobName and yearsOfExperience

Type (Class)	Methods	Responsibilities
EmployeeInfo	public double **calculateSalary()**	This method should calculate the salary based on the jobName and yearsOfExperience ***Condition:*** - jobName is **case in-sensitive** - The jobName must be either one of [" **Video Editor** "," **Teaching Assistant** ","**Telecalling Executive**"," **Graphic Designer** "], else return 0 - If the yearsOfExperience and workedHours is less than 0, then return 0.

Note: The class and methods should be declared as public, and all the attributes should be declared as private.

Guidelines to calculate the salary based on jobName and yearsOf Experience

jobName	basicPay	yearsOfExperience	increasedPercentage
Video Editor	15	<=2	10%
		<=5	25%
		>=6	40%
Teaching Assistant	50	<=2	10%
		<=5	25%
		>=6	40%

Telecalling Executive	25	<=2	10%
		<=5	25%
		>=6	40%
Graphic Designer	60	<=2	10%
		<=5	25%
		>=6	40%

Formula to calculate the Salary in the EmployeeInfo class:

salary =(workedHours*basicPay)+(workedHours*basicPay*increasedPercentage)

[Obtain the basicPay and increasedPercentage from the above table]

Example

Sample Input: E012:Keni:Graphic Designer:4:5

Here the jobName is "Graphic Designer" and yearsOfExperience is "4", then increasedPercentage is 25%(that is 0.25)

salary = (5*60) +(5*60 *0.25) = 300+(300+0.25) = 625.0

The main method in the UserInterface class is excluded from the evaluation. You are free to write your own code in the main method to invoke the business methods to check its correctness.

Note:

- In the Sample Input / Output provided, the highlighted text in bold corresponds to the input given by the user and the rest of the text represents the output.

- Ensure to follow the object-oriented specifications provided in the question.

- Ensure to provide the names for classes, attributes, and methods as specified in the question.

- Adhere to the code template, if provided.

Sample Input/Output 1:

Enter the Employee details

E001:Mark:Graphic Designer:1:8

Employee Id : E001

Employee Name : Mark

Job name: Graphic Designer

Years Of Experience:1

Worked Hours : 8

Salary Amount : $528.0

Sample Input/Output 2:

Enter the Employee details

E042:Antony:Video Editor:2:0

Invalid Employee details

[**Explanation**: The workedHours should be greater than 0]

Sample Input/Output 3:

Enter the Employee details

E079:Amer:Telecalling Executive:-2:2

Invalid Employee details

[**Explanation**: The yearsOfExperience should be greater or equal to 0]

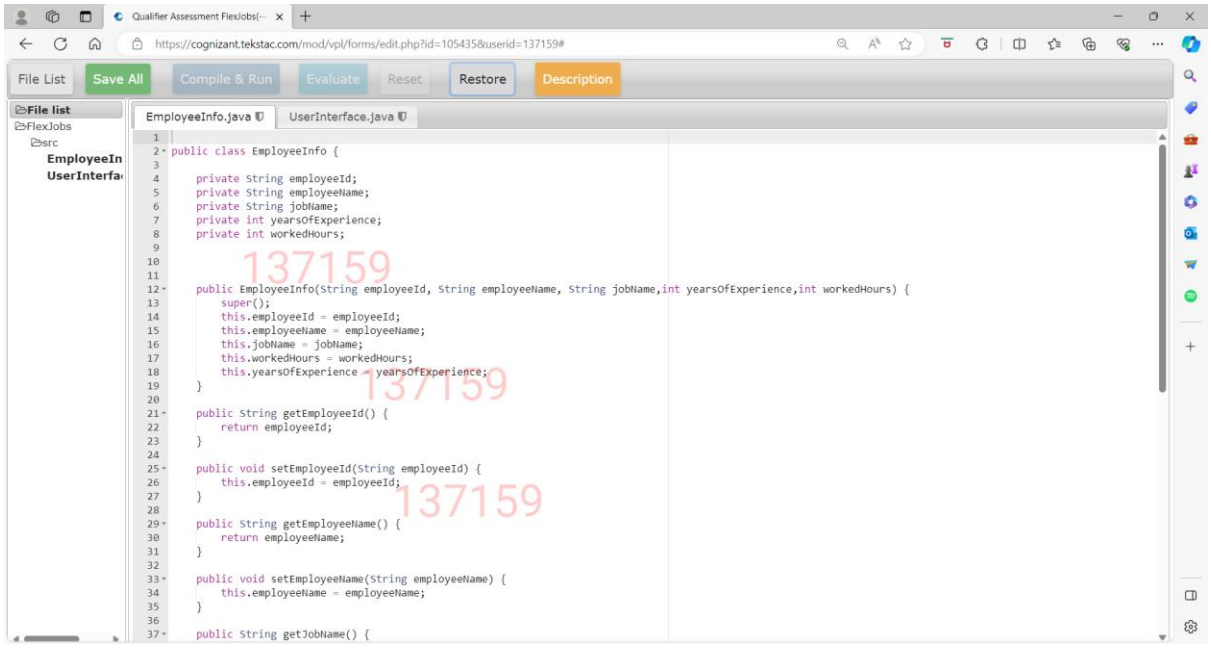
Sample Input/Output 4:

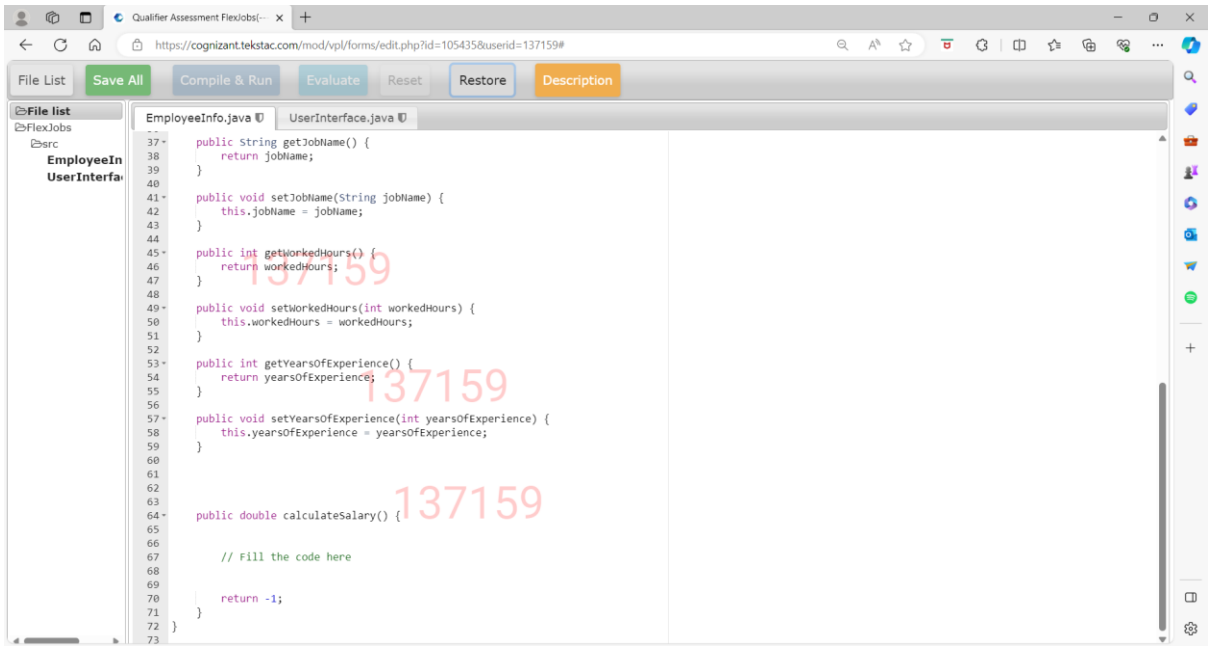
Enter the Employee details

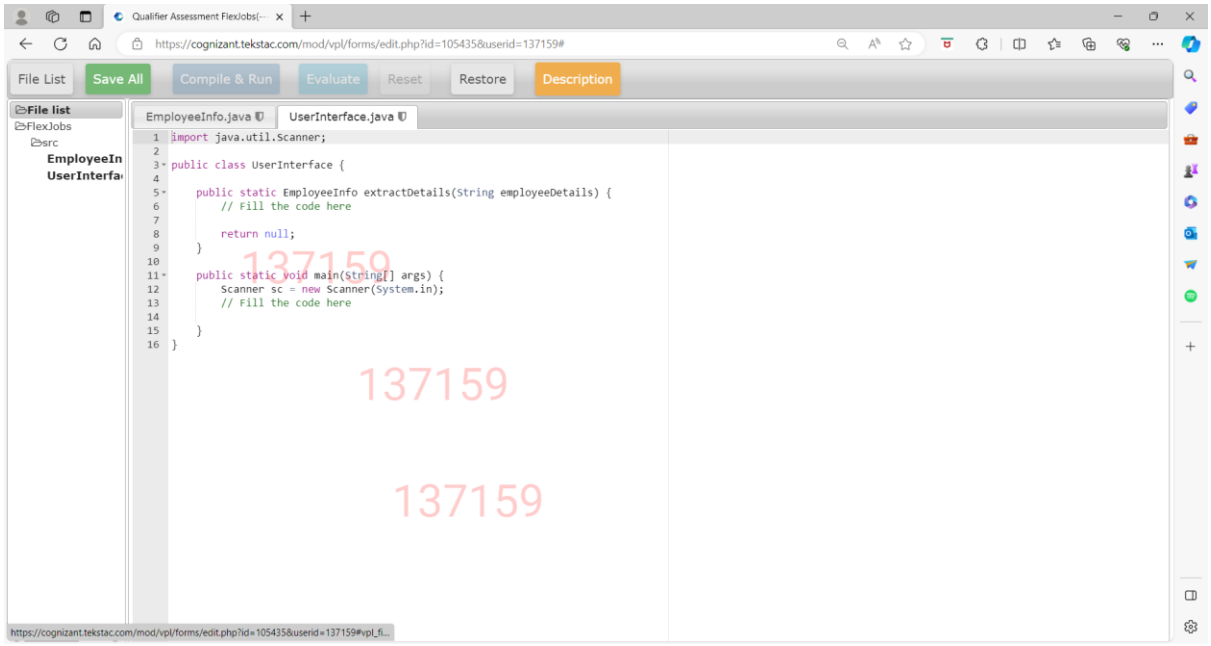
E082:Paul:Data Entry:3:2

Invalid Employee details

[**Explanation**: The jobName is not present in the above mentioned type.]


Qualifier Assessment FlexJobs(
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=105435&userid=137159#
File List
Save All
Compile & Run
Evaluate
Reset
Restore
Description
File list
FlexJobs
src
EmployeeIn
UserInterfa
EmployeeInfo.java
UserInterface.java
public class EmployeeInfo {
private String employeeId;
private String employeeName;
private String jobName;
private int yearsOfExperience;
private int workedHours;
137159
public EmployeeInfo(String employeeId, String employeeName, String jobName,int yearsOfExperience,int workedHours) {
super();
this.employeeId = employeeId;
this.employeeName = employeeName;
this.jobName = jobName;
this.workedHours = workedHours;
this.yearsOfExperience = yearsOfExperience;
}
137159
public String getEmployeeId() {
return employeeId;
}
public void setEmployeeId(String employeeId) {
this.employeeId = employeeId;
}
137159
public String getEmployeeName() {
return employeeName;
}
public void setEmployeeName(String employeeName) {
this.employeeName = employeeName;
}
public String getJobName() {


Qualifier Assessment FlexJobs(
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=105435&userid=137159#
File List
Save All
Compile & Run
Evaluate
Reset
Restore
Description
File list
FlexJobs
src
EmployeeIn
UserInterfa
EmployeeInfo.java
UserInterface.java
public String getJobName() {
return jobName;
}
public void setJobName(String jobName) {
this.jobName = jobName;
}
public int getWorkedHours() {
return workedHours;
}
137159
public void setWorkedHours(int workedHours) {
this.workedHours = workedHours;
}
public int getYearsOfExperience() {
return yearsOfExperience;
}
137159
public void setYearsOfExperience(int yearsOfExperience) {
this.yearsOfExperience = yearsOfExperience;
}
137159
public double calculateSalary() {
// Fill the code here
return -1;
}
}


Qualifier Assessment FlexJobs(
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=105435&userid=137159#
File List
Save All
Compile & Run
Evaluate
Reset
Restore
Description
File list
FlexJobs
src
EmployeeIn
UserInterfa
EmployeeInfo.java
UserInterface.java
1 import java.util.Scanner;
2
3 public class UserInterface {
4
5 public static EmployeeInfo extractDetails(String employeeDetails) {
6 // Fill the code here
7
8 return null;
9 }
10
11 public static void main(String[] args) {
12 Scanner sc = new Scanner(System.in);
13 // Fill the code here
14
15 }
16 }
137159
137159
137159
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=105435&userid=137159#vpl_fi...