#### **Task 1**

**Study the following output**

|  |  |
| --- | --- |
| **Code** | **Output** |
| **public class StudentTest{**  **public static void main(String [] args){**  **Student s1 = new Student();**  **System.out.println(s1.getName());**  **Student s2 = new Student("Matin");**  **System.out.println(s2.getName());**  **Student s3 = new Student("Saad");**  **System.out.println(s3.getName());**  **System.out.println(Student.numberOfStudents);**  **}**  **}** | **default name**  **Matin**  **Saad**  **3** |

Write the code for the Student class so that the StudentTest class generates the output shown above.

**TASK 2**

Create a class called BankAccount as described below:

* **Fields:**name, address, accountID, balance
* **Methods:**   
  public String getName()  
  public void setName(String n)  
  public String getAccountID()  
  public void setAccountID(String i)  
  public String getAddress()  
  public void setAddress(String a)  
  public double getBalance()  
  public void setBalance(double c)  
  public void addInterest() //adds 7% of the balance

Write a class called AccountTester to write a main() method:

* public static void main(String[] args){  
    
  }
* Inside the main() method
  + Create 3 objects/instances of BankAccount called acc1, acc2 and acc3
  + Set their fields to some value using the public methods.
  + Call addInterest() on acc1 and acc3
  + Print the information of each BankAccount using System.out.println()

Add constructors to BankAccount and use the constructor to set the field values.

##### Task 3

Create a class **SavingsAccount,** which will use a **static** class variable to store the **annualInterestRate** for all account holders.

* Each object of the class contains a **private** instance variable **savingsBalance** indicating the amount the saver currently has on deposit.
* Provide method **calculateMonthlyInterest()** to calculate the monthly interest [by multiplying the **savingsBalance** by **annualInterestRate** divided by 12], this interest should be added to **savingsBalance**.
* Provide a **static** method **modifyInterestRate()** that sets the **annualInterestRate** to a new value.

Write a driver program to test class **SavingsAccount**.

* Instantiate two **SavingsAccount** objects, **saver1** and **saver2**, with balances $20000.00 and $30000.00, respectively using constructor.
* Set **annualInterestRate** to 4.2%, then calculate the monthly interest and print the new balances for each of the savers using **printSavingsBalance( )** method.
* Then set the **annualInterestRate** to 5.5% and calculate the next month’s interest and print the new balances for each of the savers.

Write the code for the Student class so that the StudentTest class generates the output shown above.

**Task 4**

**Output:**

**========================**

**Name: Saad Abdullah**

**Department: CSE**

**List of courses**

**========================**

**CSE 110 Programming Language I**

**CSE 111 Programming Language-II**

**========================**

**========================**

**Name: Mumit Khan**

**Department: CSE**

**List of courses**

**========================**

**CSE 220 Data Structures**

**CSE 221 Algorithms**

**CCSE 230 Discrete Mathematics**

**========================**

**========================**

**Name: Sadia Kazi**

**Department: CSE**

**List of courses**

**========================**

**CSE 310 Object Oriented Programming**

**CSE 320 Data Communications**

**CSE 340 Computer Architecture**

**========================**

**Tester:**

public class TestTeacher{

public static void main(String [] args){

Teacher t1 = new Teacher("Saad Abdullah", "CSE");

Teacher t2 = new Teacher("Mumit Khan", "CSE");

Teacher t3 = new Teacher("Sadia Kazi", "CSE");

Course c1 = new Course("CSE 110 Programming Language I");

Course c2 = new Course("CSE 111 Programming Language-II");

Course c3 = new Course("CSE 220 Data Structures");

Course c4 = new Course("CSE 221 Algorithms");

Course c5 = new Course("CCSE 230 Discrete Mathematics");

Course c6 = new Course("CSE 310 Object Oriented Programming");

Course c7 = new Course("CSE 320 Data Communications");

Course c8 = new Course("CSE 340 Computer Architecture");

t1.addCourse(c1);

t1.addCourse(c2);

t2.addCourse(c3);

t2.addCourse(c4);

t2.addCourse(c5);

t3.addCourse(c6);

t3.addCourse(c7);

t3.addCourse(c8);

t1.printDetail();

t2.printDetail();

t3.printDetail();

}

}

Write the Teacher and Course classes so that the TestTeacher class produces the outputs given above

**Task 5**

Consider the following code:

**public class AccountTester**

**{**

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

**{**

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

**Account a1 = new Account();**

**System.out.println(a1.nameKi());**

**System.out.println(a1.balanceKi());**

**a1.nameBoshao("Mashrafe Murtaza");**

**a1.balanceBoshao(5000);**

**System.out.println(a1.nameKi());**

**System.out.println(a1.balanceKi());**

**Account a2 = new Account("Mustafizur Rahman",50);**

**System.out.println(a2.nameKi());**

**System.out.println(a2.balanceKi());**

**a1.withdraw(4900);**

**a2.withdraw(25);**

**a1.withdraw(4899);**

**}**

**}**

**Write the account class given that the output for the above code is :**

**5.0**

**Default Account**

**0.0**

**Mashrafe Murtaza**

**5000.0**

**Mustafizur Rahman**

**50.0**

**The account balance after deducting withdraw amount is equal to or less than minimum. Cannot withdraw**

**The account balance after deducting withdraw amount is equal to or less than minimum. Cannot withdraw**

**Withdraw successful! New balance is: 101.0**

**Task 6**

Complete the **Student** class so that the **main** method prints the following:

**Name of the Student: Bob**

**ID of the Student: 1**

**Name of the Student: Tom**

**ID of the Student: 2**

**Name of the Student: Jack**

**ID of the Student: 3**

**Name of the Student: Jill**

**ID of the Student: 4**

**public class Student{**

**//Your code here**

**}**

**public class Printer{**

**public void printDetail(Student s){**

**System.out.println("Name of the Student: "+s.name);**

**System.out.println("ID of the Student: "+s.id);**

**}**

**}**

**public class Test{**

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

**Student s1 = new Student("Bob", 1);**

**Student s2 = new Student("Tom", 2);**

**Student s3 = new Student("Jack", 3);**

**Student s4 = new Student("Jill", 4);**

**Printer pr = new Printer();**

**pr.printDetail(s1);**

**pr.printDetail(s2);**

**pr.printDetail(s3);**

**pr.printDetail(s4);**

**}**

**}**

**Task 7**

**public class Dog {**

**private String color = "Black";**

**//your code here**

**}**

**public class Quiz {**

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

**Dog odie = new Dog("Red");**

**Dog goofy = new Dog("Blue");**

**odie.bark();**

**goofy.bark();**

**odie.changeColor("Brown");**

**odie.bark();**

**}**

**}**

**//Complete the Dog class so the main method above produces the following output:**

**Red dog is barking**

**Blue dog is barking**

**Brown dog is barking**

**Task 8**

Page 1 of 1 SET-A

|  |
| --- |
| **public class Cat{** |
| public String color = "White"; |
| public String action = "sitting"; |
| //your code here |
| **}** |
| **public class Test{** |
| public static void main(String [] args){ |
| Cat c1 = new Cat(); |
| Cat c2 = new Cat("Black"); |
| Cat c3 = new Cat("Brown", "jumping"); |
| Cat c4 = new Cat("Red", "purring"); |
| c1.printCat(); |
| c2.printCat(); |
| c3.printCat(); |
| c4.printCat(); |
| c1.changeColor("Blue"); |
| c3.changeColor("Purple"); |
| c1.printCat(); |
| c3.printCat(); |
| } |
| **}** |