1. AirVoice - Registration

Grade settings: Maximum grade: 100
Disable external file upload, paste and drop external content: Yes
Run: Yes **Evaluate**: Yes
Automatic grade: Yes **Maximum execution time**: 16 s
SmartBuy is a leading mobile shop in the town. After buying a product, the customer needs to provide a few personal details for the invoice to be generated.

You being their software consultant have been approached to develop software to retrieve the personal details of the customers, which will help them to generate the invoice faster.

Component Specification: Customer

Type(Class)	Attributes	Methods	Responsibilities
Customer	String customerName long contactNumber String emailId int age	Include the getters and setters method for all the attributes.	

In the **Main** class, create an object for the Customer class.

Get the details as shown in the sample input and assign the value for its attributes using the setters.

Display the details as shown in the sample output using the getters method.

All classes and methods should be public, Attributes should be private.

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 1:

Enter the Name:

john

Enter the ContactNumber:

9874561230

Enter the EmailId:

john@gmail.com

Enter the Age:

32

Sample Output 1:

Name:john

ContactNumber:9874561230

EmailId:john@gmail.com

Age:32

Automatic evaluation[+]

Customer.java

```
1 public class Customer {
2    private String customerName;
3
4    private long contactNumber;
5
6    private String emailId;
7
8    private int age;
9
10    public String getCustomerName() {
11       return  customerName;
12    }
13
14    public void setCustomerName(String customerName) {
15       this.customerName = customerName;
16    }
17
18    public long getContactNumber() {
19       return contactNumber;
20    }
21
22    public void setContactNumber(long contactNumber) {
23       this.contactNumber = contactNumber;
24    }
25
26    public String getEmailId() {
27       return emailId;
28    }
29
30    public void setEmailId(String emailId) {
31       this.emailId= emailId;
32    }
```

```
33
34    public int getAge() {
35        return age;
36    }
37
38    public void setAge(int age){
39        this.age = age;
40    }
41
42
43
44 }
```

Main.java

```
1  import java.util.Scanner;
2
3  public class Main {
4
5     public static void main (String[] args) {
6        Scanner sc=new Scanner(System.in);
7
8        //Fill the code
9        Customer c=new Customer();
10        System.out.println("Enter the Name:");
11        String name=(sc.nextLine());
12        System.out.println("Enter the ContactNumber:");
13        long no=sc.nextLong();
14        sc.nextLine();
15        System.out.println("Enter the EmailId:");
16        String mail=sc.nextLine();
17
18        System.out.println("Enter the Age:");
19        int age=sc.nextInt();
20        c.setCustomerName(name);
21        c.setContactNumber(no);
22        c.setEmailId(mail);
23        c.setAge(age);
24        System.out.println("Name:"+c.getCustomerName());
25        System.out.println("ContactNumber:"+c.getContactNumber());
26        System.out.println("EmailId:"+c.getEmailId());
27        System.out.println("Age:"+c.getAge());
28
29
30
31    }
32
33 }
```

2. Grade

Reviewed on Friday, 10 December 2021, 6:14 PM by Automatic grade
Grade 100 / 100
Assessment report
[+]**Grading and Feedback**

3. ZeeZee bank

Grade settings: Maximum grade: 100
Disable external file upload, paste and drop external content: Yes
Run: Yes **Evaluate**: Yes
Automatic grade: Yes **Maximum execution time**: 16 s
ZeeZee is a leading private sector bank. In the last Annual meeting, they decided to give their customer a 24/7 banking facility. As an initiative, the bank outlined to develop a stand-alone device that would offer deposit and withdrawal of money to the customers anytime.

You being their software consultant have been approached to develop software to implement the functionality of deposit and withdrawal anytime.

Component Specification: Account

Type(Class)	Attributes	Methods	Responsibilities
Account	long accountNumber double balanceAmount	Include the getters and setters method for all the attributes. Include a parametrized constructor of two arguments in the order – accountNumber,balanceAmount to intialize the values for the account object	

Requirement 1: Being able to deposit money into an account anytime

As per this requirement, the customer should be able to deposit money into his account at any time and the deposited amount should reflect in his account balance.

Component Specification: Account

Component Name	Type(Class)	Methods	Responsibilities
Deposit amount to an account	Account	public void deposit(double depositAmt)	This method takes the amount to be deposited as an argument This method should perform the deposit,by adding the deposited amount to the balanceAmount

Requirement 2: Being able to withdraw money from the account anytime

As per this requirement, the customer should be able to withdraw money from his account anytime he wants. The amount to be withdrawn should be less than or equal to the balance in the account. After the withdrawal, the account should reflect the balance amount

Component Specification: Account

Component Name	Type(Class)	Methods	Responsibilities
Withdraw amount from an account	Account	public boolean withdraw(double withdrawAmt)	This method should take the amount to be withdrawn as an argument. This method should check the balanceAmount and deduct the withdraw amount from the balanceAmount and return true. If there is insufficient balance then return false.

In the **Main** class, Get the details as shown in the sample input.

Create an object for the Account class and invoke the deposit method to deposit the amount and withdraw method to withdraw the amount from the account.

All classes and methods should be public, Attributes should be private.

Note:

Balance amount should be displayed corrected to 2 decimal places.

Order of the transactions to be performed (Display,Deposit,Withdraw).

If the balance amount is insufficient then display the message as shown in the Sample Input / Output.

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 account number:

1234567890

Enter the available amount in the account:

15000

Enter the amount to be deposited:

1500

Available balance is:16500.00

Enter the amount to be withdrawn:

500

Available balance is:16000.00

Sample Input/Output 2:

Enter the account number:

1234567890

Enter the available amount in the account:

15000

Enter the amount to be deposited:

1500

Available balance is:16500.00

Enter the amount to be withdrawn:

18500

Insufficient balance

Available balance is:16500.00

Automatic evaluation[+]

Main.java

```
1 import java.util.Scanner;
2 import java.text.DecimalFormat;
3
4 public class Main{
5
6    public static void main (String[] args) {
7        Scanner sc=new Scanner(System.in);
8        DecimalFormat decimalFormat=new DecimalFormat("0.00");
9        System.out.println("Enter the account number:");
```

```
        long accountNumber= sc.nextLong();
        System.out.println("Enter the available amount in the account:");
        double balanceAmount= sc.nextDouble();
        Account account=new Account(accountNumber,balanceAmount);
        System.out.println("Enter the amount to be deposited:");
        double depositAmount=sc.nextDouble();
        account.deposit(depositAmount);
        double availableBalance=account.getBalanceAmount();
        System.out.println("Available balance is:"+decimalFormat.format(availableBalance));
        System.out.println("Enter the amount to be withdrawn:");
        double withdrawAmount= sc.nextDouble();
        boolean isWithdrawn=account.withdraw(withdrawAmount);
        availableBalance=account.getBalanceAmount();
        if(!isWithdrawn){
            System.out.println("Insufficient balance");
        }
        System.out.println("Available balance is:"+decimalFormat.format(availableBalance));

        //Fill the code
    }
}
```

Account.java

```

public class Account {
    private long accountNumber;
    private double balanceAmount;
    public Account(long accountNumber,double balanceAmount){
        this.accountNumber=accountNumber;
        this.balanceAmount=balanceAmount;
    }
    public long getAccountNumber(){
        return accountNumber;
    }
    public void setAccountNumber(long accountNumber){
        this.accountNumber=accountNumber;
    }
    public double getBalanceAmount(){
        return balanceAmount;
    }
    public void setBalanceAmount(double balanceAmount){
        this.balanceAmount=balanceAmount;
    }
    public void deposit(double depositAmount){
        balanceAmount+=depositAmount;
    }
    public boolean withdraw(double withdrawAmount){
        if(withdrawAmount<=balanceAmount){
            balanceAmount-=withdrawAmount;
            return true;
        }
        return false;
    }
}
```

Grade

Reviewed on Thursday, 27 May 2021, 3:28 AM by Automatic grade
Grade 100 / 100
Assessment report
[+]**Grading and Feedback**

3.Call Details

Grade settings: Maximum grade: 100
Disable external file upload, paste and drop external content: Yes
Run: Yes **Evaluate**: Yes
Automatic grade: Yes **Maximum execution time**: 16 s
AirCarrier is a leading mobile network provider. They maintain a record of all the calls made by their postpaid customers.The details are stored in a particular format [callId:calledNumber:noOfMinutes] .At the end of every month, the network provider wants to extract the information from the file and populate it to the Call object for calculating the bill.

You being their software consultant have been approached to develop software to implement the functionality of extracting the data from the given format.

Component Specification: Call

Type(Class)	Attributes	Methods	Responsibilities
Call	int callId long calledNumber float duration	Include the getters and setters method for all the attributes.	

Requirement 1: Extracting the data from the callDetails

This requirement is responsible for extracting the customer's callId, calledNumber and duration from the callDetails. After the extraction set the callId, calledNumber and duration to the call object.

Component Specification: Call

Component Name	Type(Class)	Methods	Responsibilities
Extraction from file	Call	public void parseData(String callDetails)	This method takes the callDetails as an argument This method should perform the extraction process, to set the callId, calledNumber and duration of the call object.

In the **Main** class, Get the details as shown in the sample input.

Create an object for the Call and invoke the **parseData** method to set the callId, calledNumber and duration for each customer.

Invoke the corresponding getters to display the call details as shown in the Sample Output

All classes and methods should be public, Attributes should be private.

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 1:

Enter the call details:

102:6547891230:2.15

Sample Output 1:

Call id:102

Called number:6547891230

Duration:2.15

==========

Automatic evaluation[+]

Main.java

```
1 import java.util.Scanner;
2
3 public class Main {
4
5    public static void main (String[] args) {
6        Scanner sc=new Scanner(System.in);
7        System.out.println("Enter the call details:");
8        String a=sc.nextLine();
9        Call obj=new Call();
10        obj.parseData(a);
11        System.out.println("Call id:"+obj.getCallId());
12        System.out.println("Called number:"+obj.getCalledNumber());
13        System.out.println("Duration:"+obj.getDuration());
14        //Fill the code
15
16    }
17 }
```

Call.java

```
1
2 public class Call {
3    private int callId;
4    private long calledNumber;
5    private float duration;
6    public Call(){
7    }
8    public int getCallId(){
9       return callId;
10   }
11   public long getCalledNumber(){
12      return calledNumber;
13   }
14   public float getDuration(){
15      return duration;
16   }
17   public void setCallId(int callId){
18      this.callId=callId;
19   }
20   public void setCalledNumber(long calledNumber){
21      this.calledNumber=calledNumber;
22   }
23   public void setDuration(float duration){
24      this.duration=duration;
25   }
26   public void parseData(String calld){
27      callId=Integer.parseInt(calld.split(":")[0]);
28      setCallId(callId);
29      calledNumber=Long.parseLong(calld.split(":")[1]);
30      setCalledNumber(calledNumber);
31      duration=Float.parseFloat(calld.split(":")[2]);
32      setDuration(duration);
33   }
34 }
```

Grade

Reviewed on Tuesday, 4 May 2021, 4:58 AM by Automatic grade
Grade 100 / 100
Assessment report
[+]**Grading and Feedback**

---

4. Pair of Two digits

Grade settings: Maximum grade: 100
Disable external file upload, paste and drop external content: Yes
Run: Yes **Evaluate**: Yes
Automatic grade: Yes **Maximum execution time**: 32 s
Jerold teacher assigned a task to him. The task is to find the pair of two-digit numbers.

The pair is found by checking whether the product of the numbers is same as the product of the reversed numbers. If it is same, then print "Correct pair found". If not print, "Correct pair not found".

Write a Java program to find the correct pair of two-digit numbers.

Assume both the inputs are 2-digit values.

Note:

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

Adhere to the code template, if provided.

Hint: [13*62=31*26]

Sample Input 1:

13

62

Sample Output 1:

13 and 62 are correct pair

Sample Input 2:

10

56

Sample Output 2:

10 and 56 are not correct pair

Automatic evaluation[+]

Main.java

```
1  import java.util.*;
2
3  public class Main{
4
5    public static void main (String[] args) {
6      Scanner sc=new Scanner(System.in);
7
8        int a=sc.nextInt();
9        int b=sc.nextInt();
10        if(a>99||a<10||b>99||b<10){
11            System.out.println("No");
12
13        }
14        Main obj=new Main();
15        int ra=obj.rvs(a);
16        int rb=obj.rvs(b);
17        if(a*b==ra*rb){
18            System.out.println(a+" and "+b+" are correct pair");
19        }
```

```
20        else{
21            System.out.println(a+" and "+b+" are not correct pair");
22        }
23    }
24    int rvs(int num){
25        int r,rnum=0;
26        while(num>0)
27        {
28            r=num%10;
29            rnum=rnum*10+r;
30            num/=10;
31        }
32        return(rnum);
33    }
34 }
35
36
37
38
39
40
```

Grade

Reviewed on Thursday, 27 May 2021, 3:38 AM by Automatic grade
Grade 100 / 100
Assessment report
TEST CASE PASSED
[+]**Grading and Feedback**

----------End----------