~

## CS23333-Object Oriented Programming Using Java-2023

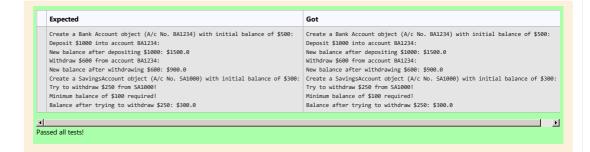
Dashboard / My courses / CS23333-OOPUJ-2023 / Lab-05-Inheritance / Lab-05-Logic Building

## Quiz navigation



Show one page at a time Finish review

```
Status Finished
               Started Saturday, 5 October 2024, 10:19 PM
           Completed Saturday, 5 October 2024, 10:58 PM
          Duration 38 mins 33 secs
Question 1
                          Create a class known as "BankAccount" with methods called deposit() and withdraw().
                          Create a subclass called SavingsAccount that overrides the withdraw() method to prevent withdrawals if the account balance falls below one hundred.
Marked out of
5.00
Flag question
                            Result
                            Create a Bank Account object (A/c No. BA1234) with initial balance of $500: Deposit $1000 into account BA1234:
                            New balance after depositing $1000: $1500.0
                            Withdraw $600 from account BA1234:
New balance after withdrawing $600: $900.0
                            Create a SavingsAccount object (A/c No. SA1000) with initial balance of $300:
                            Try to withdraw $250 from SA1000!
Minimum balance of $100 required!
                            Balance after trying to withdraw $250: $300.0
                           Answer: (penalty regime: 0 %)
                            Reset answer
                             // Private field to store the balance
                                           private double balance;
                                          // Constructor to initialize account number and balance
                              10
11
                                           public BankAccount(String accountNumber,double balance)
                              12
                                                 this.accountNumber=accountNumber;
                              13
                                                 this.balance=balance;
                              14
15
16
                              17
18
19
20
21
22
                                          // Method to deposit an amount into the account public void deposit(double amount) {
    // Increase the balance by the deposit amount
                                            balance+=amount;
                              23
                                               Method to withdraw an amount from the account
                                           public void withdraw(double amount) {
    // Check if the balance is sufficient for the withdrawal
    if (balance >= amount) {
        // Decrease the balance by the withdrawal amount
                              25
26
27
28
                              29
                                                       balance -= amount;
                                                 30
                               31
32
                                                }
                              33
34
35
36
37
                                          // Method to get the current balance
public double getBalance() {
    // Return the current balance
                              38
                              39
40
41
                                                 return balance;
                                           public String getAccountNumber()
                              42
                              43
44
45
                                                return accountNumber:
                              46
                                      class SavingsAccount extends BankAccount {
    // Constructor to initialize account number and balance
    public SavingsAccount(String accountNumber,double balance) {
        // Call the parent class constructor
        super(accountNumber,balance);
}
                              47
                              48
49
50
51
52
                              53
                                           // Override the withdraw method from the parent class
                              55
56
57
                                            gOverride
public void withdraw(double amount) {
    // Check if the withdrawal would cause the balance to drop below $100
    if (getBalance() - amount < 100) {
        // Print a message if the minimum balance requirement is not met
        System.out.println("Minimum balance of $100 required!");</pre>
                              58
                              59
60
                              61
62
                                                 } else {
    // Call the parent class withdraw method
                              63
64
                                                       super.withdraw(amount);
```



65 66 67 Student class, with department attribute, Course() method to sub class. Print the details of the Student. College: String collegeName; public College() { } public admitted() { } Student: String studentName; String department; public Student(String collegeName, String studentName,String depart) { } public toString() Expected Output: A student admitted in REC CollegeName : REC StudentName : Venkatesh Department : CSE For example: Result A student admitted in REC CollegeName : REC StudentName : Venkatesh Department : CSE

Answer: (penalty regime: 0 %)

Correct

Marked out of

Flag question

```
Reset answer
1 class College
        protected String collegeName;
  public College(String collegeName) {
    // initialize the instance variables
    this.collegeName=collegeName;
  10 ·
11
12
       public void admitted() {
   System.out.println("A student admitted in "+collegeName);
  13
  class Student extends College{

15

16 String studentName;

17 String department;
  18
  19
20
21
        public Student(String collegeName, String studentName,String department) {
            // initialize the instance variables
super(collegeName);
            this.studentName=studentName;
this.department=department;
  22
23
24
25
  26
       public String toString(){
   // return the details of the student
   return "CollegeName : "+collegeName+"\nStudentName : "+studentName+"\nDepartment : "+department;
  27
28
29
30
31
32
33
       34
  35
36
37
  38 }
```

```
Expected

A student admitted in REC
CollegeName : REC
StudentName : Venkatesh
Department : CSE

CollegeName : CSE

CollegeName : REC
CollegeName : REC
StudentName : Venkatesh
Department : CSE

Passed all tests!
```

Question **3**Correct
Marked out of 5.00
Friag question

Create a class Mobile with constructor and a method basicMobile().

 $\label{thm:constructor} Create\ a\ subclass\ Camera Mobile\ which\ extends\ Mobile\ class\ ,\ with\ \ constructor\ and\ \ a\ method\ \ new Feature ().$ 

 $Create\ a\ subclass\ Android Mobile\ which\ extends\ Camera Mobile, with\ \ constructor\ and\ \ a\ method\ android Mobile().$ 

display the details of the Android Mobile class by creating the instance.

class Mobile{

}
class CameraMobile extends Mobile {
}
class AndroidMobile extends CameraMobile {
}
expected output:
Basic Mobile is Manufactured
Camera Mobile is Manufactured
Android Mobile is Manufactured
Camera Mobile with 5MG px
Touch Screen Mobile is Manufactured

For example:

Result

Basic Mobile is Manufactured

Camera Mobile is Manufactured

Android Mobile is Manufactured

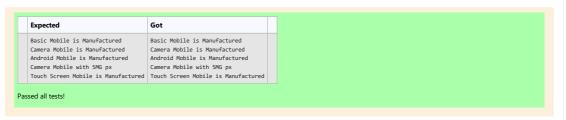
Camera Mobile with 5MG px

Touch Screen Mobile is Manufactured

Answer: (penalty regime: 0 %)

1 class Mobile

```
public Mobile()
            System.out.println("Basic Mobile is Manufactured");
public void basicMobile()
     class CameraMobile extends Mobile
        public CameraMobile()
            super();
System.out.println("Camera Mobile is Manufactured");
        public void newFeature()
{
            System.out.println("Camera Mobile with 5MG px");
     class AndroidMobile extends CameraMobile
        public AndroidMobile()
{
            super();
System.out.println("Android Mobile is Manufactured");
        public void androidMobile()
{
            System.out.println("Touch Screen Mobile is Manufactured");
     public class Main
        public static void main(String args[])
            AndroidMobile a =new AndroidMobile();
            a.newFeature();
a.androidMobile();
```



\$

Finish review

**◄** Lab-05-MCQ

Jump to...

Is Palindrome Number? ►