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

Status	Finished
Started	Tuesday, 8 October 2024, 6:21 PM
Completed	Tuesday, 8 October 2024, 6:51 PM
Duration	30 mins 11 secs

Question **1**

Correct

Marked out of 5.00

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

Create a subclass CameraMobile which extends Mobile class , with constructor and a method newFeature().

Create a subclass AndroidMobile which extends CameraMobile, with constructor and a method androidMobile().

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 %)

```
class mobile{
    mobile(){
        System.out.println("Basic Mobile is Manufactured");
    }
}
class camera extends mobile{
    camera(){
        System.out.println("Camera Mobile is Manufactured");
    }
    void newfeature1(){
        System.out.println("Camera Mobile with 5MG px");
    }
}
class android extends camera{
    android(){
        System.out.println("Android Mobile is Manufactured");
    }
    void newfeature(){
        System.out.println("Touch Screen Mobile is Manufactured");
    }
}

class prog{
    public static void main(String args[]){
        android a = new android();
        a.newfeature1();
        a.newfeature();
    }
}
```

	Expected	Got	

✓	`Basic Mobile is Manufactured`
	`Camera Mobile is Manufactured`
	`Android Mobile is Manufactured`
	`Camera Mobile with 5MG px`
	`Touch Screen Mobile is Manufactured`
		`Basic Mobile is Manufactured`
		`Camera Mobile is Manufactured`
		`Android Mobile is Manufactured`
		`Camera Mobile with 5MG px`
		`Touch Screen Mobile is Manufactured`
			✓

Passed all tests! ✓

Question **2**

Correct

Marked out of 5.00

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.

For example:

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

```
class ba{
    int bal;
    ba(int b){
        this.bal=b;
    }
    void deposit(int a){
        bal+=a;
    }
    void withdraw(int a){
        bal-=a;
    }
    int gb(){
        return bal;
    }
}
class sa extends ba{
    sa(int b){
        super(b);
    }
    void withdraw(int a){
        if((bal-a)<100){
            System.out.println("Minimum balance of $100 required!");
        }
        else{
            bal-=a;
        }
    }
}
public class hello{
    public static void main(String[] args){
        ba BA1234=new ba(500);
        sa SA1000=new sa(300);
        System.out.println("Create a Bank Account object (A/c No. BA1234) with initial balance of $500:");
        System.out.println("Deposit $1000 into account BA1234:");
        BA1234.deposit(1000);
        System.out.println("New balance after depositing $1000: $"+BA1234.gb()+".0");
        System.out.println("Withdraw $600 from account BA1234:");
        BA1234.withdraw(600);
        System.out.println("New balance after withdrawing $600: $"+BA1234.gb()+".0");
        System.out.println("Create a SavingsAccount object (A/c No. SA1000) with initial balance of $300:");
        System.out.println("Try to withdraw $250 from SA1000!");
        SA1000.withdraw(250);
        System.out.println("Balance after trying to withdraw $250: $"+SA1000.gb()+".0");
    }
}
```

	Expected	Got	
✓	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	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	✓

Passed all tests! ✓

Question **3**

Correct

Marked out of 5.00

create a class called College with attribute String name, constructor to initialize the name attribute , a method called Admitted(). Create a subclass called CSE that extends 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 %)

Reset answer

```
class College
{
protected   String collegeName;

public College(String collegeName) {
    this.collegeName=collegeName;
    }

public void admitted() {
    System.out.println("A student admitted in "+collegeName);
}
}
class Student extends College{

String studentName;
String department;

public Student(String collegeName, String studentName,String depart) {
    super(collegeName);
    this.studentName=studentName;
    this.department=depart;
}

public String toString(){
    return "CollegeName : "+collegeName+"\n"+"StudentName : "+studentName+"\n"+"Department : "+department;

}
}
class prog {
public static void main (String[] args) {
        Student  s1 = new Student("REC","Venkatesh","CSE");
        s1.admitted();                            // invoke the admitted() method
        System.out.println(s1.toString());
}
}
```

	Expected	Got	
✓	A student admitted in REC CollegeName : REC StudentName : Venkatesh Department : CSE	A student admitted in REC CollegeName : REC StudentName : Venkatesh Department : CSE	✓

Passed all tests! ✓

◄ Lab-05-MCQ

Jump to...

Is Palindrome Number? ►