Quiz Submissions - Quiz #7

✕

Hoang Phuong Truong (username: htruong26)

Attempt 1

Written: Mar 7, 2020 9:08 AM - Mar 7, 2020 9:33 AM

Submission View

You successfully submitted your quiz.

Question 1

1 / 1 point

Given that this code is valid

new X();

You can infer that X can be

✓ ○ a) Concrete class

○ b) Abstract class

○ c) Both

Question 2

0 / 1 point

Given that this code is valid

X thing;   // a variable declaration statement

You can infer that X could be

✗ ○ a) Concrete class

○ b) Abstract class

➡ ○ c) Both

Question 3

0 / 1 point

Given that this code is valid

public void dosomething(X thing) { ... }

You can infer that X could be

✗ ○ a) Concrete

○ b) Abstract

➡ ○ c) Both

Question 4 **0 / 1 point**

Given that this code is valid

public void doSomething(X thing){ ... }

Things that my be passed as an argument to doSomething() method may be of what type?

○ a) Type X only

➡ ○ b) Type X or subtype of X

✗ ○ c) Type X or supertype of X

○ d) Type X, subtype or supertype of X

Question 5 **1 / 1 point**

Given that this code is valid

X thing = new Y();

What can you infer about the relationship between X and Y?

✓ ○ a) Y is a subtype of X

○ b) X is a subtype of Y

○ c) Y is an object in X

Question 6 **0 / 2 points**

What’s the output of the following code

```
public abstract class MyAbstract{
    private int number;

    public MyAbstract(){
        number *=2;
        System.out.println("number is "+anotherMethod(3));
    }

    public  abstract int method();

    public abstract int anotherMethod(int value);

    public int getNumber(){
        return number;
    }

    public void setNumber(int input){
        number = input;
    }


public class MySuper extends MyAbstract{
    private int number;
    public MySuper() {
        number = 2;
        System.out.println(" number is "+method());
    }

    public int method() {
        number++;
        return (number * 5 + super.getNumber());
    }

    public int anotherMethod(int i) {
        return (method() + i);
    }

    public static void main(String[] args){
        MySuper s = new MySuper();

    }
```

number is 22
number is 19

The correct answer is not displayed for Written Response type questions.

▼ Hide Feedback

number is 8

number is 15

Question 7

3 / 3 points

What would the output of the following piece of code be?

```
  class Base {
    Base() {
      System.out.println("Message 1 : from the base class ");
    } }

  abstract class Derived1 extends Base
  {
    Derived1()  {
      System.out.println("Message 2 : from Derived1 class");
    } }


public class Derived2 extends Derived1 {
  public Derived2()  {
    System.out.println("Message 3 : from derived2 class");
  }

  public static void main(String args[]) {
    Derived2 d2 = new Derived2();
  } }
```

Message 1 : from the base class
Message 2: from Derived1 class
Message 3: from derived2 class

The correct answer is not displayed for Written Response type questions.

▼ Hide Feedback

Message 1 : from the base class

Message 2 : from the Derived1 class

Message 3 : from the Derived2 class

Question 8

2 / 2 points

State two reasons as to why one would make a class abstract?

avoid instantiation
it like a placeholder to put ideas

The correct answer is not displayed for Written Response type questions.

▼ Hide Feedback

To prevent direct instantiation of the class

To force all the subclasses to implement all the abstract methods to be concrete classes otherwise they will be abstract classes

Attempt Score:7 / 12

Overall Grade (highest attempt)**:**7 / 12

Done