



## **Institute of Software Engin**

Graduate Diploma in Software Engineering

Batch - GDSE69

Module - Object Oriented Programming

Name : Shimara Appuhami

Nic : 200362310434



## Assignment 04

01. Which of the statements below are true?

- A. Abstract classes cannot be used to instantiate objects, because they're incomplete.
- B. An abstract class must contain at least one abstract method.
- C. Constructors and static methods can be declared abstract.
- D. Abstract methods do not provide implementations.
- E. Abstract classes sometimes constitute several levels of a hierarchy.
- F. Classes that can be used to instantiate objects are called concrete classes. They provide implementations of every method they declare.
- G. A class that contains any abstract methods must be declared as an abstract

class. Each concrete subclass must provide implementations of each of the superclass's abstract methods.

////////////////////////////////////  
////////

02. Which of the follow are true statements?

- A. Any named class can be declared abstract.
- B. An incompletely implemented class must be declared abstract.
- C. An abstract class can be instantiated.
- D. An abstract class is implicitly final.
- E. An abstract class must declare at least one abstract method.
- F. An abstract class cannot extend a concrete class.

////////////////////////////////////  
////////

03. Which of the statements below are true?

- A. A superclass reference variable can be used to invoke only methods declared in the superclass.
- B. Operator 'instanceof' determines if an object has the 'has a' relationship with a specific type.
- C. The 'is-a' relationship applies only between the subclass and its super classes, not vice versa.
- D. Most method calls are resolved at execution time, based on the type of the object being manipulated.

This process is known as dynamic binding or late binding.

////////////////////////////////////  
////////

04. Given Code:

```
abstract class Base{  
    abstract public void myfunc();  
    public void another(){  
        System.out.println("Another method");  
    }  
}
```

```

class Abs extends Base{
public static void main(String
argv[]){ Abs a = new Abs();
a.amethod();
}
public void
myfunc(){ System.out.println(" My func");
}
public void
amethod(){ myfunc
();
}
}

```

What will happen when you attempt to compile and run this code?

- A. The code will compile and run, printing out the words "My Func"
- B. The compiler will complain that the Base class has non abstract methods
- C. The code will compile but complain at run time that the Base class has non abstract methods
- D. The compiler will complain that the method 'myfunc' in the base class has no body, nobody at all to lose it.

////////////////////////////////////  
 //////////////////////////////////

05. Which of the statements below are true?

- A. An interface specifies what operations are allowed but not how they're performed.
- B. A Java interface describes a set of methods that can be called on an object.
- C. An interface cannot declare variables or constants.
- D. All objects of a class that implement multiple interfaces have the 'is-a' relationship with each implemented interface type.

////////////////////////////////////  
 //////////////////////////////////

06. Which of the following methods can be legally inserted at line 2

```

abstract class Customer{
//Line 2

```

}

- A. void search(String name){}
- B. abstract void search(String name){}
- C. **abstract void search(String name);**
- D. private abstract void search(String name);
- E. static abstract void search(String name);
- F. private static abstract void search(String name);
- G. **public abstract void search(String name);**

////////////////////////////////////  
////////

07. Compare and contrast abstract classes and interfaces. Why would you use an abstract class? Why would you use an interface?

- We can run an abstract class if it has a main method but we can't run an interface because they can't have a main method implementation. Interfaces are used to define contracts for the subclasses whereas abstract classes also define contracts but it can provide other methods implementations for subclasses to use.

////////////////////////////////////  
////////

08. Given:

```
interface A {  
void m1(); // 1  
public void m2(); // 2  
protected void m3(); // 3  
private void m4(); // 4  
}
```

What is the result of attempting to compile the code?

- A. Compiler error at line 1.
- B. Compiler error at line 2.
- C. **Compiler error at line 3**
- D. **Compiler error at line 4.**

////////////////////////////////////  
////////

09. Which of the following statements are true?

- A. All of the variables in an interface are implicitly static .
- B. All of the variables in an interface are implicitly final .
- C. All of the methods in an interface are implicitly abstract.
- D. A method in an interface can access class level variables .

////////////////////////////////////  
////////

10. Which of the following are legal declarations for nonnested classes and interfaces?

- A. final abstract class Test
- B. public static interface Test{}
- C. final public class Test{}
- D. protected abstract class Test{}
- E. protected interface Test{}
- F. abstract public class Test{}

////////////////////////////////////  
////////

11. Which of the following statements are true?

- A. An interface can only contain method and not variables .
  - B. Interfaces cannot have constructors.
  - C. class may extend only one other class and implement only one interface. D.
- Interfaces are the Java approach to addressing its lack of multiple inheritance, but require.
- E. implementing classes to create the functionality of the Interfaces.

////////////////////////////////////  
////////

12. Given Code:

- 1. interface I10 {
- 2. String name = "I10"; 3. String s10 = "I10.s10";
- 4. }
- 5. interface I20 {

```

6. String name = "I20"; 7. String s20 = "I20.s20";
8. }
9. class C20 implements I10, I20 {
10. public static void main(String[] args) {
11. System.out.print(I10.s10+",");
12. System.out.print(I20.s20+",");
13. System.out.print(I20.name);
14. }
15. }

```

What is the result of attempting to compile and run the program?

- A. Prints:I10.s10,I20.s20,I10
- B. **Prints:I10.s10,I20.s20,I20**
- C. Prints: I10.s10,I20.s20,
- D. Prints: I10.s10,I20.s20,null
- E. Compiler error at line 9
- F. Compiler error at line 11.
- G. Compiler error at line 12.
- H. Compiler error at line 13.

```

////////////////////////////////////
////////

```

\*\*\*\*\*