

Started on	Saturday, 27 April 2024, 12:13 PM
State	Finished
Completed on	Saturday, 27 April 2024, 12:18 PM
Time taken	4 mins 31 secs
Marks	2.25/6.00
Grade	37.50 out of 100.00

Question

1

Complete

Mark 0.00 out of 1.00

```
interface WithDefinitionsInter {
    default void definedMeth() {
        System.out.println("inside interface");
    }
}

class WithDefinitionsImpl implements WithDefinitionsInter {
    public void definedMeth() {
        super.definedMeth();
        System.out.println("inside class");
    }
}

public class QuizDef {
    public static void main(String par[]) {
        WithDefinitionsInter withDef = new WithDefinitionsImpl();
        withDef.definedMeth();
    }
}
```

What will happen to the above code when compiled?

Select one:

- ☐ a. The code will not get compiled because the method definedMeth() is undefined in Object class
- ☐ b. The code will not get compiled because the interface holds method definition
- ☒ c. The code will get compiled and executed successfully
- ☐ d. The code will not get compiled because the interface does not even have a single abstract method

Question

2

Complete

Mark 0.00 out of 1.00

```
1: interface HasExoskeleton {  
2: abstract int getNumberOfSections();  
3: }  
4: abstract class Insect implements HasExoskeleton {  
5: abstract int getNumberOfLegs();  
6: }  
7: public class Beetle extends Insect {  
8: int getNumberOfLegs() { return 6; }  
9: }
```

Select one:

- ☒ **a. The code will not compile because of line 2.**
- ☐ **b. The code will not compile because of line 7.**
- ☐ **c. It compiles and runs without issue.**
- ☐ **d. The code will not compile because of line 4.**
- ☐ **e. It compiles but throws an exception at runtime.**

Question

3

Complete

Mark 0.50 out of 1.00

```
public interface Deduction {
    static void deduct() {
        System.out.println("deduct");
    }
}

@SuppressWarnings("")
public class Customer implements Deduction {
    public void serviceCharge() {
        //deduction functionality being invoked
        deduct();
    }
}

public class TechSol {
    public static void main(String[] args) {
        new customer().serviceCharge();
    }
}
```

What will happen to the above code when compile and execute? (Select two valid options)

Select one or more:

- ☒ a. The code will not get compiled because the static method of interface is not accessed using interface name
- ☐ b. Will lead to run time ambiguity as the static method is not accessed using interface name
- ☐ c. Will get executed successfully leaving the output “deduct”
- ☐ d. The code will not get compiled as the method deduct() is undefined for the Customer class

Question

4

Complete

Mark 0.00 out of 1.00

```
public interface Deduction {
    static void deduct() {
        System.out.println("deduct");
    }
}

public class Customer implements Deduction {
    public static void deduct() {
        System.out.println("deduction for customer");
    }
}

public class TechSol {
    public static void main(String[] args) {
        Deduction deduction = new Customer();
        deduction.deduct();
    }
}
```

What will happen when the code is subjected to compilation and execution?

Select one:

- ☐ a. The code will not get compiled as the static method of interface is not accessed using interface name
- ☐ b. Will get executed successfully leaving the output “deduct”
- ☒ c. Will lead to run time ambiguity as the interface’s static method is overridden
- ☐ d. Will get executed successfully leaving the output “deduction for customer”

Question

5

Complete

Mark 0.75 out of 1.00

```
public interface CanHop {}

public class Frog implements CanHop {
    public static void main(String[] args) {
        _____ frog = new TurtleFrog();
    }
}

public class BrazilianHornedFrog extends Frog {}

public class TurtleFrog extends Frog {}
```

Select one or more:

- ☐ a. BrazilianHornedFrog
- ☐ b. Object
- ☒ c. Frog
- ☐ d. Long
- ☒ e. TurtleFrog
- ☒ f. CanHop

Question

6

Complete

Mark 1.00 out of 1.00

What is expected when the following code gets compiled and executed?

```
interface WithDefinitionsInter {  
    default void definedMeth() {  
        System.out.println("inside interface");  
    }  
}  
  
class WithDefinitionsImpl implements WithDefinitionsInter {  
    public void definedMeth() {  
        System.out.println("inside class");  
    }  
}  
  
public class QuizDef {  
    public static void main(String par[]) {  
        WithDefinitionsInter withDef = new WithDefinitionsImpl();  
        withDef.definedMeth();  
    }  
}
```

Select one:

- ☐ a. No successful compilation because the interface does not even have a single abstract method
- ☒ b. The code will be executed successfully. And, the execution result will be, inside class
- ☐ c. No successful compilation because the interface holds method definition
- ☐ d. The code will print, inside interface and inside class as a result of successful execution



logo icon

x

