Question 1: Data types

If you wanted to store $\pi$ in a data type, which of these would be best?

A. int
B. float
C. char
D. long
Question 1: Data types

If you wanted to store $\pi$ in a data type, which of these would be best?

A. int
B. float
C. char
D. long
Question 2: Modifiers

Which modifier goes before a variable when you want the entire class to share only one copy?

A package
B class
C static
D void
Question 2: Modifiers

Which modifier goes before a variable when you want the entire class to share only one copy?

A  package
B  class
C  static
D  void
Question 3: Protected

A protected variable or method is visible to:

A only the class in which it is declared.
B the class in which it is declared and any parent classes of that class.
C the class in which it is declared and any classes that extend that class.
D only classes in the same package as the class in which it is declared.
Question 3: Protected

A protected variable or method is visible to:

A only the class in which it is declared.
B the class in which it is declared and any parent classes of that class.
C the class in which it is declared and any classes that extend that class.
D only classes in the same package as the class in which it is declared.
Question 4: Interface

Which class implements the interface?

```java
public interface MyInterface {
    void doSomething(int i, double x);
    int doSomethingElse(String s);
}
```

A

```java
public class MyClass implements MyInterface {
    public int doSomething(int i, double x) {
        return 0;
    }
    public void doSomethingElse(String s) {
    }
}
```
Question 4: Interface

Which class implements the interface?

```java
public interface MyInterface {
    void doSomething(int i, double x);
    int doSomethingElse(String s);
}
```

A The method signatures don’t match.

```java
public class MyClass implements MyInterface {
    public int doSomething(int i, double x) {
        return 0;
    }
    public void doSomethingElse(String s) {
    }
}
```
Question 4: Interface

Which class implements the interface?

```java
public interface MyInterface {
    void doSomething(int i, double x);
    int doSomethingElse(String s);
}
```

B

```java
public class MyClass {
    public void doSomething(int i, double x) {
    }
    public int doSomethingElse(String s) {
        return 0;
    }
}
```
Question 4: Interface

Which class implements the interface?

```java
public interface MyInterface {
    void doSomething(int i, double x);
    int doSomethingElse(String s);
}

B Not using implements

public class MyClass {
    public void doSomething(int i, double x) {
    }
    public int doSomethingElse(String s) {
        return 0;
    }
}
```
Question 4: Interface

Which class implements the interface?

```java
public interface MyInterface {
    void doSomething(int i, double x);
    int doSomethingElse(String s);
}
```

C

```java
public class MyClass implements MyInterface {
    public void doSomething(int i, double x) {
    }
    public int doSomethingElse(String s) {
        return 0;
    }
}
```
Question 4: Interface

Which class implements the interface?

```java
public interface MyInterface {
    void doSomething(int i, double x);
    int doSomethingElse(String s);
}
```

C Correct!

```java
public class MyClass implements MyInterface {
    public void doSomething(int i, double x) {
    }
    public int doSomethingElse(String s) {
        return 0;
    }
}
```
Question 4: Interface

Which class implements the interface?

```java
public interface MyInterface {
    void doSomething(int i, double x);
    int doSomethingElse(String s);
}

public class MyClass extends MyInterface {
    public void doSomething(int i, double x) {
    }
    public int doSomethingElse(String s) {
        return 0;
    }
}
```
Question 4: Interface

Which class implements the interface?

public interface MyInterface {
    void doSomething(int i, double x);
    int doSomethingElse(String s);
}

D Using extends instead of implements

public class MyClass extends MyInterface {
    public void doSomething(int i, double x) {
    }
    public int doSomethingElse(String s) {
        return 0;
    }
}