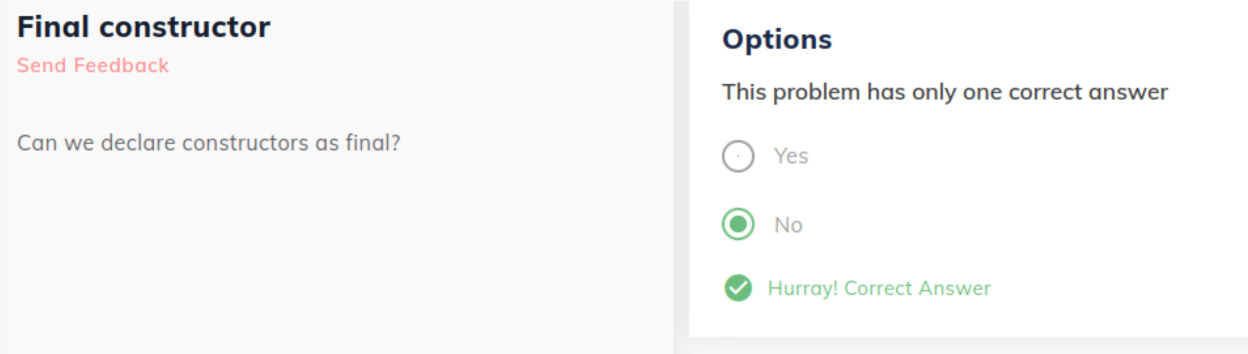
OOPS-3



Final constructor

Send Feedback

Can we declare constructors as final?

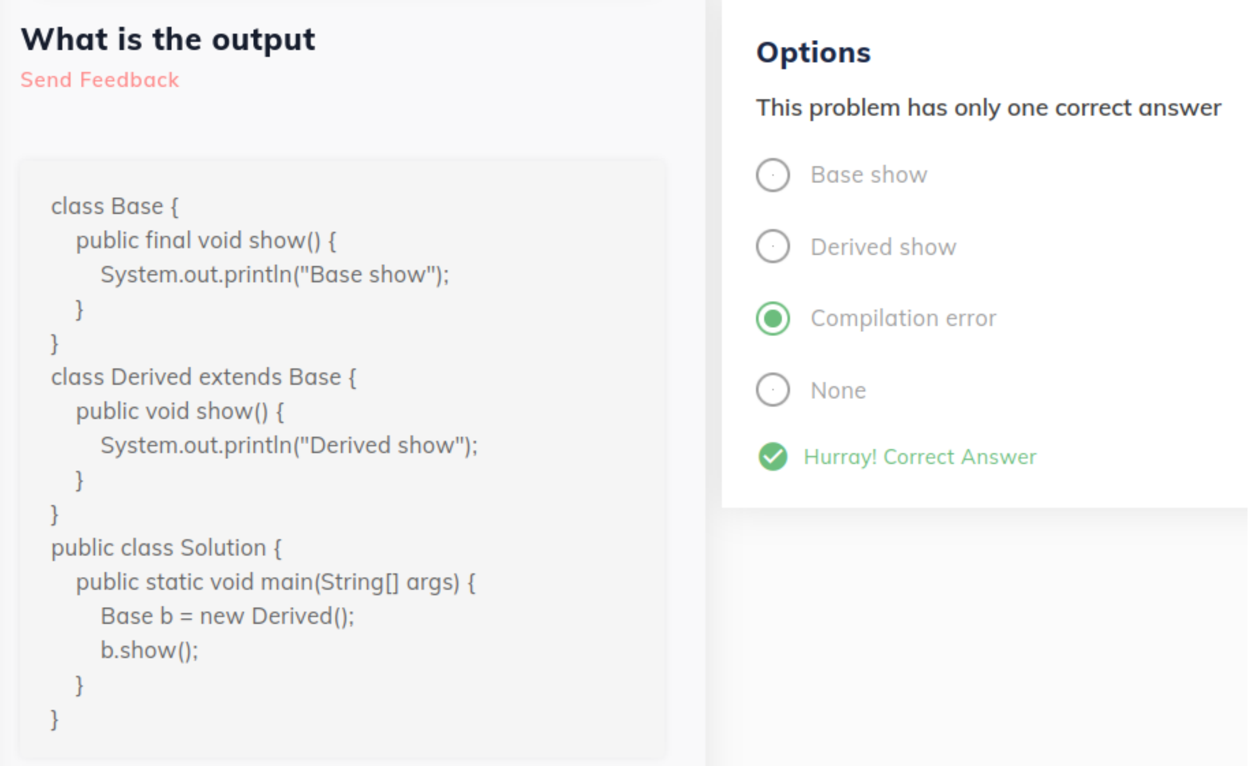
Options

This problem has only one correct answer

○ Yes

● No

✓ Hurray! Correct Answer



What is the output

Send Feedback

```
class Base {
    public final void show() {
        System.out.println("Base show");
    }
}
class Derived extends Base {
    public void show() {
        System.out.println("Derived show");
    }
}
public class Solution {
    public static void main(String[] args) {
        Base b = new Derived();
        b.show();
    }
}
```

Options

This problem has only one correct answer

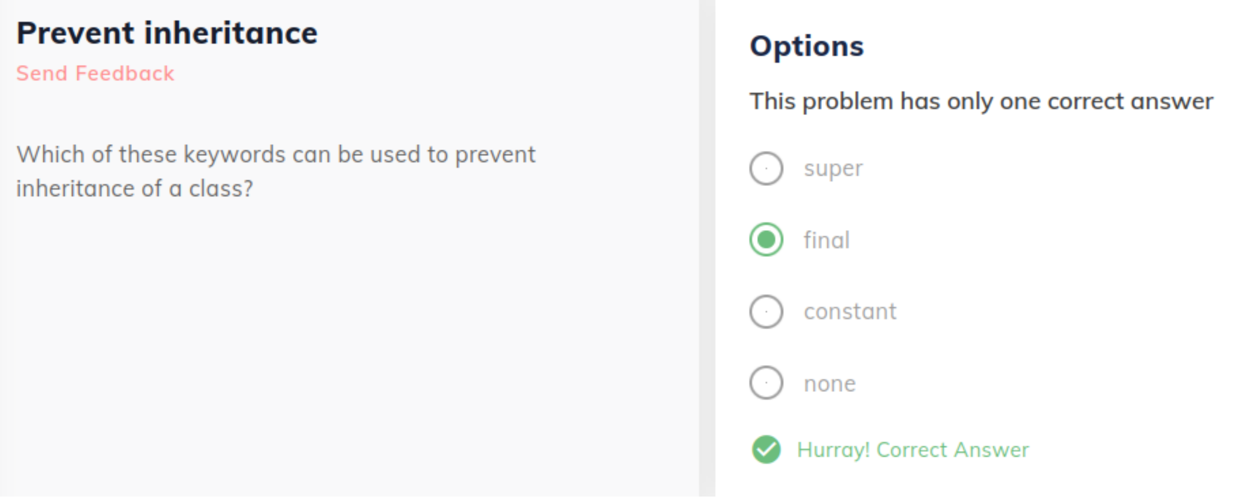
○ Base show

○ Derived show

● Compilation error

○ None

✓ Hurray! Correct Answer



Prevent inheritance

Send Feedback

Which of these keywords can be used to prevent inheritance of a class?

Options

This problem has only one correct answer

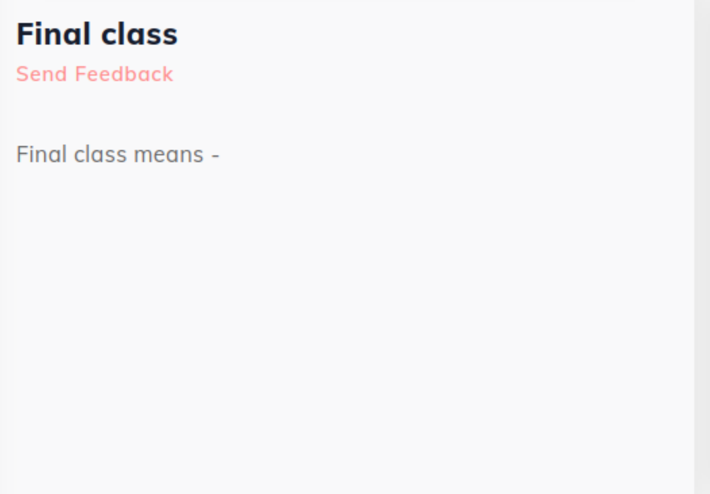
○ super

● final

○ constant

○ none

✓ Hurray! Correct Answer



Final class

Send Feedback

Final class means -

Options

This problem has only one correct answer

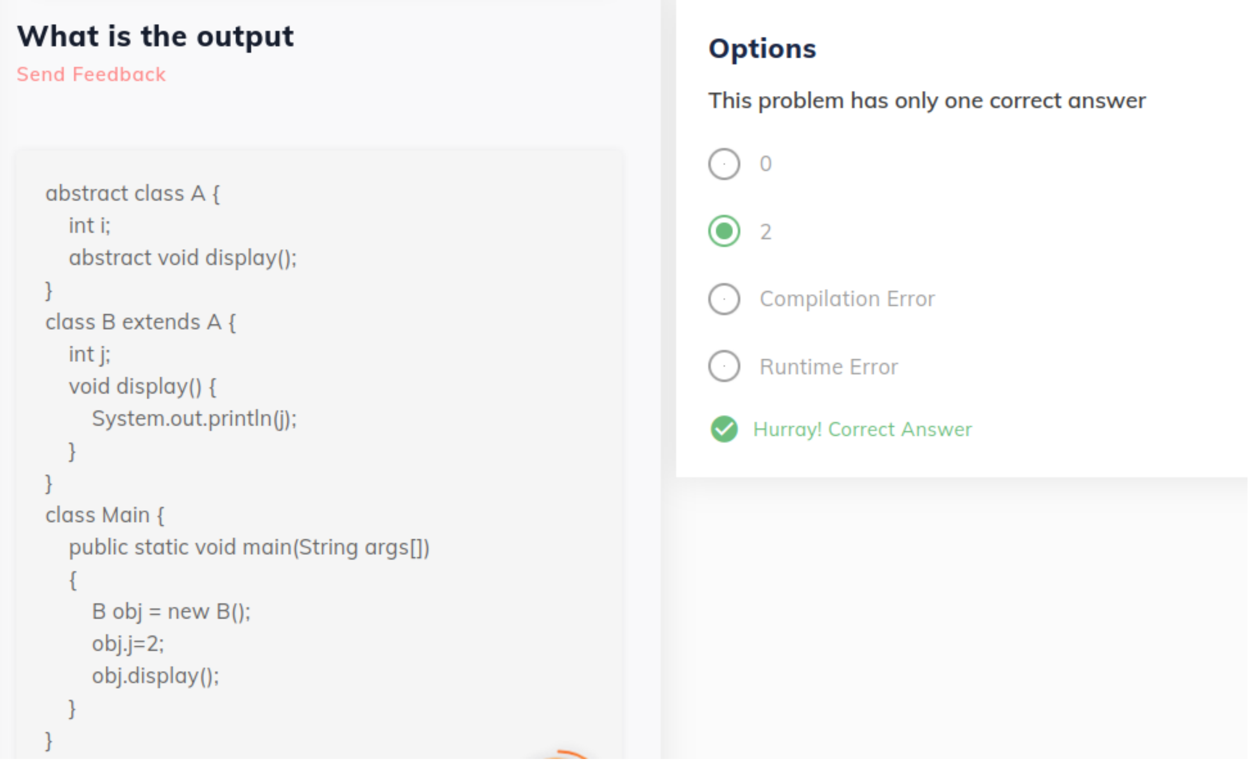
○ It cannot have non-final members

● It cannot be extended

○ Its objects cannot be created

○ All

✓ Hurray! Correct Answer



What is the output

Send Feedback

```
abstract class A {
  int i;
  abstract void display();
}
class B extends A {
  int j;
  void display() {
    System.out.println(j);
  }
}
class Main {
  public static void main(String args[])
  {
    B obj = new B();
    obj.j=2;
    obj.display();
  }
}
```

Options

This problem has only one correct answer

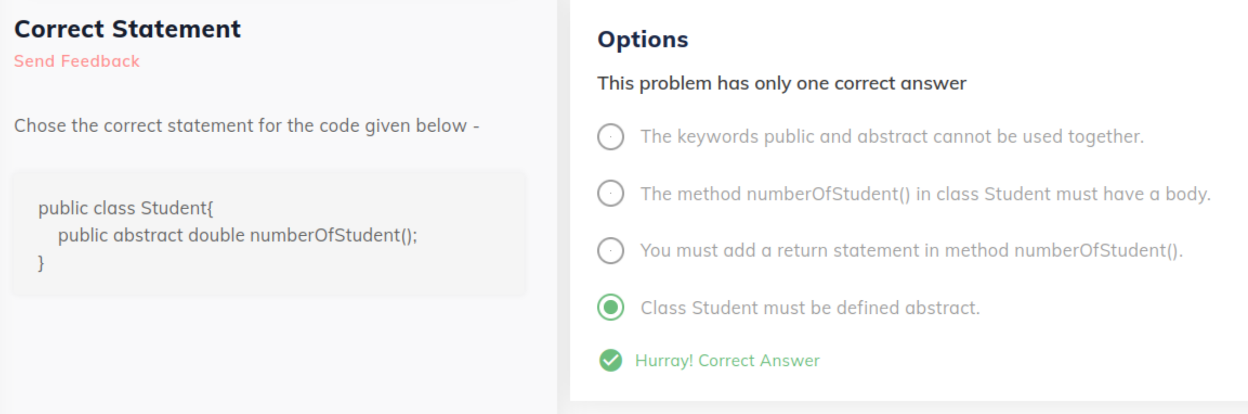
○ 0

● 2

○ Compilation Error

○ Runtime Error

✓ Hurray! Correct Answer



Correct Statement

Send Feedback

Chose the correct statement for the code given below -

```
public class Student{
    public abstract double numberOfStudent();
}
```

Options

This problem has only one correct answer

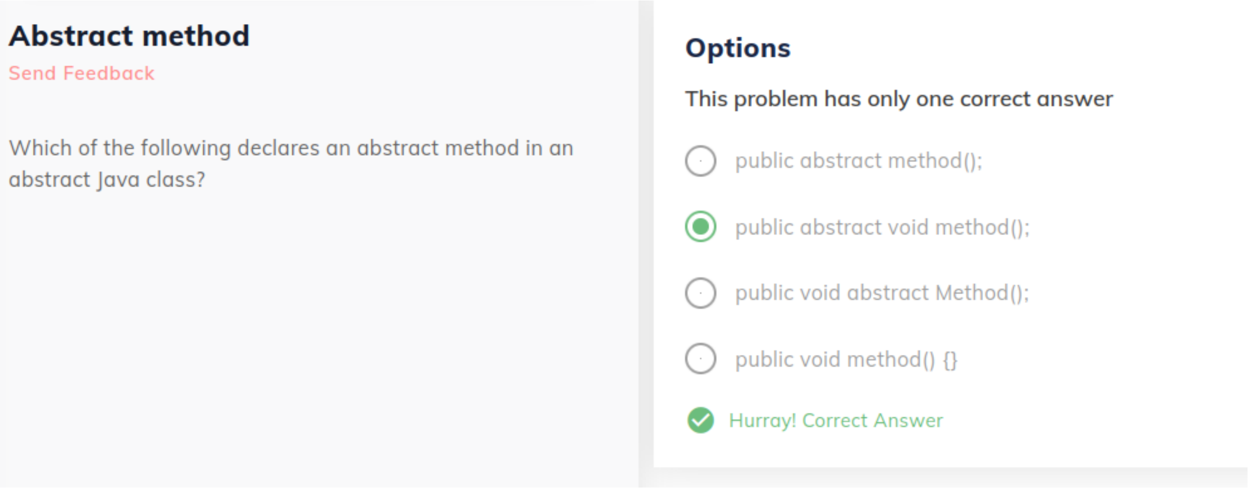
○ The keywords public and abstract cannot be used together.

○ The method numberOfStudent() in class Student must have a body.

○ You must add a return statement in method numberOfStudent().

● Class Student must be defined abstract.

✓ Hurray! Correct Answer



Abstract method

Send Feedback

Which of the following declares an abstract method in an abstract Java class?

Options

This problem has only one correct answer

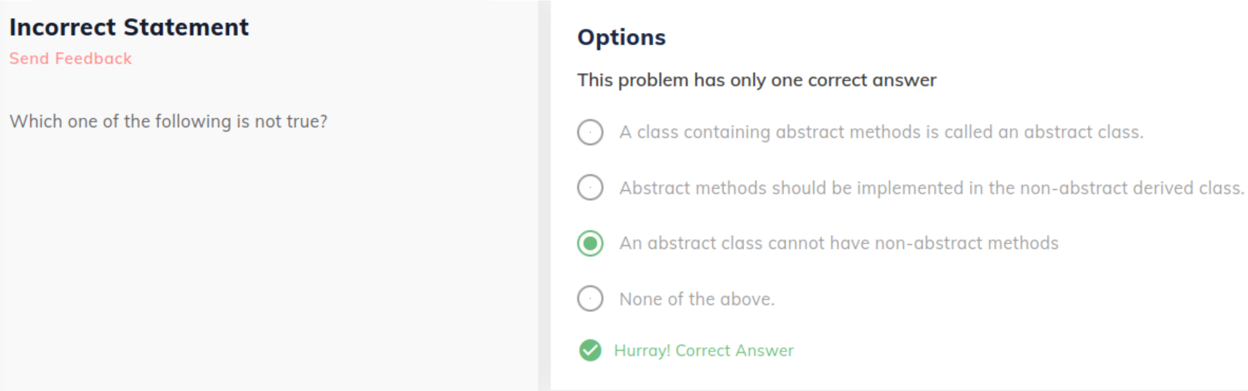
○ public abstract method();

● public abstract void method();

○ public void abstract Method();

○ public void method() {}

✓ Hurray! Correct Answer



Incorrect Statement

Send Feedback

Which one of the following is not true?

Options

This problem has only one correct answer

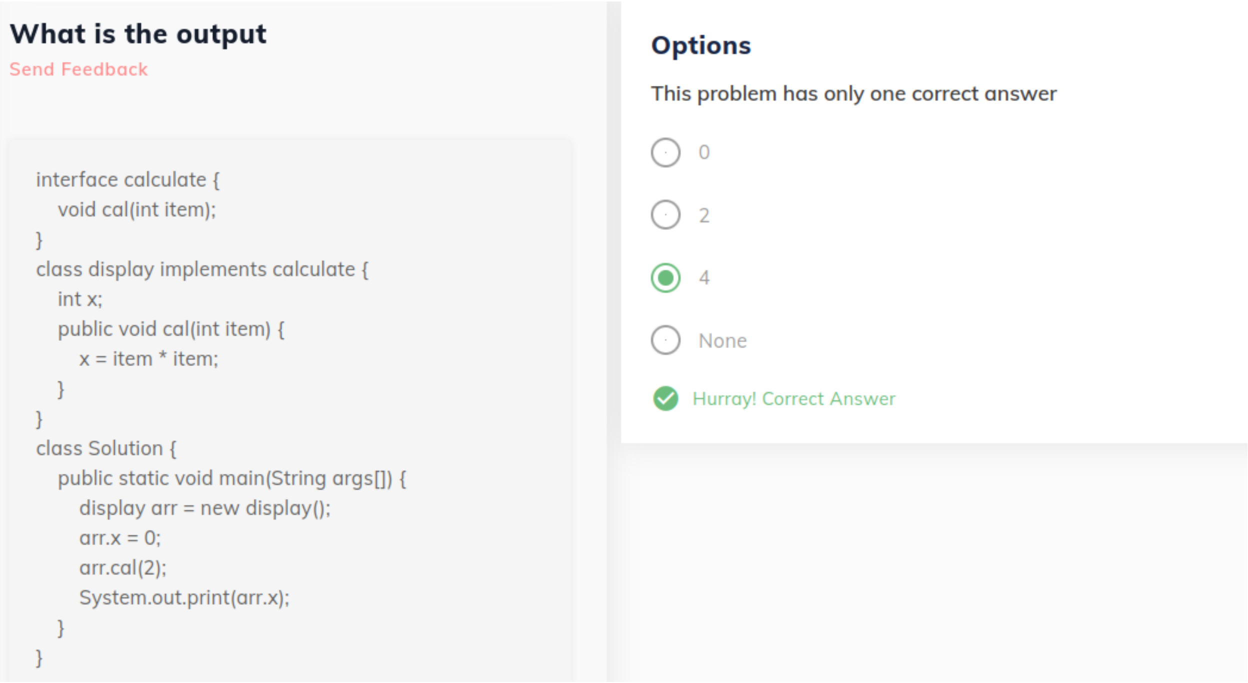
○ A class containing abstract methods is called an abstract class.

○ Abstract methods should be implemented in the non-abstract derived class.

● An abstract class cannot have non-abstract methods

○ None of the above.

✓ Hurray! Correct Answer



What is the output

Send Feedback

```
interface calculate {
    void cal(int item);
}
class display implements calculate {
    int x;
    public void cal(int item) {
        x = item * item;
    }
}
class Solution {
    public static void main(String args[]) {
        display arr = new display();
        arr.x = 0;
        arr.cal(2);
        System.out.print(arr.x);
    }
}
```

Options

This problem has only one correct answer

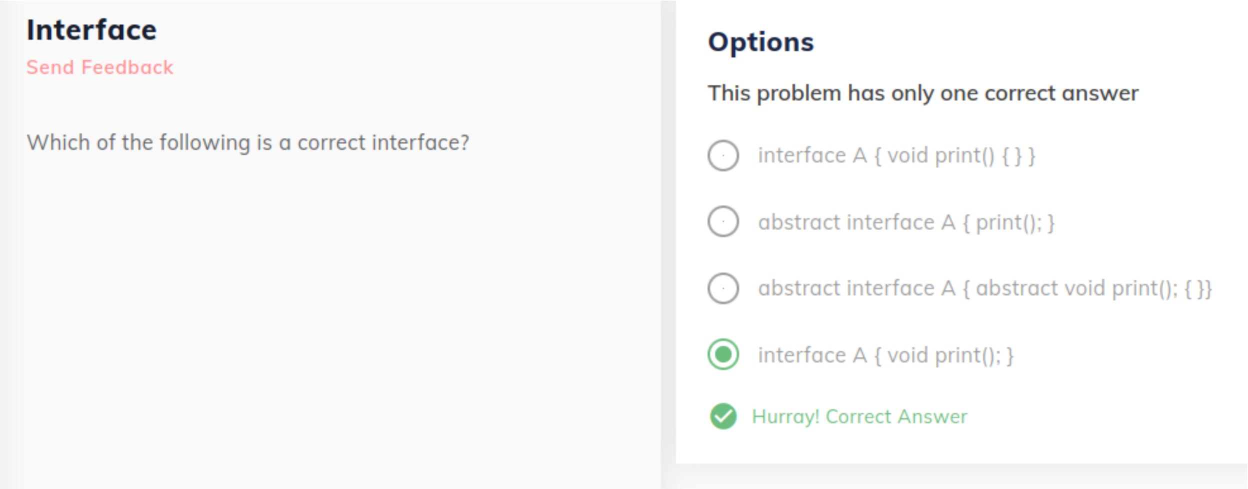
○ 0

○ 2

● 4

○ None

✓ Hurray! Correct Answer



Interface

Send Feedback

Which of the following is a correct interface?

Options

This problem has only one correct answer

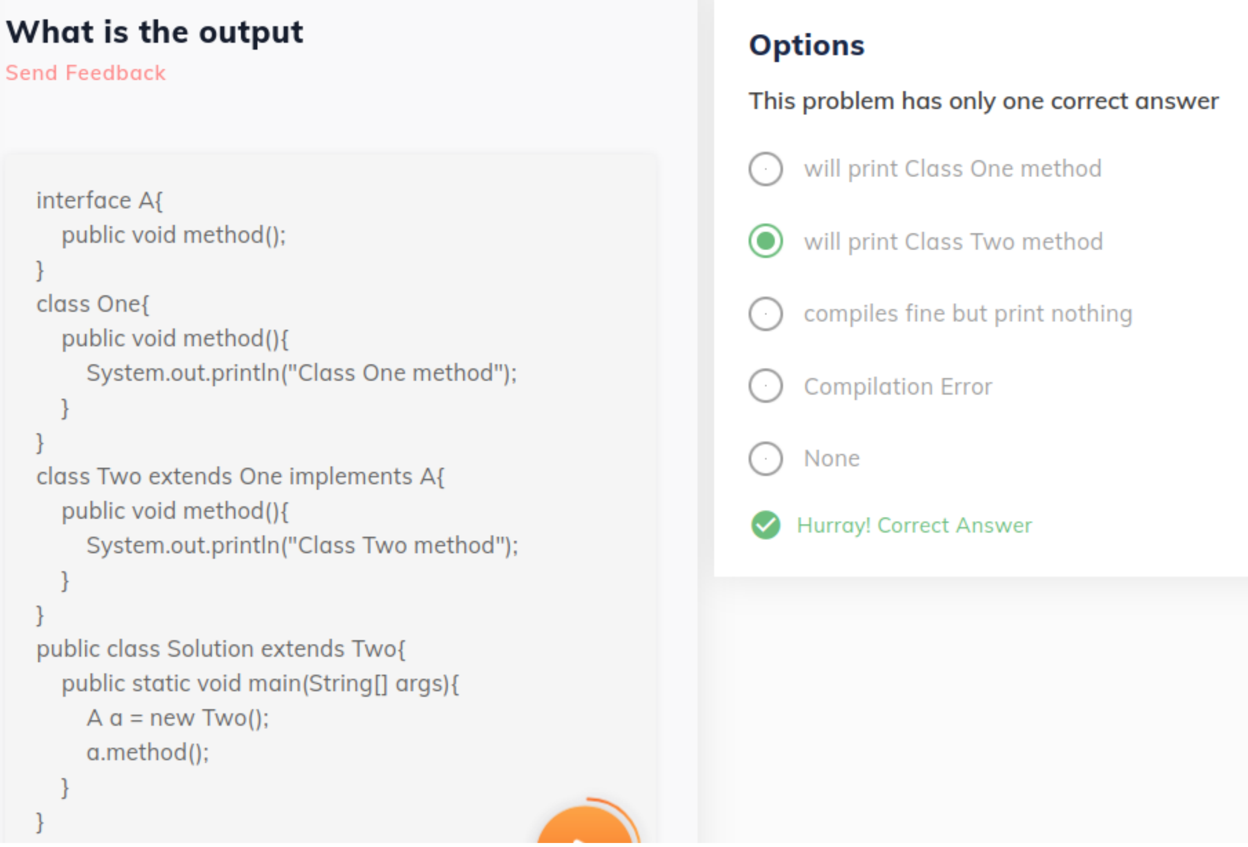
○ interface A { void print() { } }

○ abstract interface A { print(); }

○ abstract interface A { abstract void print(); { }}

● interface A { void print(); }

✓ Hurray! Correct Answer



What is the output

Send Feedback

```
interface A{
   public void method();
}
class One{
   public void method(){
      System.out.println("Class One method");
   }
}
class Two extends One implements A{
   public void method(){
      System.out.println("Class Two method");
   }
}
public class Solution extends Two{
   public static void main(String[] args){
      A a = new Two();
      a.method();
   }
}
```

Options

This problem has only one correct answer

○ will print Class One method

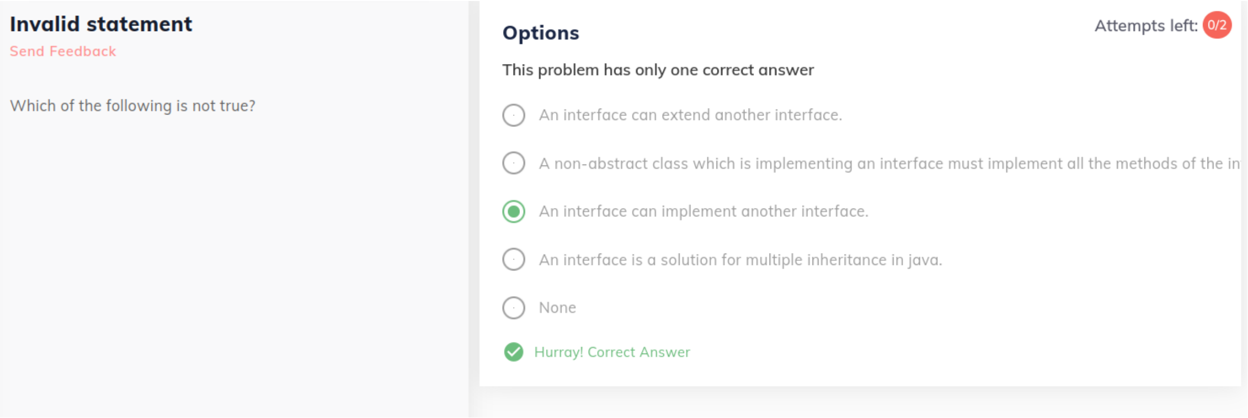
● will print Class Two method

○ compiles fine but print nothing

○ Compilation Error

○ None

✓ Hurray! Correct Answer



Invalid statement

Send Feedback

Which of the following is not true?

Options

Attempts left: 0/2

This problem has only one correct answer

○ An interface can extend another interface.

○ A non-abstract class which is implementing an interface must implement all the methods of the in

● An interface can implement another interface.

○ An interface is a solution for multiple inheritance in java.

○ None

✓ Hurray! Correct Answer