

The Java™ Tutorials

[« Previous](#) • [TOC](#)

[Home Page](#) > [Learning the Java Language](#) > [Interfaces and Inheritance](#)

The Java Tutorials have been written for JDK 8. Examples and practices described in this page don't take advantage of improvements introduced in later releases.

Answers to Questions and Exercises: Interfaces

Questions

Question 1: What methods would a class that implements the `java.lang.CharSequence` interface have to implement?

Answer 1: `charAt`, `length`, `subSequence`, and `toString`.

Question 2: What is wrong with the following interface?

```
public interface SomethingIsWrong {  
    void aMethod(int aValue) {  
        System.out.println("Hi Mom");  
    }  
}
```

Answer 2: It has a method implementation in it. Only default and static methods have implementations.

Question 3: Fix the interface in Question 2.

Answer 3:

```
public interface SomethingIsWrong {  
    void aMethod(int aValue);  
}
```

```
}
```

Alternatively, you can define `aMethod` as a default method:

```
public interface SomethingIsWrong {  
    default void aMethod(int aValue) {  
        System.out.println("Hi Mom");  
    }  
}
```

Question 4: Is the following interface valid?

```
public interface Marker {  
}
```

Answer 4: Yes. Methods are not required. Empty interfaces can be used as types and to mark classes without requiring any particular method implementations. For an example of a useful empty interface, see `java.io.Serializable`.

Exercises

Exercise 1: Write a class that implements the `CharSequence` interface found in the `java.lang` package. Your implementation should return the string backwards. Select one of the sentences from this book to use as the data. Write a small `main` method to test your class; make sure to call all four methods.

Answer 1: See [CharSequenceDemo.java](#)

Exercise 2: Suppose that you have written a time server, which periodically notifies its clients of the current date and time. Write an interface that the server could use to enforce a particular protocol on its clients.

Answer 2: See [TimeClient.java](#).

[« Previous](#) • [TOC](#)

[About Oracle](#) | [Contact Us](#) | [Legal Notices](#) | [Terms of Use](#) | [Your Privacy Rights](#)

Copyright © 1995, 2017 Oracle and/or its affiliates. All rights reserved.