

Java Interfaces

Spoken Tutorial Project

<http://spoken-tutorial.org>

National Mission on Education through ICT

<http://sakshat.ac.in>

Narration: Priya K

Script: Joms Antony, Amal Jyothi College of Engineering



23 December 2015



Learning Objectives

We will learn about



Learning Objectives

We will learn about

- **Creating an Interface**



Learning Objectives

We will learn about

- Creating an Interface
- **Creating an implementation class**



Learning Objectives

We will learn about

- Creating an Interface
- Creating an implementation class
- **Usage of interfaces**



Software Requirements



Software Requirements

- **Ubuntu 12.04**



Software Requirements

- Ubuntu 12.04
- **JDK 1.7**



Software Requirements

- Ubuntu 12.04
- JDK 1.7
- Eclipse 4.3.1



Prerequisite



Prerequisite

- **Basic knowledge of Java and Eclipse IDE**



Prerequisite

- Basic knowledge of **Java** and **Eclipse IDE**
- **Knowledge of Subclassing and Abstract classes**



Prerequisite

- Basic knowledge of **Java** and **Eclipse IDE**
- Knowledge of **Subclassing** and **Abstract classes**
- **For relevant tutorials, please visit**
<http://www.spoken-tutorial.org>



What is an Interface?



What is an Interface?

- **An interface contains a set of abstract methods and static data members**



What is an Interface?

- An interface contains a set of **abstract methods** and **static data members**
- It defines the signatures of a set of methods, without the body



What is an Interface?

- An interface contains a set of **abstract methods** and **static data members**
- It defines the signatures of a set of methods, without the body
- It is declared using the **interface** keyword



Implementing an Interface



Implementing an Interface

- A class can implement an interface using the **implements** keyword



Implementing an Interface

- A class can implement an interface using the **implements** keyword
- A class can implement more than one interface



Implementing an Interface

- A class can implement an interface using the **implements** keyword
- A class can implement more than one interface
- **All the abstract methods defined in the interface must be implemented in such classes**



Interfaces Vs Abstract Classes



Interfaces Vs Abstract Classes

- **All the methods in an interface should be abstract**



Interfaces Vs Abstract Classes

- All the methods in an interface should be **abstract**
- **An interface should not contain :**



Interfaces Vs Abstract Classes

- All the methods in an interface should be **abstract**
- An interface should not contain :
 - **constructors**



Interfaces Vs Abstract Classes

- All the methods in an interface should be **abstract**
- An interface should not contain :
 - constructors
 - **concrete methods**



Interfaces Vs Abstract Classes

- All the methods in an interface should be **abstract**
- An interface should not contain :
 - constructors
 - concrete methods
 - **static methods**



Interfaces Vs Abstract Classes

- All the methods in an interface should be **abstract**
- An interface should not contain :
 - constructors
 - concrete methods
 - static methods
 - **main method**



Interfaces Vs Abstract Classes



Interfaces Vs Abstract Classes

- But an abstract class can have all these inside it



Interfaces Vs Abstract Classes

- But an **abstract class** can have all these inside it
- **All the variables in an interface should be static and final**



Interfaces Vs Abstract Classes

- But an **abstract class** can have all these inside it
- All the variables in an interface should be **static** and **final**
- **No such restrictions for an abstract class**



Interface

Example:

```
public interface Animal {  
    String type1="Mammal";  
    String type2="Reptiles";  
    void talk();  
    void see();  
    void move();  
}
```



Implementation Class



Implementation Class

Example:

```
class Human implements Animal{  
    // other variables and methods  
  
    //implementation for talk()  
    //implementation for see()  
    //implementation for move()  
}
```



Implementing multiple interfaces



Implementing multiple interfaces

Example:

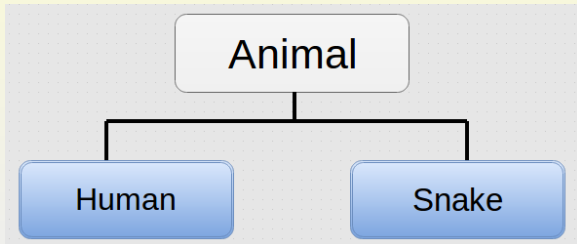
```
class Human implements Animal, Action{  
    // other variables and methods  
  
    //implementation for all the abstract  
    //methods in Animal and Action interface  
}
```



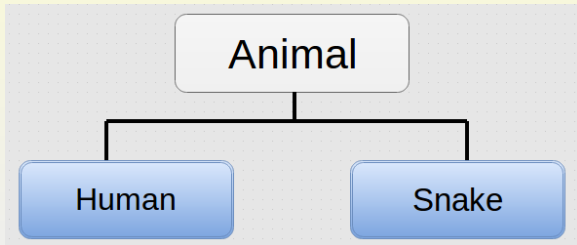
Usage of an Interface - Example



Usage of an Interface - Example



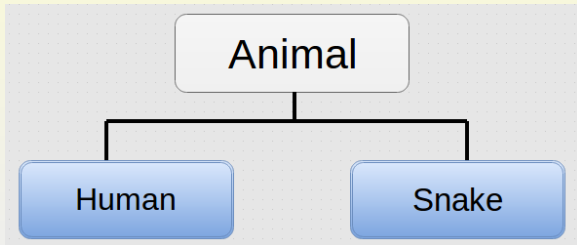
Usage of an Interface - Example



`talk()`, `see()`, `move()`



Usage of an Interface - Example



talk(), see(), move()

talk(), see(), move()



Summary

- **Creating an Interface**
- **Creating an implementation class**
- **Usage of interfaces**



Assignment

- Create an interface **Vehicle** which contains the methods **brake()** and **run()**
- Create another interface **Fuel** which contains the following methods
 - **fill(String type, int quantity)**
 - **pay(int quantity, int price)**



Assignment (cont.)

- Create a subclass **Car** which implements both the interfaces **Vehicle** and **Fuel**
 - **brake** method should print "**Car Applies Power brake**"
 - **run** method should print "**Car is running on 4 wheels**"
- **fill()** method can print the type and quantity of the fuel filled



Example: 10 Litres of Petrol



Assignment (cont.)

- **pay()** method can be used to print the price to be paid

Example: Pay Rs.640

- Create another subclass **Bike** which again implements both the interfaces **Vehicle** and **Fuel**
 - **brake** method should print "**Bike Applies hand brake**"
 - **run** method should print "**Bike is running on 2 wheels**"



Assignment (cont.)

- Implement the **fill()** and **pay()** methods as explained earlier
- Finally create a **Demo** class containing the **main method** to verify the results



About the Spoken Tutorial Project

- Watch the video available at http://spoken-tutorial.org/What_is_a_Spoken_Tutorial
- It summarises the Spoken Tutorial project
- If you do not have good bandwidth, you can download and watch it



About the Spoken Tutorial Project

- Watch the video available at http://spoken-tutorial.org/What_is_a_Spoken_Tutorial
- It summarises the Spoken Tutorial project
- If you do not have good bandwidth, you can download and watch it



Spoken Tutorial Workshops

The Spoken Tutorial Project Team

- Conducts workshops using spoken tutorials
- Gives certificates to those who pass an online test
- For more details, please write to contact@spoken-tutorial.org



Acknowledgements

- Spoken Tutorial Project is a part of the Talk to a Teacher project
- It is supported by the National Mission on Education through ICT, MHRD, Government of India
- More information on this Mission is available at <http://spoken-tutorial.org/NMEICT-Intro>



About the contributor

- This script has been contributed by:
Dept. of Information Technology
Amal Jyothi College of Engineering
<http://itamaljyothi.in>
- Thanks for joining

