8

- (interface)
-
-
-
- instanceof

- (interface)

- .

- ,
.

```
[public] interface
{
    //
    //
}
```

: class interface ,
.
.

NumberInput.java(1)

```
public interface NumberInput
{
  int ONE = Ox31;
  int TWO = Ox32;
  int THREE =Ox33;
  void onePressed();
  void twoPressed();
  void threePressed();
}
```

NumberInput

.

public static final ,
public abstract .

1 2 .

NumberInput.java(2)

```
public interface NumberInput
{
  public static final int ONE = Ox31;
  public static final int TWO = Ox32;
  public static final int THREE =Ox33;
  public abstract void onePressed();
  public abstract void twoPressed();
  public abstract void threePressed();
}
```

* [.java] , javac.exe ,
[.class] .

-

-

  .

-

  ,   .

  [public|<final||abstract>] class   implements   , ...
  {
     //
     //
     //   **(.)**
     //
  }

      :   **, class   ,   ,**
                      implements   **.   ,**

                   .

KeyTest.java(3)

```
import java.io.*;
class KeyTest implements NumberInput
{
  public void onePressed(){
    System.out.println("1            .");
  }
  public void twoPressed(){
    System.out.println("2            .");
  }
  public void threePressed(){
    System.out.println("3            .");
  }
}
```

KeyTest   NumberInput
                   3
                     .

4

```
public class ExecuteKeyTest
{
   public static void main(String[] args) throws IOException{
     int num;
     KeyTest key = new KeyTest();
     num = System.in.read();
     switch(num){
       case KeyTest.ONE: key.onePressed();
          break;
       case KeyTest.TWO: key.twoPressed();
          break;
       case KeyTest.THREE: key.threePressed();
          break;
       default:
          System.out.println("1, 2, 3                    .");
     }
   }
}
```

KeyTest
key   .

.

•

1   Enter   .
1   .

A   Enter   .
1, 2, 3   .

4

```java
public interface Conversions
{
    double INCH_TO_MM = 25.4;
    double OUNCE_TO_GRAM = 28.349523125;
    double POUND_TO_GRAM = 453.5924;
    double HP_TO_WATT = 745.7;
    double WATT_TO_HP = 1.0/HP_TO_WATT;
    double inchToMM(double inches);
    double ounceToGram(double ounces);
    double poundToGram(double pounds);
    double hpToWatt(double hp);
    double wattToHP(double watts);
}
class UnitConversions implements Conversions
{
    public double inchToMM(double inches){
        return inches*INCH_TO_MM; //
    }
    public double ounceToGram(double ounces){
        return ounces*OUNCE_TO_GRAM;
    }
    public double poundToGram(double pounds){
        return pounds*POUND_TO_GRAM;
    }
    public double hpToWatt(double hp){
        return hp*HP_TO_WATT;
    }
    public double wattToHP(double watts){
        return watts*WATT_TO_HP;
    }
}
```

Conversions

ExecuteUnitConversions(4)

```
public class ExecuteUnitConversions
{
  public static void main(String[] args){
    UnitConversions cv = new UnitConversions();
    System.out.println("                    "+cv.inchToMM(120.2));
    System.out.println("           "+cv.poundToGram(200.1));
    System.out.println("          "+cv.hpToWatt(50.0));
  }
}
```

UnitConversions
cv   .

.

•

Conversions   UnitConversions

,   .

3053.08
90763.83924
37285.0

•

• _____.

• (   ),
.

[public] interface   [extends   1, ...]
{
    //
    //
}

:   extends   .

,   {   }
.

NumberInputs.java(5)

```
public interface NumberInputs extends NumberInput
{
  int FOUR = 0x34;
  int FIVE = 0x35;
  void fourPressed();
  void fivePressed();
}
```

NumberInputs   NumberInput
                    2

                                    .

AlphaInput.java(6)

```
public interface AlphaInput
{
  int A_CHAR = 0x41;
  int B_CHAR = 0x42;
  void charAPressed();
  void charBPressed();
}
```

AlphaInput   2   2

                    .

NumberAlphaInputs.java(7)

```
public interface NumberAlphaInputs extends NumberInputs, AlphaInput
{
  int BS_CHAR = 0x08;
  int CR_CHAR = 0x0D;
  void bsPressed();
  void crPressed();
}
```

NumberAlphaInputs	NumberInputs
AlphaInput	2
	.

●

● ,

,   .

[public|<final||abstract>] class   extends   implements   , ...
{
    //
    //
    //
    //   .
}

    :
                . ,   extends   implements

                                                                                                        .

                ,   ,
                                        .

MultiInheritance.java(8)

```
import java.io.*;
class MultiInheritance extends KeyTest implements AlphaInput
{
  public void charAPressed(){
    System.out.println("     A          .");
  }
  public void charBPressed(){
    System.out.println("     B          .");
  }
}
```

MultiInheritance   KeyTest   ,
AlphaInput   ,
            2   .

ExMultiInheritance.java(8)

```
public class ExMultiInheritance
{
    public static void main(String[] args) throws IOException{
        int num;
        MultiInheritance key = new MultiInheritance();
        num = System.in.read();
        switch(num){
            case MultiInheritance.ONE: key.onePressed();
                break;
            case MultiInheritance.TWO: key.twoPressed();
                break;
            case MultiInheritance.THREE: key.threePressed();
                break;
            case MultiInheritance.A_CHAR: key.charAPressed();
                break;
            case MultiInheritance.B_CHAR: key.charBPressed();
                break;
            default:
                System.out.println("1, 2, 3, A   B                              .");
        }
    }
}
```

MultiInheritance   5
                        .

                                                        .

- **   :** 1, 2, 3, A   B   Enter   ,
                                        ,   [1, 2, 3, A   B
                    .]   .

instanceof

• instanceof

,   .

instanceof type(**)**

:   (   )
   true   ,   false   .

ExampleInstanceOf.java(9)

```
class IntegerVariable
{
   int i, j;
}
class CharVariable
{
   char ch1, ch2;
}
class ExampleInstanceOf
{
   public static void main(String[] args){
     IntegerVariable integer = new IntegerVariable();
     CharVariable character = new CharVariable();
     if( integer instanceof IntegerVariable )
        System.out.println("integer    IntegerVariable              ");
     else
        System.out.println("integer    IntegerVariable                ");
     if( character instanceof CharVariable )
        System.out.println("character    CharVariable            ");
     else
        System.out.println("character    CharVariable                  ");
   }
}
```

integer   IntegerVariable

- : if   integer   IntegerVariable

    true,   [integer   IntegerVariable   ]   ,

        if   character   CharVariable

[character   CharVariable   ]