Java Exercise date 26-02-2020 Part 3

**Question:** Describe the difference between Abstract class and Interface in Java. Give examples

**Answer:**

|  |  |  |
| --- | --- | --- |
| **N** | **Abstract** | **Interface** |
| **1** | Can define method. | Can define method. |
| **2** | Can define static properties. | Can define static properties. |
| **3** | A class can extends one class or one object. | A class can implement many interface. |
| **4** | Can contain instance property. | Can not contain instance property. |
| **5** | Can not implement a method. | Can not implement a method. |
| **6** | Can not create object directly. | Can not create object directly |
| **7** | The **abstract** keyword is used to declare abstract class. | The **interface** keyword is used to declare interface. |
| **8** | class can provide the implementation of interface. | Can not provide the implementation of abstract class. |
| **9** | doesn't support multiple inheritance. | supports multiple inheritance. |

**Example: Abstract class can have protected and public abstract methods**

abstract class Example1{

protected abstract void display1();

public abstract void display2();

public abstract void display3();

}

class Example2 extends Example1{

public void display1(){

System.out.println("display1 method");

}

public void display2(){

System.out.println("display2 method");

}

public void display3(){

System.out.println("display3 method");

}

}

class Demo{

public static void main(String args[]){

Example2 obj=new Example2();

obj.display1();

}

}

**Interface can have only public abstract methods**

interface Example1{

void display1();

}

class Example2 implements Example1{

public void display1(){

System.out.println("display1 method");

}

public void display2(){

System.out.println("display2 method");

}

}

class Demo{

public static void main(String args[]){

Example2 obj=new Example2();

obj.display1();

}

}