

Q Resolving confusion between Abstract class and Interface:

① Here we can use Interface and abstract class:

code:

interface Switch

void click();

}

abstract class BatterySwitch implements Switch

void checkBattery() {

System.out.println("Battery is full.");

}

;

7

class SimpleSwitch implements Switch

public void check() {

System.out.println("Light turned ON");

};

class RemoteSwitch extends BatterySwitch

public void check() {  
checkBattery();}

System.out.println("Remote light turned  
on.");

}

public class Abstraction

public static void main(String[] args) {

Sketch sketch = new SimpleSwitch();

s1.click();

sketch.click();

s2.click();

sketch.click();

System.out.println("Switch

" + sketch.getState() + " was triggered two stages

(2) calling a method from an interface can be slightly slower than from an abstract class because the JVM uses a different lookup mechanism. Interfaces use an interface table which abstract classes use a virtual table. The virtual table is usually faster to access. For example code:

interface playable characters

void play();

abstract class fighters

void attack();

System.out.println("Attack with sword")

}

class Hero implements playable characters

public void play(){}

System.out.println("Hero is playing the game")

}

Class Warriors extends Fighter{

public class abstractors?

```
public static void main(String[] args){
```

playable character. h = new hero

n. players, ~~not~~ <sup>INT</sup> bestiv o 22N 2020(D)

warrior w = new. works (j)

w. attack ( ) ; old names not

Hence Hero() class is slower than  
warrior class.

1000 ft. (the "Doff A") striking two miles

### ③ The difference between Abstract class and Interface class:

Feature method Type	Abstract class	Interface
Abstraction	Abstract and concrete method	Abstract default static from java
Object creation	Here can be abstraction classes or normal classes or both	strictly have abstract classes
Multiple Inheritance	cannot be created	can be created
Access Modifiers	Not supported	Supported
	can be use public protected	All method are public by default