

④ 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.");

}

}

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) {
```

```
        switch s1 = new simpleswitch();
```

```
        s1.click();
```

```
        switch s2 = new Remoteswitch();
```

```
        s2.click();
```


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

For example code:

```
interface playable character {  
    void play();  
}
```

```
abstract class fighters {  
    void attack() {  
        system.out.println("Attack with sword!");  
    }  
}
```

```
class Hero implements playable character {  
    public void play() {  
        system.out.println("Here is playing the game");  
    }  
}
```


class warrior extends Fighter {

public class abstract {
public static void main(String[] args) {
playable character h = new Hero();

h.play();

warrior w = new warrior();

w.attack();

}

Here Hero() class is slower than warrior class.

("I know this stuff A") attacking two mages

}

("I know this stuff A") attacking two mages

③ The difference between Abstract class and Interface class:

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