第四组

一、选择题

1.参考：<https://www.runoob.com/java/java-modifier-types.html>

关键点总结：

default使用对象：类、接口、变量、方法。

private使用对象：变量、方法。 注意：不能修饰类（外部类）

public使用对象：类、接口、变量、方法

protected使用对象：变量、方法。 注意：不能修饰类（外部类）。

private和protected可以修饰内部类。

static使用对象：成员变量(类变量)、方法(类方法)、内部类

final使用对象：类、方法、变量

abstract使用对象：类、方法

方法中的变量，即局部变量，不能用public、protected、private以及static修饰符



内部类介绍：

<https://www.runoob.com/java/java-inner-class.html>

<https://www.runoob.com/w3cnote/java-inner-class-intro.html>(此链接中有关深入理解内部类第二个问题的说明应该是错的)

静态内部类中，注意以下代码是合法的：

**class** OuterClass {

**static** **int** *x* = 10;

**static** **class** InnerClass {

**public** **int** myInnerMethod() {

*x*=*x*+10;

**return** *x*;

}

}

}

**public** **class** test1 {

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

OuterClass.InnerClass myInner = **new** OuterClass.InnerClass();

System.***out***.println(myInner.myInnerMethod());

}

}

另外，下面的代码是非法的：**class** OuterClass {

**static** **int** *x* = 10;

**static** **class** InnerClass {

**public** **int** myInnerMethod() {

*x*=*x*+10;

**return** *x*;

}

}

}

**public** **class** test1 {

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

OuterClass myOuter = **new** OuterClass();

OuterClass.InnerClass myInner=myOuter.**new** InnerClass();//该句报错 因为内部类是static的

System.***out***.println(myInner.myInnerMethod());

}

}

protected关键字详解(参考用，好像讲的不是太对)：

<https://www.runoob.com/w3cnote/java-protected-keyword-detailed-explanation.html>

注意clone方法的修饰符是protected:

该链接中示例一，注意clone方法会自动被Son1,Son11继承，无需显示声明，这样Son1,Son11才可以在合适的地方调用clone()。

Java接口介绍：

<https://www.runoob.com/java/java-interfaces.html>

default关键字介绍：

<https://www.runoob.com/java/java8-default-methods.html>

接口里的变量都隐式声明为public static final,而接口里的方法默认情况下访问权限为public。

接口及接口的成员变量和成员方法不能声明为protected、private