一、选择题

1.知识点：

(1)this不能用在静态方法中,可以在非静态方法中用this：

示例1：

**public** **class** Test1 {

**private** **float** f =10.6f;

**int** i=16;

**static** **int** *si*=10;

**void** a() {

**this**.*si*=15;//使用this关键字

}

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

Test1 test1=**new** Test1();

test1.a();

System.***out***.println(*si*);//输出15

}

}

示例2：

**public** **class** Test1 {

**private** **float** f =10.6f;

**int** i=16;

**static** **int** *si*=10;

**static** **void** a() {

**this**.si=15;//报错，不能在静态方法中用this

}

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

Test1 test1=**new** Test1();

test1.*a*();

System.***out***.println(*si*);

}

}

(2)静态方法可以直接调用，非静态方法必须依赖于具体的对象调用.

示例1：

**public** **class** Test1 {

**private** **float** f =10.6f;

**int** i=16;

**static** **int** *si*=10;

**static** **void** a() {

*si*=15;

}

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

Test1 test1=**new** Test1();

*a*();//静态方法可以直接调用，也可以test1.*a*()这样调用

System.***out***.println(*si*);//输出15

}

}

示例2：

**public** **class** Test1 {

**private** **float** f =10.6f;

**int** i=16;

**static** **int** *si*=10;

**void** a() {

*si*=15;

}

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

Test1 test1=**new** Test1();

a();//报错，必须是test1.*a*()

System.***out***.println(*si*);

}

}

2.复习static的使用范围：成员变量(类变量)、方法(类方法)、内部类

3.A.接口中方法默认是公有抽象的，即public abstract，不能有方法体，否则会报错。若用default修饰方法，方法才能有方法体。

B.接口中方法可以使用的修饰符有：public，abstract，default，static，strictfp

D.接口中的变量默认是公有常量，即public static final类型变量，接口中变量可用修饰符有：public，static，final，接口中的变量必须初始化。

4.重载和重写(覆盖)的详解

参考链接:<https://www.runoob.com/java/java-override-overload.html>

注意重写的第二个bark()示例很有迷惑性。尤其要注意重写的规则。有关super用法也可参见上面链接。

6、7、8题:

注意6、7题的选项混淆了多重继承和多继承，这两者是不同的概念，注意区分，参考链接：

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

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

要点总结：一个类可以实现(implements)多个接口(8题);

一个子类只能继承(extends)一个父类,即Java语言中类间的继承关系是单继承(6题);

一个子接口可以继承(extends)多个父接口, 即Java语言中接口间的继承关系是多继承(7题)。

9.参考(4)组知识点总结

10.另外参考(5)组第一题对this的总结,super用法参考4题。

二、填空题