

## Chapter 05 面向对象基础

---

### Key Point :

- 类和对象的概念
- 实例变量
- 方法重载
- 构造方法
- 引用的概念
- this关键字

### 答案：

1. C

- I. 调用第一个 method 方法时，根据方法重载，调用的是无参的 method 方法，打印的 value 为实例变量的值，结果为 10。
- II. 调用第二个 method 方法时，会调用 method(int value)这个方法，由于这个方法具有一个 value 形参，相当于局部变量。局部变量会和实例变量命名冲突，打印时会打印局部变量的值，也就是打印参数 20。

2. 不能编译通过，因为 int method(int)方法没有返回值。需添加相应的 return 语句。

3. ABD

A 正确，因为即使不写构造方法，编译器也会自动生成一个默认构造方法。

B 正确，一个类中可以有多个构造方法，这些方法间构成重载

C 错误，构造方法没有返回值

D 正确。

4. 输出结果为：10、11 （考点：引用的语义）
5. 不能编译通过，需要增加两个构造函数：MyClass()以及 MyClass(int)。
6. 参考代码：

//定义一个 Dog 类

```
class Dog{  
  
    String name;  
  
    int age;  
  
    boolean sexual; //true 表示为公，false 表示为母。  
  
    public Dog(){  
  
        //分别根据参数，设置 Dog 类的属性  
  
        public Dog(String name,int age,boolean sexual){  
  
            this.name=name;  
  
            this.age=age;  
  
            this.sexual=sexual;  
  
        }  
  
        public void play(){  
  
            System.out.println(name+" play" );  
  
        }  
  
        public void play(int n){
```

```

        System.out.println(name+" play" +n+" minutes" );

    }

}

```

```

public class TestDog{

    public static void main(String args[]){

        Dog d;

        //创建一个 Dog 对象，调用有参构造函数

        //名字为 joy，年龄为 2 岁，性别为母

        d=new Dog( "joy" ,2,false);

        //调用 Dog 对象无参的 play 方法。

        d.play();

        //调用 Dog 对象有参的 play 方法，参数为 30

        d.play(30);

    }

}

```

## 7. C

根据对象创建过程，创建 ClassC 对象时，

- I. 为 ClassC 对象分配空间
- II. 初始化 ClassC 属性。此时，需要创建 ClassA 对象
  - 1). 为 ClassA 对象分配空间
  - 2). 初始化 ClassA 对象的属性
  - 3). 调用 ClassA 对象的构造方法，输出 "ClassA()"

### III. 调用 ClassC 构造方法

- 1). 输出 "ClassC()"
- 2). 在 ClassC 构造方法中创建 ClassB 对象，创建时会调用 ClassB 构造方法，输出 ClassB()"

8. C；请参考“Java 方法参数传递规则”相关的例子。

9. 参考代码：

```
public class Student{

    public void Student(){} //构造函数不能有返回值类型，应把 void 去掉。

    void init(){

        age=10;

        name="limy";

    }

    public Student(String name){

        this.init();

        this.name=name;

    }

    public Student(String name,int age){

        this.init();

        this(name);//this()必须放在构造函数中的第一行 this.age=age;

    }

    int age;

    String name;

}
```

10. 参考代码：

```
Public class Worker{

    String name; //工人的姓名

    int age; //工人的年龄

    double salary; //工人的工资

    public Worker(String name,int age,double salary){

        super();

        this.name=name;

        this.age=age;

        this.salary=salary;

    }

    public Worker(){

        super();

    }

    public void work(){

    }

    public void work(int time){

    }

}
```

11. 参考代码：

```
public class Address{

    String address;//表示地址 String zipCode; //表示邮编

    public Address(String address,String zipCode){

        super();

    }

}
```

```
        this.address=address;

        this.zipCode=zipCode;
    }

    public Address(){

        super();
    }
}
```

## 12. 参考 TestWorker.java