回顾：

1. for:应用率高、与次数相关
2. break: 跳出循环

continue: 跳过循环体中剩余语句而进入下一次循环

1. 循环选择情况：
2. 嵌套循环:

循环中套循环，外层走一次，内层走所有次，层数越少越好，break跳出一层循环

1. 数组:

相同数据类型元素的集合、数据类型

int[ ] arr = new int[3];

int[ ] arr = {7,8,9};

int[ ] arr = new int[ ]{7,8,9};

arr[0] =100;

System.out.println(arr[arr.length-1]);

for(int i=0;i<arr.length;i++){ //遍历arr数组

arr[i] = 100;

arr[i] = (int)(Math.random()\*100);

arr[i] = scan.nextInt();

arr[i]------代表arr中的每一个元素

}

正课：

1. 数组:

7）数组的复制：

7.1) System.arraycopy(a,1,b,0,4);

7.2) int[ ] b = Arrays.copyOf(a,6);

a = Arrays.copyOf(a,a.length+1); //数组的扩容

8）数组的排序：

8.1) Arrays.sort(arr); //简单并且效率高----升序

1. 方法：函数、过程

1) 封装一段特定的业务逻辑功能

2) 方法要尽可能独立，一个方法只干一件事

3) 方法可以被反复调用多次

4) 减少代码重复、有利于代码的维护，有利于团队协作

1. 方法的定义:

修饰词 返回值类型 方法名(参数列表){

方法体-----具体的业务逻辑功能实现

}

1. 方法的调用:

1) 无返回值: 方法名(有参传参);

2) 有返回值: 数据类型 变量 = 方法名(有参传参);

1. return:

1) return 值; //1.1)结束方法的执行 1.2)返回结果给调用方

2)

晚课-----方法三个小点

-----方法与数组相结合的例子

方法怎么设计------需要后期项目需求的积累(飞机大战)

方法可以有参，也可以无参

------无参意味着数据都是写固定了的

------有参可以使方法更加灵活(数据不是固定的)

System.out. println(25);

System. arraycopy(a,1,b,0,4);

int[] b = Arrays. copyOf(a,6);

Arrays. sort(arr); ------------------有参数

int a = scan. nextInt();

double b = scan. nextDouble();

double c = Math. random(); ----------------无参数

方法可以有返回值，也可以没有返回值：

1） 无返回值-------返回值类型写成void

2） 有返回值-------返回值类型写为特定的数据类型即可

何时有返回值？何时没有返回值？

\1. 方法操作完后：

1） 若还需要用到方法中的某个数据--------有返回值

2） 若不需要用到方法中的某个数据--------无返回值

System.out.println(25); //输出后，目的已经达到，不需要用到其中的数据

System.arraycopy(a,1,b,0,4); //复制后，数据已经好了，不需要用到其中的数据

Arrays.sort(arr); //排序后，数据已经排好了，不需要用到其中的数据-----------无返回值

int a = scan.nextInt(); //接收用户输入的整数后，我还需要用这个数做操作

double b = scan.nextDouble(); //接收用户输入的小数后，我还需要用这个数做操作

double c = Math.random(); //生成随机数后，我还需要用这个数做操作---------------有返回值

public static

main(){

入口

}

冒泡排序6句话

10个地方得排序-----6句话写10次

调用10次

冒泡排序(){

6句话

}

方法-----------讲语法(规定)

-----------具体如何设计方法（面向对象中涉及得比较多一些）

API

System.out.println( num ) ;

写代码----不要贪个数多

听得懂，写不出来------------绝大部分同学的状态

代码不是听懂的，一定是练懂的-------多练

明天案例复习课---------把基础打牢固一些

API-----------小零件

int[ ] arr = new int[3];

arr[0]-------代表第一个元素

arr[2]-------代表最后一个元素

for(int i=0;i<arr.length;i++){

arr[i] = 100;

arr[i]------代表每一个元素

}

arr.length-1

Arr[] = max;

i=0. arr[0]=100;

i=1. arr[1]=100;

i=2. arr[2]=100;

引用类型数组-------------------下周一讲

数组的数组(二维数组)------------就业冲刺课讲

冒泡排序-----------------------就业冲刺课讲

**int**[] a = {10,20,30,40,50}; //数组一旦定义好了，长度就不能改变了

//创建了一个新的数组

a = Arrays.copyOf(a,a.length+1);

数组 新数组

10,20,30,40,50 10,20,30,40,50,0

a

int[ ] a = {10, 20, 30, 40, 50};

int[ ] b = new int[6]; //0,0,0,0,0,0

for(int i=0;i<a.length;i++){

b[i] = a[i];

}

b[0] = a[0];

b[1] = a[1];

b[2] = a[2];

b[3] = a[3];

b[4] = a[4];

int[ ] arr = { 56, 78, 89, 3 };

找最大值算法:

\1. 假设第一个数为最大值：

int max = arr[0];

\2. 遍历剩余元素，将剩余元素与max做对比，若剩余元素大于max，则修改max的值为较大的

for(int i=1;i<arr.length;i++){

if( arr[i]>max ){

max = arr[i];

}

}

max=56/78/89

复习课（小案例为主）

代码一定要多写