1. 现有字符串a=”abcdf”和b=”eedsf”，分别用String和StringBuffer进行合并链接成新的字符串c打印输出。并截取字符串3-6位的值。

package com.test;

public class Test04 {

public static void main(String[] args) {

String str1 = "abcdf";

String str2 = "eedsf";

String c = str1.concat(str2);

StringBuffer buff = new StringBuffer();

buff.append("abcdf");

buff.append("eedsf");

System.out.println(buff.toString());

System.out.println(c);

System.out.println(c.substring(2, 6));

}

}

1. 使用java代码对十进制数150进行进制转换成 二进制，8进制，16进制。

package com.test;

public class Test07 {

public static void main(String[] args) {

System.out.println(Integer.toBinaryString(150));

System.out.println(Integer.toOctalString(150));

System.out.println(Integer.toHexString(150));

}

}

1. 判断一下代码运行结果\_\_\_\_B\_\_\_\_\_.

public class Test{

public static void main(String[] args) {

int a = 3;

a+=a;

System.out.println(a);

}

}

A:4 B:6 C:3 D: 运行时出错

1. 对34，56，32，10，45进行从小到大冒泡排序

package com.test;

public class Maopao {

public static void main(String[] args) {

int[] arr = { 34, 56, 32, 10, 45 };

System.out.println("排序前数组为：");

for (int num : arr) {

System.out.print(num + " ");

}

for (int i = 0; i < arr.length - 1; i++) {

for (int j = 0; j < arr.length - 1 - i; j++) {

if (arr[j] > arr[j + 1]) {

int temp = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = temp;

}

}

}

System.out.println();

System.out.println("排序后的数组为：");

for (int num : arr) {

System.out.print(num + " ");

}

}

}

1. for 或者while或者do while 写出九九乘法表。

//for循环九九乘法

package com.test;

public class Chengfabiao {

public static void main(String[] args) {

for(int i=1;i<=9;i++){

for(int j=1;j<=9;j++){

System.out.println(i+"\*"+j+"="+i\*j+"\t");

}

}

附加题：有条件的通彻可以写一下杨辉三角

package com.test;

public class Yanghui {

public static void main(String[] args) {

int [][] iArray = new int[10][];

for(int i=0;i<10;i++) {

iArray[i] = new int[i+1];

for(int k=9;k>i;k--){

System.out.print(" ");

}

for(int j=0;j<=i;j++) {

if(j==0 || j==i) {

iArray[i][j]=1;

System.out.print(iArray[i][j]+" ");

}else{

iArray[i][j]=iArray[i-1][j]+iArray[i-1][j-1];

System.out.print(iArray[i][j]+" ");

}

}

System.out.println();

}

}

}