1.Java语言中，流程控制分为哪些？

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 分支语句 | if-else | break | switch | return |
| 循环语句 | while | do-while | for | continue |
| 异常处理语句 | try | catch | finally | throw |
| 跳转语句 | break | continue | return |  |

2.if-else和switch在断定条件上各位何种数据类型？

①if-else根据其布尔型表达式的值来确定程序流程的选择；

②switch的断定条件可以是byte、short、char、int，但不能是布尔型的值。

3.循环语句中由哪几个部分组成，各有何作用？

①初始化部分：为循环设置初始量；

②判断部分：布尔表达式，其值决定循环是否继续；

③循环体部分：循环的代码段（语句）；

④迭代部分：每循环一次，改变循环控制变量的值。

4.比较break和continue语句的区别。

①break的作用是跳出当前循环块（for、while、do while）或程序块（switch）。在循环块中的作用是跳出当前正在循环的循环体。在程序块中的作用是中断和下一个case条件的比较。

②continue用于结束循环体中其后语句的执行，并跳回循环程序块的开头执行下一次循环。

5.数组的声明形式和Java数组初始化的含义。

①type arrayName[][[]…]; //type为Java任意的数据类型，arrayName为数组名

②type[][[]…] arrayName; //[]表示数组的一维，[[]…]表示数组的多维

③Java数组初始化的含义为对数组中的每个元素进行储存空间分配，并赋初值。当创建一个数组时，每个元素都被初始化。所有变量的初始化（包括数组元素）是保证系统安全的基础，变量绝不能在未初始化状态使用。

6.数组的length属性是指什么，在多维数组中如何来获取length的值？

①Java数组的length属性代表的是数组的长度属性，length的值取决于数组在这个维上的元素个数，length的值随着new运算符实例化指定的大小而定。

②在多维数组中，获取length的值应指定多维数组中特定一位数组所在的位置。例如在二维数组Array[3][4]中，若要得到Array.length指的是数组的行数，值为3；Array[2].length指的是第二行的长度，值为4。

7.创建一个2×3的二维int数组，并用1到6的整数作为元素的初始值。

|  |
| --- |
| makeArray.java |
| public class makeArray{  public static void main(String args[]){  int arr[][] = new int [2][3];  int count = 1;  for(int i=0; i<arr.length; i++){  for(int j=0; j<arr[i].length; j++){  arr[i][j] = count++;  }  }    for(int i=0; i<arr.length; i++){  for(int j=0; j<arr[i].length; j++){  System.out.println("arr[" + i + "]" + "[" + j+ "]" + "=" + arr[i][j]);  }  }  }  } |
|  |

8.

|  |
| --- |
| TestArray.java |
| import java.util.Scanner;  public class TestArray{  public static void main(String args[]){  Scanner input = new Scanner(System.in);  System.out.println("Please enter the length of the array:");    int len = input.nextInt();  int sim[] = new int[len];    System.out.println("Please input the number in array:");  for(int i = 0; i < sim.length; i++){  int num = input.nextInt();  sim[i] = num;    }    for(int i = 0; i < sim.length - 1; i++){  for(int j = 0; j <sim.length - i -1; j++){  if(sim[j] > sim[j+1]){  int temp = sim[j];  sim[j] = sim[j+1];  sim[j+1] = temp;  }  }  }      for(int i=0; i< sim.length; i++){  System.out.print(sim[i] + " ");  }  }  } |
|  |