**Java开发方向：**

1. Java SE（Java Platform Standard Edition）：Java标准版
2. Java EE（Java Platform Enterprise Edition ）：Java企业版
3. Java ME（Java Platform Micro Edition）：Java微版

**Java程序的运行机制和Java虚拟机：**

在Java中源文件名称的扩展名为\*.java，通过编译生成\*.class文件，在计算机上执行\*.class文件，但此时执行\*.class的计算机并不是物理上可见的计算机，而是Java自己设计的虚拟机，简称Java虚拟机（JVM ，Java Virtual Machine），Java也是通过JVM进行可移植操作的。

**Java中的类：**

Java中的程序是以类为单位的，对于类的定义有两种形式：

1. public class定义的类要求文件名与类名称保持一致，而在\*.java文件中只允许有一个public class定义的类。
2. class定义的类文件名可以与类名称不一致，但编译后生成的\*.class文件的名称就是class定义的类名称。

**Java中的注释：**

1. 单行注释：//
2. 多行注释：/\* \*/
3. 文档注释：/\*\* \*/

**Java中的标识符：**

Java中的标识符由字母、数字、下划线（\_）和美元符号（$）组成，但不能以数字开头，也不能是Java中的关键字（保留字）。

**Java数据类型的划分：**

Java中的数据类型可分为基本数据类型和引用数据类型。

1. 基本数据类型：数值型（（整型：byte、short、int、long）（浮点型：float、double）），字符型（char），布尔型（boolean）
2. 引用数据类型：类（class）、接口（interface）、数组

**数据定义规则：**

1. 在Java中对于一个整型常量，其为int型，所以在声明常量时不要超过int数据类型的范围。
2. 使用浮点型数值时，默认的类型是double型，在数值后面可加上D或d，作为double类型的标识。在Java中，D或d是可有可无的。在数据后面加上F或f，则作为float类型，如果没有加上，Java就会将该数据视为double类型，而在编译时就会发生错误提示会告诉设计者可能失去精确度。

**方法重载：**

方法重载是指方法名称相同，但参数类型或参数的个数不同。方法重载发生在同一个类中。

**数组排序与数组复制：**

1. 数组排序：java.util.Arrays.sort(数组名称);
2. 数组复制：System.arraycopy(源数组名称，源数组开始点，目标数组名称，目标数

组开始点，复制长度）；

**Java新特性下的可变参数：**

可变参数即方法中可以接收的参数不再是固定的，而是随着需要传递的，定义格式如下：

访问权限 返回值类型 方法名称（类型…参数名称）{}

**Java新特性下的foreach输出：**

数组一般使用for循环输出，但同时提供了一种foreach语法，使用格式如下:

for(数据类型 变量名称：数组名称){

System.out.println（变量名称）；

}

但从实际应用来看，还是使用原始的输出操作更为合适。