**一、构造器**

构造器：用于产生类的对象，

构造器的定义语法 ：[访问权限修饰符] 类名（[<参数列表>]）{}

构造器产生对象语法 ： 类名 对象名 = new 构造器(参数列表)；

--> 通过对象可以获取属性，并调用方法 ：

语法 : DataType name = 对象名.属性名 ； 获取属性

[DataType name ] = 对象名.方法名（参数列表） 调用方法

**二、信息的封装与隐藏**：

信息的隐藏：属性或则方法的访问权限修饰符写为 private

属性定义为 private 后 ，只能在本类中访问，在别的类中则不能访问

同理方法 定义为 private 后，也只能在本类中调用；

信息的封装：为私有的属性提供一个 public 修饰的get 和 set方法

**get方法: 用于获取属性的值**

语法规则 ： <public> <属性的Type> <getAttrName>(){

return attrName;

}

方法名: 将属性名首字母改为大写，前面加上get

返回值类型：与当前属性的数据类型保持一致，直接通过return返回

参数列表： 没有参数列表

**set方法：用于给属性设置值**

语法规则 ： <public> <void> <setAttrName>( <属性Type> 属性名 ){

this.attrName = attrName;

}

方法名：将属性名的首字母改为大写 ，前面加上 set

返回值：无返回值

参数列表：数据类型与当前属性数据类型保持一致

**三 数据类型**

java是一种强数据类型的编程语言,不同的数据存放在不同的**变量**中

|  |  |
| --- | --- |
| 强类型(java） | 弱类型(JavaScript) |
| int i = 5;  float f = 3.4f;  String str = “haha” | var i = 5;  var s = “haha” |

**变量:** 在系统中分配一块内存单元，用于存放数据

数据类型分为简单类型和引用类型

简单类型：byte short int long float double boolean char

引用类型：除了简单类型就是引用类型 String , int[]

**简单类型：**在栈区中开辟内存空间用于存放变量，同时在变量空间中存放值

**引用类型:** 在栈区中开辟内存空间用于存放引用，值存放在堆区中；

|  |
| --- |
|  |

**数据类型的区分**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 数  据  类  型 | 简  单  类  型 | 数  字  类  型 | 整  型 | byte | 1字节 | -2^7 ~ 2^7 -1 |
| short | 2字节 | -2^15 ~2^15-1 |
| int | 4字节 | -2^31 ~2^31-1 |
| long | 8字节 | -2^63~2^63-1 |
| 浮点型 | float | 4字节 |  |
| double | 8字节 |  |
| 字符 | Char 用‘’ | | 2字节 |  |
| 布尔 | boolean | | 1bit | true /false |
| 引用  类型 | 字符串 ： String str ; 数组： int [] a  对象 ： Student stu; | | | | |

**简单类型分为：**

整型： 用于表示一个整型的数据 int i = 5;

浮点型：用于表示浮点型的数据（小数） float = 3.4f;

字符类型： 用于表示一个字符 char c = ‘A’

布尔类型 ：用于比较活判断某一个结果是否正确（true / false）

|  |
| --- |
|  |

**四 生成 javadoc 文档**

（1）完成类，方法的注解

（2）点击的导航栏的 Project -->Grenerate Javadoc

（3）在路径中有一个 index.html 打开会有所有的属性以及方法

|  |
| --- |
|  |

**标识符：用于给变量，类，方法命名**

标识符用作给变量、类和方法命名

可以以字母、下划线“\_”和”$”符开头

首字符外，可以跟上字母、下划线“\_”和”$”符或数字

Java是大小写敏感的，标识符也不例外

|  |  |
| --- | --- |
| **简单数据类型** | byte short int long float double boolean char |
| **访问权限修饰符** | public private proctected |
| **异常处理** | try catch fianlly throw throws |
| **流程控制** | if, else ,for, while, switch ,case, break, continue,do |
| **面向对象** | new extends , interface , abstract , implements , instanceof 。。。。。。。。 |

**五 成员变量与局部变量**

|  |  |  |
| --- | --- | --- |
|  | **全局变量** | **局部变量** |
| 定义 | 定义在类中 | 定义代码块中，if,else方法 |
| 作用域 | 作用在整个类中 | 只作用域当前代码块 |
| 访问权限修饰符 | 需要给访问权限修饰符 | 不能给访问权限修饰符 |
| 初始化 | 可以初始化，也可以不初始化 | 必须初始化 |

**全局变量如果没有赋予初始值，会默认一个值**

|  |
| --- |
|  |