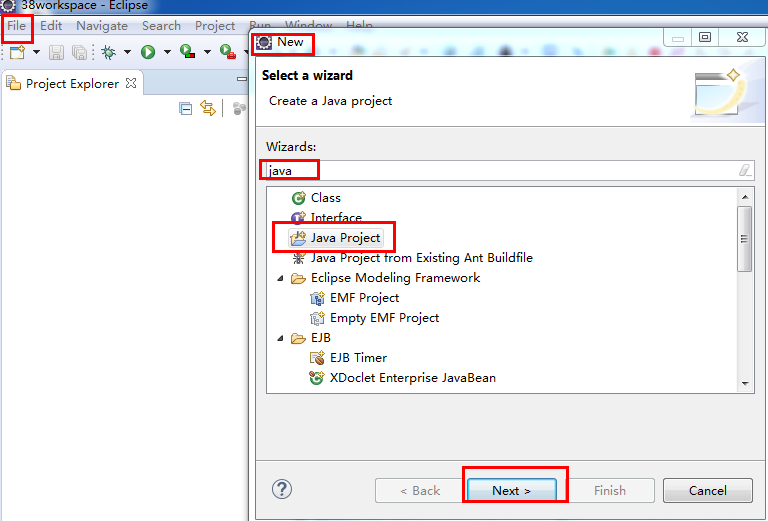
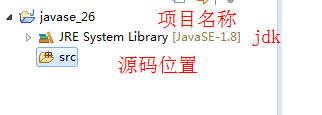
# 一：eclipse的使用

>创建java项目

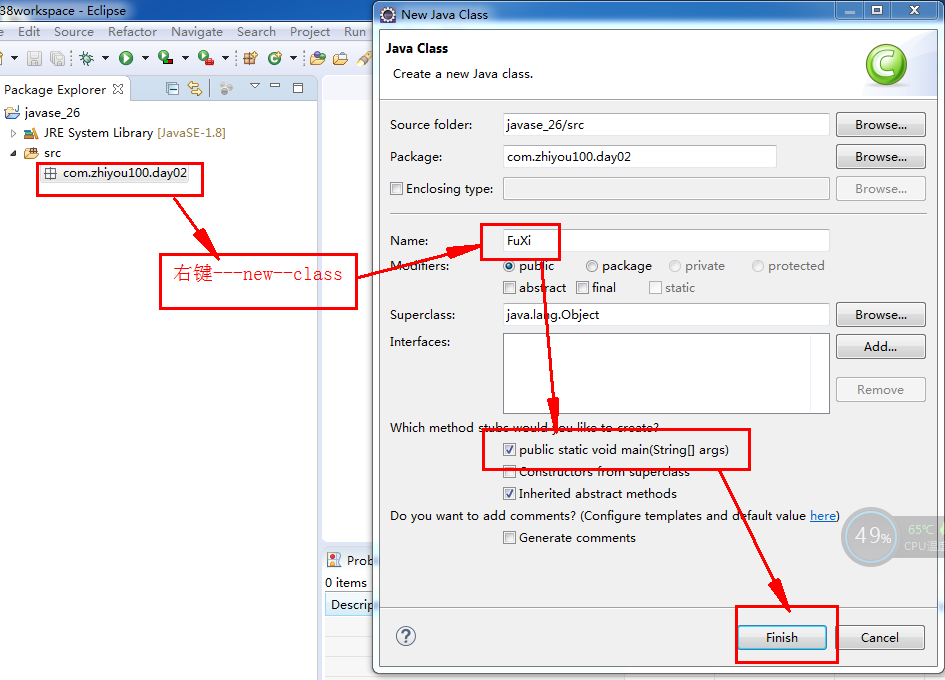


>项目目录介绍

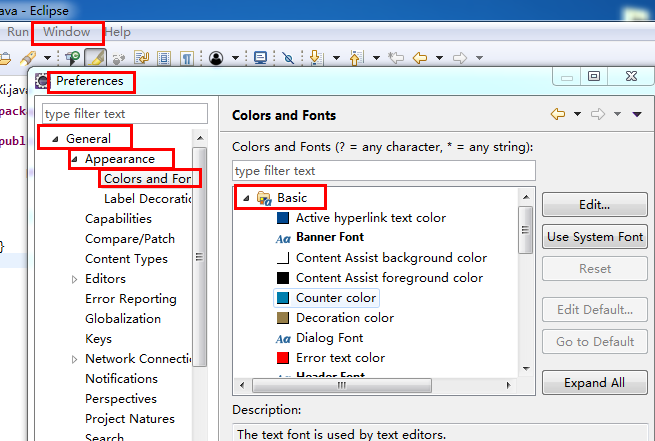


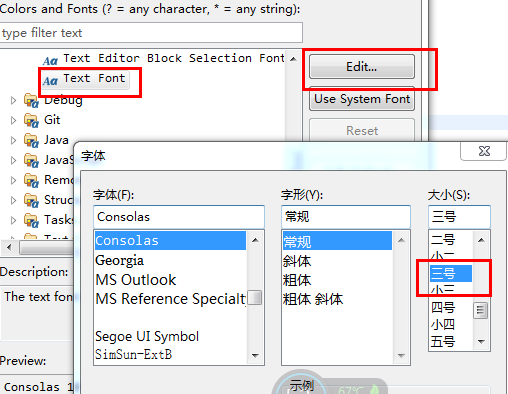
>先在src下创建package

>再在包下创建类

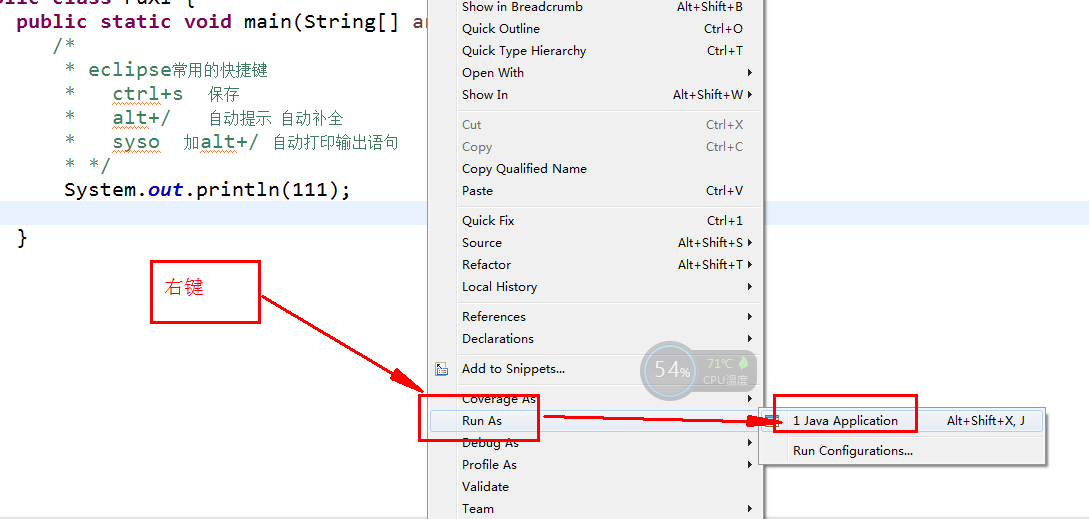


>更改字体大小

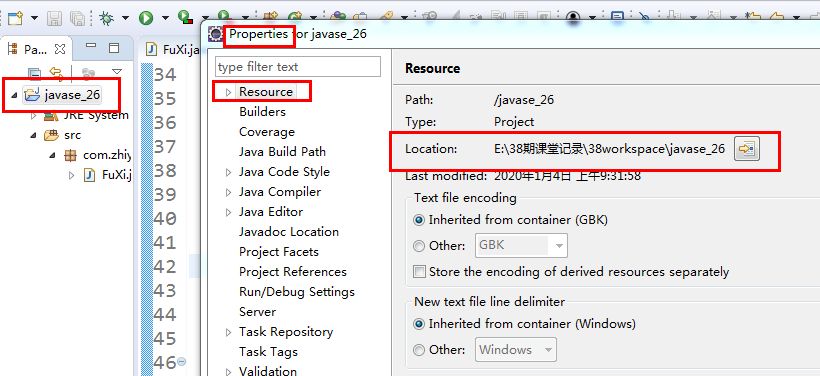




>运行类



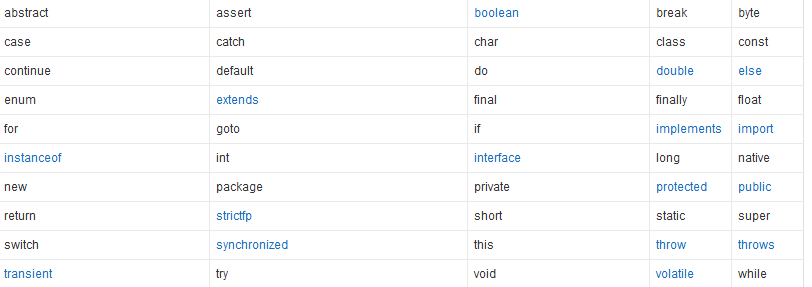
>查看源文件的位置



# 二：语法之数据类型

>标示符：所有的名字

>关键字：被java赋予特殊含义的单词或者字符



>数据类型：

按复杂程度分：

---基本数据类型：单值数据

---引用数据类型：由多个单值数据有机组合形成的整体

基本数据类型数据按数据性质分类：

---整数类型：没有小数点的数字： 1 11 -1

---浮点类型：有小数点的数字： 1.1 1.0 -1.0

---字符类型：所有的符号： ‘1’ ‘a’ ‘ ’

---布尔类型：只有两种可能 并且是相反的 : true false

基本数据类型数据按空间大小分类:八种基本数据类型

---整数类型：

---byte: 字节型 取值范围 [-128,127] 描述数据的基本单位 占 1个字节

---short：短整型 取值大概范围[-32000,32000] 占2个字节

---int: 整型 取值大概范围[-1000000000,1000000000] 占4个字节

整数数字 默认是int类型

---long：长整型 long数字后面加l/L 1L 占8个字节

---浮点类型

---float： 单精度 小数点后8到6位是精确的数据 占4个字节

Float数据 后面加f/F 1.1f

---double: 双精度 占8个字节

浮点数字 默认是double类型 1.1

---字符类型

---char: 字符型 占2个字节 字符数据必须写在单引号中 并且单引号中有且只有一个字符

‘1’ ‘a’ ‘苗’ ‘天’

----布尔类型

---boolean: 布尔型 占1个字节 只有两个值 false、 true

注意：计算机存数数据的最小单位：bit

# 三：语法之常量和变量

按具体数据的值是否可变：分成常量和变量

程序运算过程中：数据的值不能变：：常量

整数常量：1 1L

浮点常量： 1.1 1.2f

字符常量： ‘1’ ‘a’

布尔常量： true false

null

程序运算过程中：数据的值可以变化：：变量 对应的是内存中的一块区域：有名字，类型，值

定义变量语句格式：数据类型 变量名;

变量赋值语句格式：变量名=常量值;

变量的注意事项：

》 变量必须先定义 然后赋值 最后使用

》 同一个变量只能定义一次 但可以赋值和使用多次

》 赋值时 值的类型必须和变量的类型一致

》定义变量语句和变量赋值语句可以合并：格式：数据类型 变量名=值;

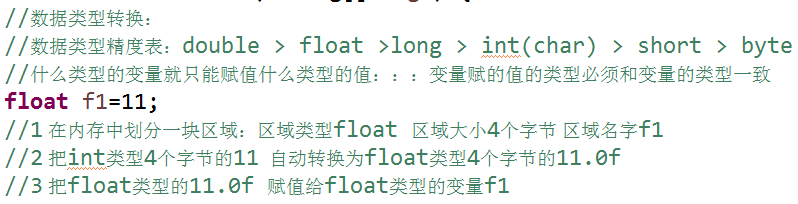


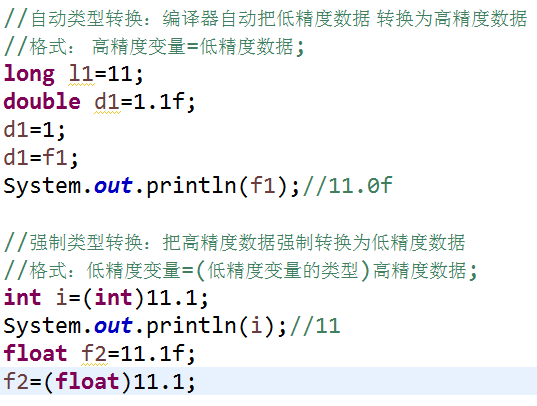
# 四：语法之数据类型转换

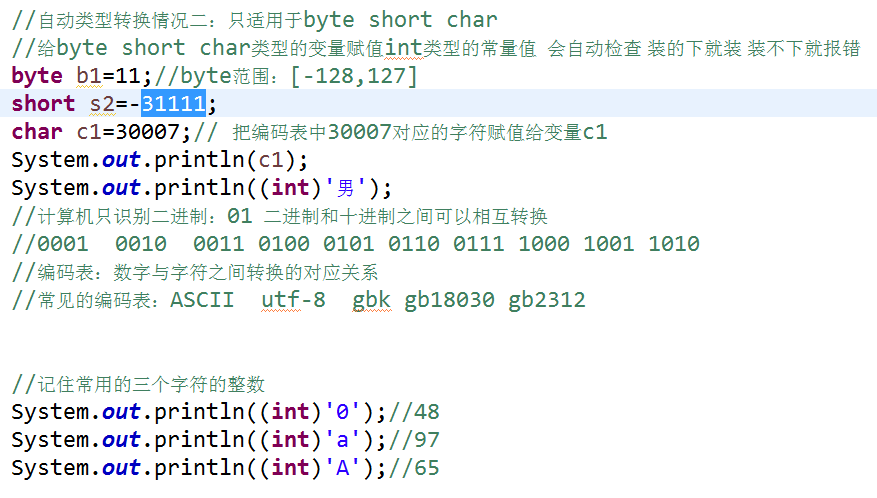
》 数据类型转换 只针对数字类型和字符类型的数据 布尔不能转换

》 数据类型精度表： double > float > long > int(char) > short > byte

数据类型转换：：：不同类型的数据之间相互转换







ASCII编码表：



# 五：语法之运算符

》算术运算符：+ - \* /

》》+ 加法运算

注意 float和double进行运算时 会出现精度丢失的情况

2.0-1.1=0.899999999

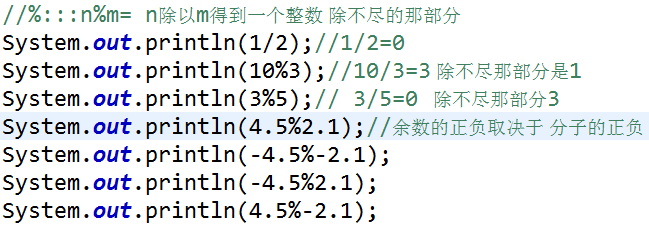
》》+ 作为正号

Int i1=+1;

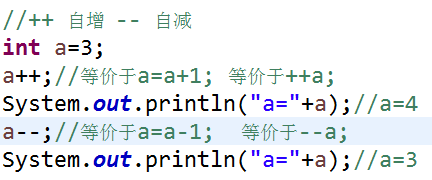
》》+ 当+前后有字符串时 +是连接符：：：：任何数据与字符串连接都形成一个新的字符串

1+2+”abc”+1+2 = ”3abc12”

》》% 取模 求余



》》++ 自增 --自减



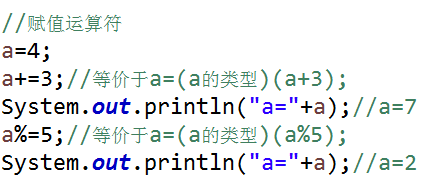
》比较运算符： > >= < <= 比较运算符运算后的结果是boolean类型的值 true/false

》》 == 恒等于

》》 != 不等于

》赋值运算符： =

》》 += -= \*= /= %=



》逻辑运算符

》位运算符

》三元运算符

# 