

课前回顾:

介绍window基本常用命令

cd
dir dir/p
md
rd
del
copy

上面这些命令属于window系统中的内部命令,内置好的命令

notepad
calc
mspaint
mstsc

上面4个命令不属于window的内置命令

jdk中安装目录中bin目录中的java命令和javac命令javadoc命令

不属于window的内置命令,需要设置环境变量,就是为了能够在任何目录中都可以敲入java命令,首先在当前目录中寻找,如果寻找不到去定义的环境变量指定的路径中去寻找%JAVA_HOME%\bin
或C:\Program Files\Java\jdk1.8.0_65\bin

配置环境变量:

系统环境变量:

JAVA_HOME C:\Program Files\Java\jdk1.8.0_65
path ;%JAVA_HOME%\bin

在doc.tedu.cn

系统环境变量:

JAVA_HOME C:\Program Files\Java\jdk1.8.0_65
path ;%JAVA_HOME%\bin
classpath .;%JAVA_HOME%\lib

检验环境变量是否能用:

win+r 启动cmd命令行模式

java -version 查看当前系统下jdk的版本

javac 回车

环境变量配置成功,命令可用

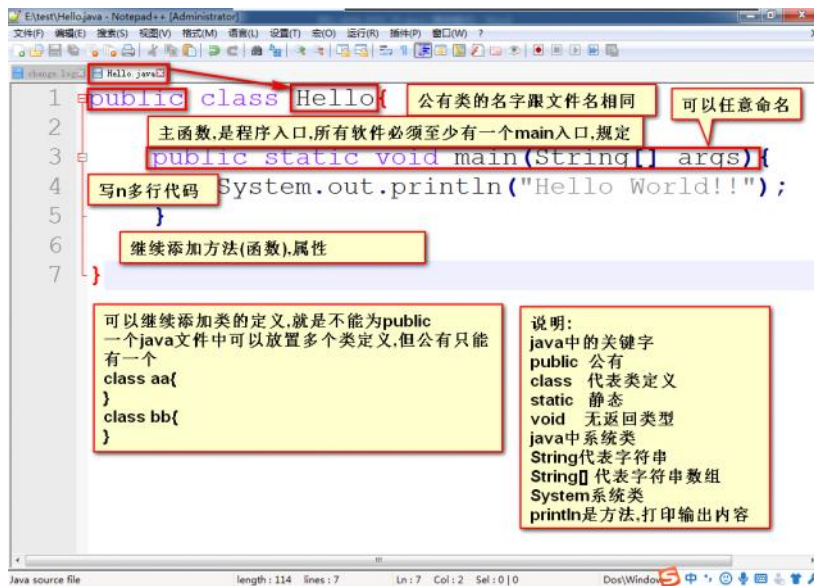
开始编写java代码:

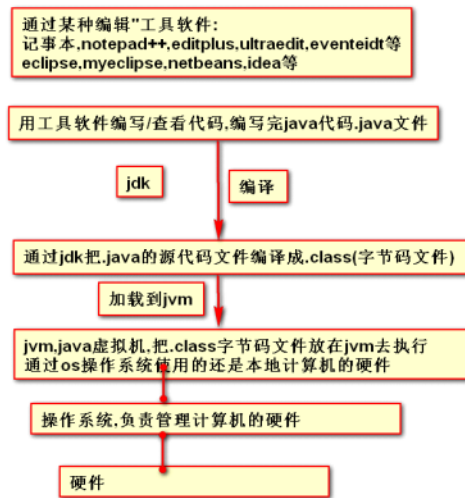
```
public class Hello{  
  
    public static void main(String[] args){  
        System.out.println("Hello World!!");  
    }  
  
}
```

在cmd中,用javac编译用java命令执行

javac Hello.java javac e:\test\Hello.java

java Hello 此时是执行的class字节码文件





Java中定义包(package)

java中叫做包package,包最终对应的是文件夹

package cn.tedu;//package是关键字,打出一个cn.tedu包

public class Hello{

```

    public static void main(String[] args){
        System.out.println("Hello World!!");
    }

```

}

带包编译和带包执行

javac -d 编译后的存放的目录 java文件名

javac -d E:\ex Demo.java---要编译的是Demo.java,编译之后放到E:\ex目录下

java -cp 编译后的存放的目录 包名.Class文件名

java -cp C:\ex\test cn.tedu.Demo --- class文件是Demo.class在cn.tedu包下, cn包再C:\ex\test目录下

总结:

0.前提必须有jdk+环境变量

1.编写java代码

2.用javac编译java代码为class字节码文件

3.用java命令执行字节码文件

4.带包 javac -d
java -cp

java中标识符:就是给某个东西起个名

java中的起名规则:

名称中(标识符中),可以有,字母(大小写),数字,_,_\$

就是以数字开头,不能是关键字,区分大小写

错的:

1ab ,public,a b,a%b

对的:

张三(不建议), Ab ,ab

命名方法:

驼峰命名法(骆驼命名法) 定义方法名,属性名

从第二单词开始所有的单词第一个字母大写,其余的小写

userName ,userPassWord

帕斯卡命名法: 定义类名字

所有单词的第一个字母都要大写

StudentClass , TestClass

用上面的命名规则

可以定义类名 帕斯卡

可以定义方法名 驼峰

可以定义变量名 int age; 驼峰

可以定义常量名 final double MATH_PI=3.14; 都要大写单词间用_间隔

可以定义域名 都是小写中间用.间隔 cn.tedu

注释:注释的东西,不是有效的java代码

//单行注释

```
/*  
    多行注释  
*/
```

```
/**  
    文档注释  
*/
```

数据在计算机(java)中的表示:

计算机/java最终需要底层硬件,

用二进制只有两个数 0 1 逢二进一

用八进制 有8个数 0-7 逢八进一

用十进制 有10个数 0-9 逢十进一

用十六进制 有16个数 0-9 a-f 逢16进一

做进制的目的,就是为了能快速,简短,准确的表达数据(二进制数据)

二进制	八进制	十六进制	十进制
0000	0	0	0
0001	1	1	1
0010	2	2	2
0011	3	3	3
0100	4	4	4
0101	5	5	5
0110	6	6	6
0111	7	7	7
1000	10	8	8
1001	11	9	9
1010	12	a	10
1011	13	b	11
1100	14	c	12
1101	15	d	13
1110	16	e	14
1111	17	f	15
10000	20	10	16

Java程序中表达二进制数据 数据的前面加上前缀0b/0B(jdk1.7及以上)

比如:

0b1010 0B1010

java程序中表达八进制数据 数据的前面加上前缀0

比如:

017+01=20

java程序中表达十六进制数据 数据的前面加上前缀0x/0X

比如:

0x1ab

进制转换:

十进制跟非十进制转换

除权(二,八,十六)取余数,倒着取

十 --> 二

十 --> 八

十 --> 十六

比如: 13十进制转二进制

$$\begin{array}{r} 2 \overline{)13} \\ 2 \overline{)6} \\ 2 \overline{)3} \\ 2 \overline{)1} \\ \hline 0 \end{array} \quad \begin{array}{c} 1 \\ 0 \\ 1 \\ 1 \end{array} \uparrow \quad 13 = 1101$$



本位乘以权位(二,八,十六)的次方序号

二-->十

八-->十

十六-->十

比如:

10011010110

10	9	8	7	6	5	4	3	2	1	0
1	0	0	1	1	0	1	0	1	1	0

$1 \times 2^{10} + 0 \times 2^9 + 0 \times 2^8 + 1 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$

非十进制(二,八,十六)之间的转换

二 <-----> 八 三个二进制位对应一个八进制位

比如:

100 101 101 111 二进制

4 5 5 7 八进制

二 <----> 十六

1001 0110 1111 二进制

9 6 f 十六进制

八 <----> 十六 借助二进制中转

计算机中的数据码制表示 原码 反码 补码

对于正数，原反补三码一致。

int i = 5;

原码：00000000 00000000 00000000 00000101

反码：00000000 00000000 00000000 00000101

补码：00000000 00000000 00000000 00000101

对于负数，反码是在原码的基础上最高位不变，其余位1变0，0变1；

补码在反码的基础上+1

注意：最高位是符号位，0表示正数，1表示负数

```
int i = -3;
```

原码：10000000 00000000 00000000 00000011

反码：11111111 11111111 11111111 11111100

补码：11111111 11111111 11111111 11111101

在java编程中如何表达数据:

在所有的程序中就做了三件事:

- 1.获取要运算的数据
- 2.运算获取的数据
- 3.把运算的结果显示或存储(持久化硬盘或数据库)

常量:量值是固定的

整型常量: 123 567

小数常量:1.23 56.7

字符常量:用单引号括起来的单个字符 'a' '1'

字符串常量:用双引号括起来的0个或多个字符

""

"abc_123"

"123"

布尔:只有两个值true 和false

true

false

空常量: null,只要看到是null,就是一个空值/空引用

带有final的量也是常量

```
final double MATH_PI=3.14D
```


变量:量值是变化的,就是存储数据的空间的内容是可以根据需求变化的
变量的定义

语法:声明一个变量

类型 变量名;

在java中一定要先定义变量,然后才能给变量赋值

比如:

先定义后赋值

```
int age; //声明一个变量
```

```
age=20; //给变量赋值
```

```
age=21;
```

在定义的同时就赋值

```
int weight=200;
```

分析变量的语法定义

类型 变量名; //为什么要这样定义一个变量

就是以变量名为理由,跟jvm管理的内存申请类型这么大空间

```
int age=20;
```

int类型在内存中占用4个字节

以age为理由跟内存申请4个字节空间,空间里存储的值是20这个常量值

以后可以通过变量名获取变量中的值,也可以把其他常量值赋值给

这个变量名指定的空间,把原来的值覆盖掉

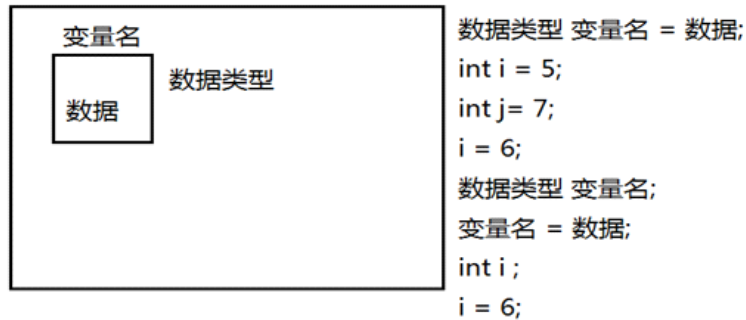
结论:

用变量和常量存储数据,可以根据这些数据,结合业务要求,做运算

内存中如何表达常量和变量

是通过变量名和常量名来表达的,

通过变量名和常量名在内存中寻址(内存地址,二进制)



java中的八个基本数据类型:

byte:
short:
int
long
float
double
char
boolean

所有语言都有基本数据类型,java也有八中基本数据类型

java规定了跟内存申请空间多大,空间中的数据就会有范围了

byte 1 个字节 范围 $-2^7 \sim 2^7-1$

byte b=129;

Type mismatch: cannot convert from int to byte

类型 不匹配 不能 转换 从 int 到 byte

short 短整形 2个字节 表示范围是 $-2^{15} \sim 2^{15}-1$ --- -32768~32767

int 整型 4个字节 表示范围是 $-2^{31} \sim 2^{31}-1$ --- $-2.1 \times 10^{10} \sim 2.1 \times 10^{10}$

long 长整型 8个字节 表示范围 $-2^{63} \sim 2^{63}-1$ --- $-9 \times 10^{18} \sim 9 \times 10^{18}$

后边添加L/l标志是一个长整型: long l = 31864853188L;

float 单精度浮点 占4个字节 表示范围 $-10^{38} \sim 10^{38}$

要求必须以F/f标志float类型的数字,

float f = 3.23f;

double 双精度浮点 占8个字节 表示范围 $-10^{308} \sim 10^{308}$

小数默认为double

System.out.println(3.4);

```
double d = 4.2;
double d2 = 5.6D;
double d3 = 6.01d;
```

char---字符型---占2个字节--- $0 \sim 2^{16}-1$ --- $0 \sim 65535$

```
char ch='a'; char ch='中';
```

转义字符： '\t' ---table 制表符 '\r'---return 回车 '\n'---
next/newLine换行

'\\' ---\ '\'' --- 单引号 '\"' --- 双引号

boolean 布尔型 只能表示true/false

内存大小根据jdk版本和操作系统来确定

```
boolean b = true;
```

总结:用八种数据类型可以存储指定类型的数据

补充内容:

用eclipse做开发工具开发java代码:

首先去www.eclipse.org下载eclipse的的安装包(选java EE版本)

下载后解压eclipse的压缩包,路径中不要带中文和空格

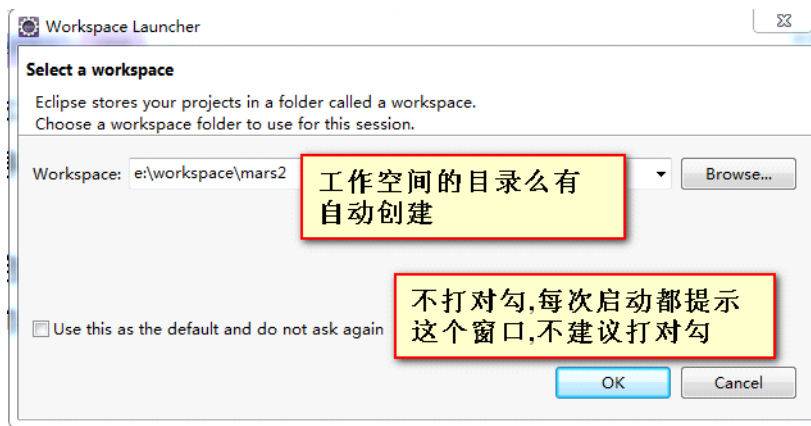
解压后的目录中有eclipse.exe的启动文件,启动eclipse

启动eclipse的时候,必须先安装好jdk,在启动eclipse的时候,

自动寻找本机的jdk,如果有引用,然后正常启动eclipse,

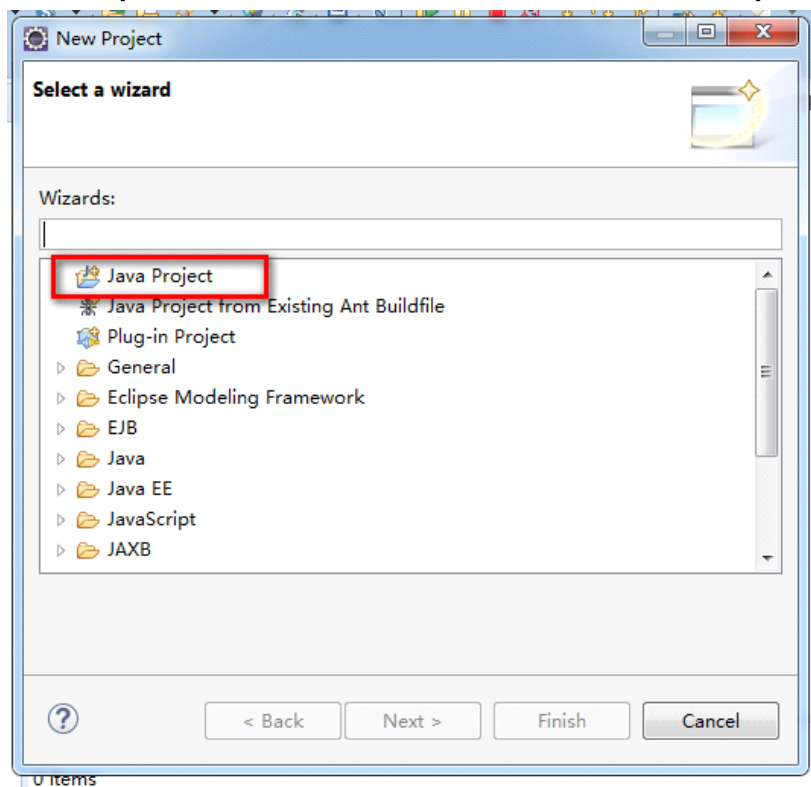
如果没有寻找到,会eclipse不能启动

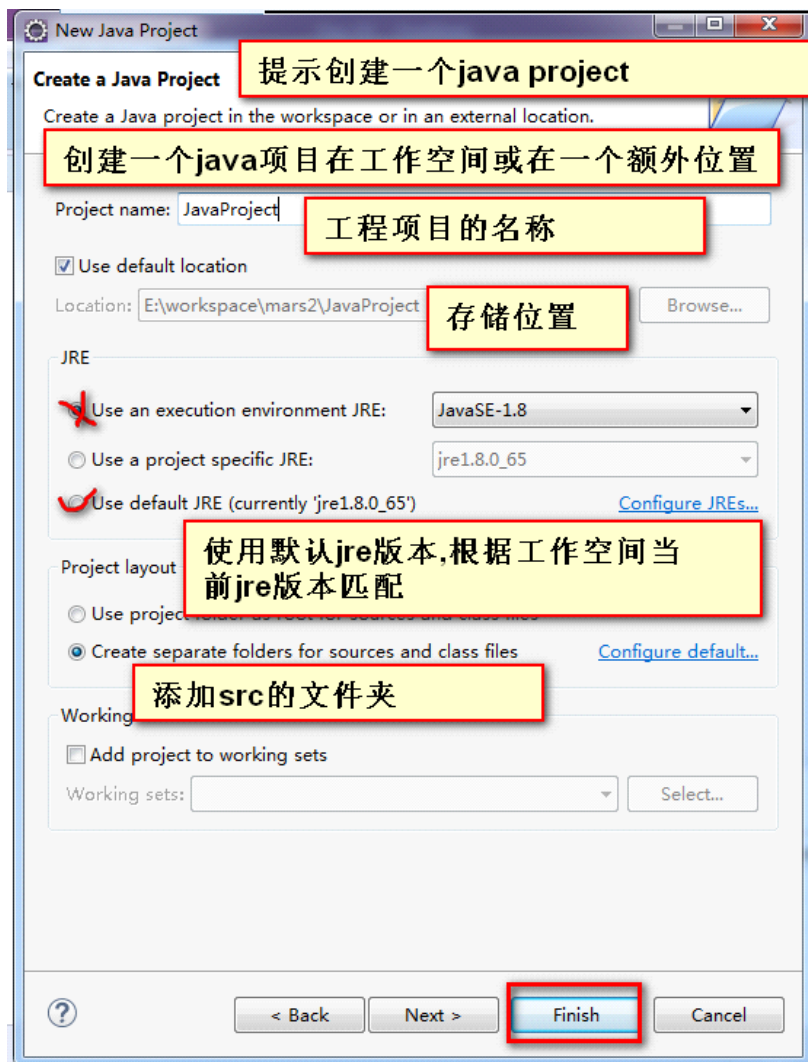
如果正常启动eclipse,会提用户指定工作空间(workspace),这个workspace
存储的当前eclipse中所有的项目源代码



点击"ok" 正常启动eclipse

在Project explorer 中右键单击空白处---new--project(工程或项目)





项目创建完后

在src创建包package,

在包中可以创建若干java类

给eclipse设置全部项目为utf-8

window --- preferences(首先项)---general---

workspace---test file encoding--- utf-8