# Day01 JAVA语言基础

## 1常见window命令：

### cmd窗口

打开开始菜单，在搜索框中输入cmd命令，点击cmd进入cmd窗口，或者打开我的电脑，在路径框输入cmd，进入cmd窗口。

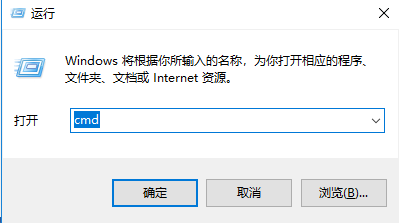


图-1

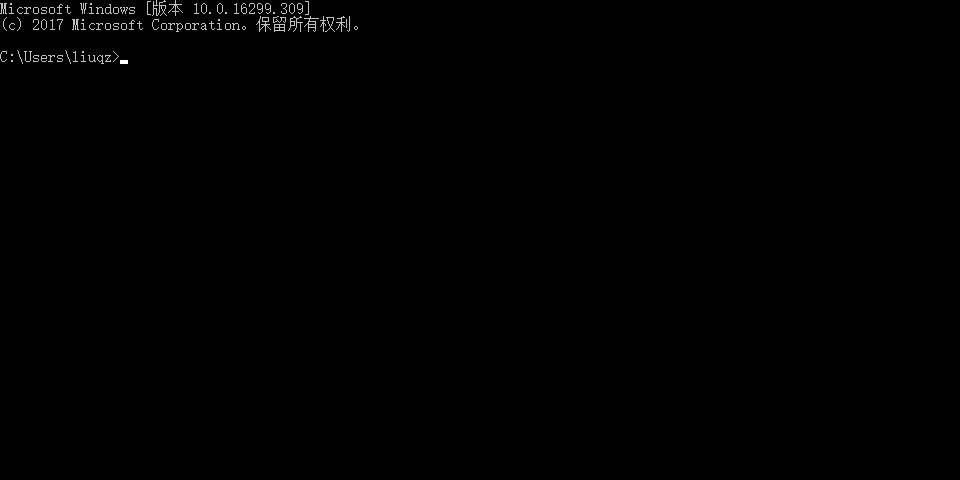


图-2

### 常见命令

cd ---打开目录--- .. 表示上一层目录 . 表示当前目录 /表示的是根目录

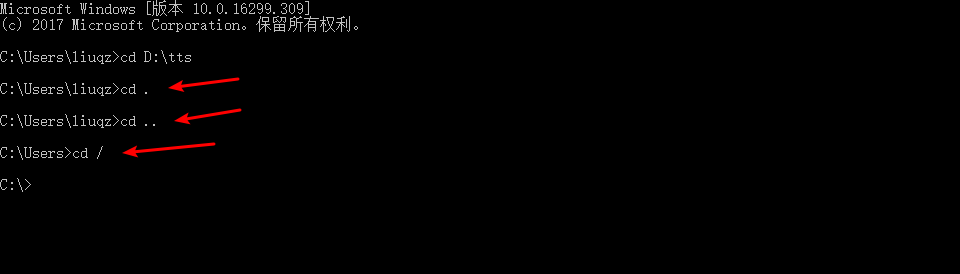


图-3

dir---展现指定目录下的所有的子文件和子目录---directory



图-4

mkdir---新建目录---md

创建之前：

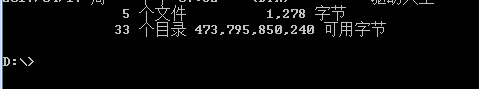


图-5

创建：



图-6

创建完后：



图-7



图-8

rmdir---删除目录---从计算机中彻底移除，不可撤销---rd---要求删除的目录必须为空目录

执行：



图-9

执行完：



图-10

del---删除文件



图-11

cls---清空屏幕---clear screen



图-12

exit---退出命令提示符



图-13

notepad---打开记事本



图-14

calc---打开计算器



图-15

mspaint---打开画图



图-16

## Java的介绍

Java是1995年由SUN公司推出的一门高级编程语言，面向互联网的编程语言，是WEB应用程序首选的语言，Java语言的特点是简单易学，安全可靠，跨平台。

JAVA之父：James Gosling

## java的技术结构

J2SE/JAVASE---标准版---为一些小应用程序与桌面程序提供了解决方案

J2EE/JAVAEE---企业版---为企业中项目的开发提供了方案

J2ME/JAVAME---移动版---主要是嵌入一些小型电子设备中，实现移动端的开发

## Java跨平台的原理：

Java跨平台的实现，基于Java虚拟机---Java Virtual Mechine---JVM。所有的Java程序是运行在Java虚拟机上的，Java虚拟机屏蔽了不同操作系统之间的差异性，使得相同的Java程序能够在不同的操作系统上运行，从而实现了Java语言的跨平台。

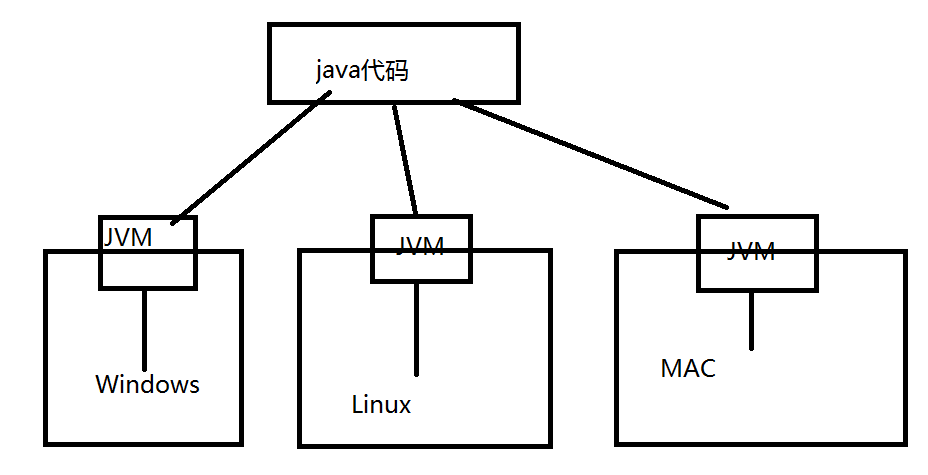


图-17

### JDK、JRE、JVM的介绍

JVM---Java Virtual Machine---是Java语言能够跨平台的条件

JRE---Java Runtime Environment --- Java运行时环境 --- JVM+核心类库

JDK---Java Development Kit --- Java开发工具包 --- 开发工具+JRE

注意：Java语言是跨平台的，但是Java虚拟机不是跨平台的

## JDK的下载和安装

下载网址：[www.oracle.com](http://www.oracle.com)

安装：安装的时候不建议安装到C://program files目录下

注意：安装路径中不要出现空格和中文

## 环境变量的配置

### 找到我们安装的JDK的目录，并复制此目录。如图-

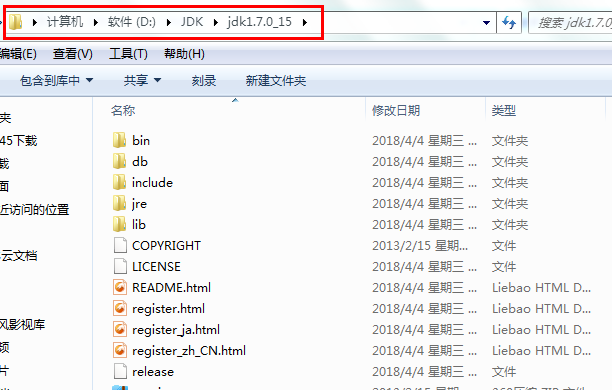


图-18

### 右击我的电脑，点击属性。

### 选择高级系统设置。如图-

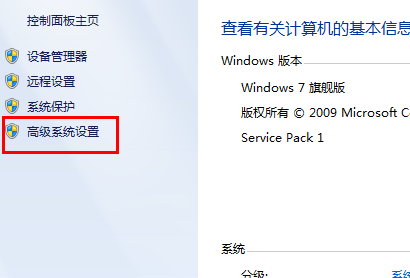


图-19

### 选择环境变量

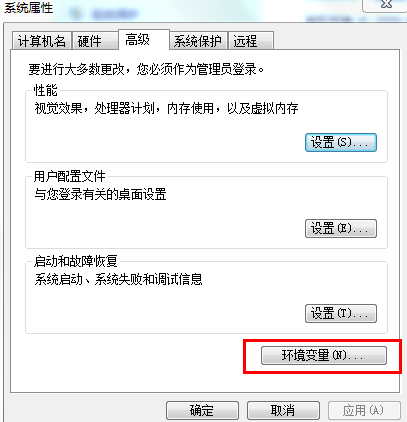


图-20

### 在系统变量处选择新建



图-21

### 在标红处填写我们刚刚复制的JDK的路径

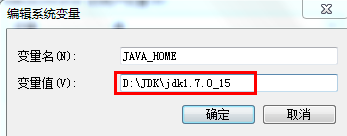


图-22

### 在名为Path的变量上添加我们配置的JAVA\_HOEM，格式为%JAVA\_HOME%\bin；注意不要丢掉分号。



图-23

### 在cmd窗口输入javac命令，出现如下格式，表示成功。



图-24

\*编译：将Java代码翻译成当前操作系统所理解的指令---编译---编译完成之后会产生一个class文件，这个文件称之为字节码文件---这个字节码文件就是给当前操作系统看的指令文件。在cmd窗口下，执行javac 文件名.java 来实现编译

## 入门程序

在任意盘符下右击，新建一个文本文档，将后缀名改成.java，然后在文档中编写以下内容。

在文件目录下打开cmd窗口，在配置好环境变量的情况下，使用javac 类名.java 进行编译，产生class文件，然后使用 Java 文件名（不加后缀） 来运行程序，发现窗口会出现HelloWorld！。

class Demo{

public static void main(String[] args){

System.out.println("Hello World !");

}

}

--主函数public static void main(String[] args){}---是程序的入口，没有主函数，程序可以编译，但是不能运行。

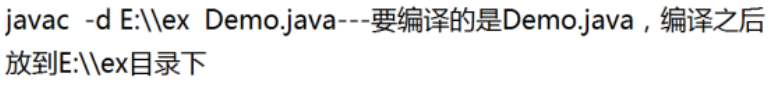
--7.2class文件名和类名是对应的---每一个类在编译完成之后会产生一个class文件

--如果一个类用public来修饰（公共类），那么要求这个类名和Java 文件名要一致

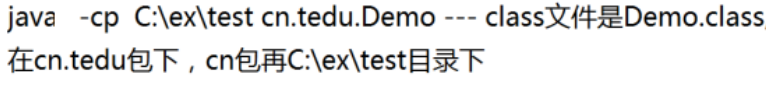
--一个Java文件中可以定义多个类，但是只能有一个公共类

--在编译完.java文件之后，如果代码中添加了包名，在运行的时候需要先将.class文件放到对应的包的目录中，然后再执行 java 包名.类名 来运行程序。

--带包编译：javac -d 编译之后的包存放的目录 要编译的Java文件 编译完成之后，生成带文件目录的.class文件。如图-：



--执行其他盘符下的java程序：java -cp class文件存放的路径 包名.类名，如图：



## 关键字

有特殊含义的单词

在Java中一共有53个关键字---其中有2个关键字目前没有使用：goto，const---保留字，关键字都为小写。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 用于定义数据类型的关键字 | | | | | | | |
| class | interface | byte | | short | int | | long |
| float | double | char | | boolean | void | | enum |
| 用于定义数据类型值的关键字 | | | | | | | |
| true | false | null | |  |  | |  |
| 用于定义流程控制的关键字 | | | | | | | |
| if | else | switch | | case | default | | while |
| do | for | break | | continue | return | |  |
| 用于定义访问权限修饰符的关键字 | | | | | | | |
| private | protected | public | |  |  | |  |
| 用于定义类、函数、变量修饰符的关键字 | | | | | | | |
| abstract | final | static | | synchronized | |  |  |
| 用于定义类与类之间关系的关键字 | | | | | | | |
| extends | implements | |  |  |  | |  |
| 用于定义建立实例、判断实例的关键字 | | | | | | | |
| new | this | super | | instanceof |  | |  |
| 用于异常处理的关键字 | | | | | | | |
| try | catch | finally | | throw | throws | |  |
| 用于包的关键字 | | | | | | | |
| package | import |  | |  |  | |  |
| 其他修饰符关键字 | | | | | | | |
| native | strictfp | transient | | volatile | assert | |  |

## 标识符

9.1 定义：在程序中我们自己定义的名称

9.2 命名规则；由字母、数字、\_、$组成,但是不建议使用$；Java支持中文命名，但是不建议使用；数字不能开头；不能使用关键字；尽量见名知意。

9.3 驼峰命名法：

类名/接口名：如果由多个单词组成，要求每一个单词的首字母大写 --- PlayBasketball,Demo

变量名/方法名：如果由多个单词组成，第一个单词的首字母小写，其余单词的首字母大写 --- playBasketball,work

常量名：所有单词全部大写，每一个单词之间用\_隔开 --- PLAY\_BASKETBALL ,SAM

包名：所有单词全部小写，每一层包之间用.隔开 --- cn.tedu

Java是大写小敏感的语言，所以只要两个名称有一点不同就是不同的标识符

## 1.10注释

10.1 定义：在程序中用来解释说明程序的文字。

10.2 格式：

单行注释 //注释文字

多行注释 /\*注释文字\*/

文档注释 /\*\*注释文字\*/---文档注释中的文字可以利用工具或者命令进行提取。

javadoc 要提取的Java源文件---注意，一个类如果能够被提取，必须是公共类。

10.3 作用：解释程序；便于阅读、维护；排错。