

day01-开发环境和第一个程序

今日学习内容

计算机语言

跨平台原理

开发和运行环境的搭建

HelloWorld程序开发

IDEA开发工具

Java的基本语法

数据类型和常量

今日学习目标

了解什么是计算机语言

了解什么是java的跨平台原理

掌握如何搭建开发和运行环境

必须掌握HelloWorld程序的开发

必须记住Java的基本语法

了解注释符号有哪些

了解什么是关键字和保留字

必须记住java的分隔符和标志符有哪些

熟悉使用IDEA工具来开发

掌握什么是常量，以及常量的分类

必须掌握数据类型有哪些

1. Java入门基础上

1.1. Java和跨平台

1.1.1. 计算机语言和编程概述（了解）

在生活中的两个人需要交流，无非是采用一种彼此都能够识别的语言。那么，我们说该语言是他们传递信息的媒介。

那么什么是计算机语言呢？计算机语言是指用于人与计算机之间通讯的一种特殊语言，是人与计算机之间传递信息的媒介。

为什么需要和计算机交流呢？计算机怎么能读懂我们给它发出的信息？

和计算机交流的目的，就是让计算机帮我们完成复杂工作（重复性、程序化），比如大量数据的运算。为了让计算机能读懂我们发出的信息，此时就需要编写一套由字符、数字所组成并按照某种语法格式的一串串计算机指令，而这些指示和命令就是计算机语言。

1.1.2. 什么是编程（了解）

计算机语言就是用来实现人和计算机通讯的，那为什么人要和计算机通讯呢，其原因就是为了让计算机帮我们完成一些人为起来比较复杂的工作。

那计算机怎么知道我们要它解决的问题是什么，怎么知道解决问题的具体的步骤是什么呢？

此时我们就得通过编程语言去告诉计算机去：**需要做什么，怎么一步一步去做**。这种人和计算机之间交流的过程，我们称之为编程。

思考：我们初学者该如何学编程？

1.1.3. Java语言

Java诞生于1995年，原属于SUN公司，2009年4月20日，美国数据软件巨头甲骨文公司（*Oracle*）宣布以74亿美元收购SUN公司。Java是最受欢迎的开发语言，已经火了20年，并将继续引领着IT的编程语言。Java的LOGO是一杯热气腾腾的咖啡，真的是令人回味无穷。



Java的三大平台（jdk1.2开始）：Java SE、Java EE，Java ME

- Java SE: Java标准平台，它允许开发软件运行在电脑桌面上(用户需要下载，安装)，最经典的JavaSE程序就是Java的开发工具Eclipse，同时JavaSE也是JavaEE的基础。（JavaSE适合做桌面端应用）



- Java EE: Java企业平台，针对**Web方向**，主要应用于开发企业项目和互联网项目，如淘宝，京东，12306，各大银行网站等。（JavaEE 适合做服务器端开发，主要针对Web方向的企业项目和互联网项目）

淘宝网
Taobao.com

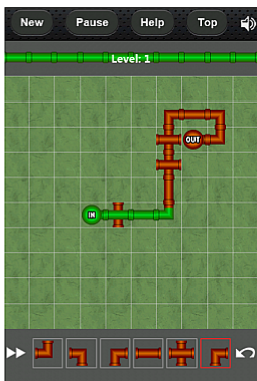
苏宁易购
suning.com



京东



- Java ME（了解）：Java微型平台，用来开发移动设备上的软件，比如早期功能机上的Java游戏，当然现在开发手机软件使用iOS和Android平台和技术。



Java在移动端的表现，现在我们使用的手机App，主要是iOS和Android平台的，比如说淘宝App，无论你是iOS手机还是Android手机，在你手机上的App仅仅是一个页面，所有的业务功能都是有后台Java完成实现的，后台和App之间通过接口就可以交互数据了。

你会发现，这个世界除了空气无处不在之外，还有Java。

Feb 2020	Feb 2019	Change	Programming Language	Rating	Change
1	1		Java	17.358%	+1.48%
2	2		C	16.788%	+4.34%
3	3		Python	9.345%	+1.77%
4	4		C++	6.164%	-1.26%
5	7	▲	C#	5.927%	+3.08%
6	5	▼	Visual Basic .NET	5.862%	-1.23%
7	6	▼	JavaScript	2.060%	-0.79%
8	8		PHP	2.018%	-0.25%
9	9		SQL	1.526%	-0.37%
10	20	▲▲	Swift	1.460%	+0.54%
11	18	▲▲	Go	1.131%	+0.17%
12	11	▼	Assembly language	1.111%	-0.27%
13	15	▲	R	1.005%	-0.04%
14	23	▲	D	0.917%	+0.28%
15	16	▲	Ruby	0.844%	-0.19%
16	12	▼▼	MATLAB	0.794%	-0.40%
17	21	▲	PL/SQL	0.764%	-0.05%
18	14	▼▼	Delphi/Object Pascal	0.748%	-0.32%
19	13	▼▼	Perl	0.697%	-0.40%
20	10	▼▼	Objective-C	0.688%	-0.76%

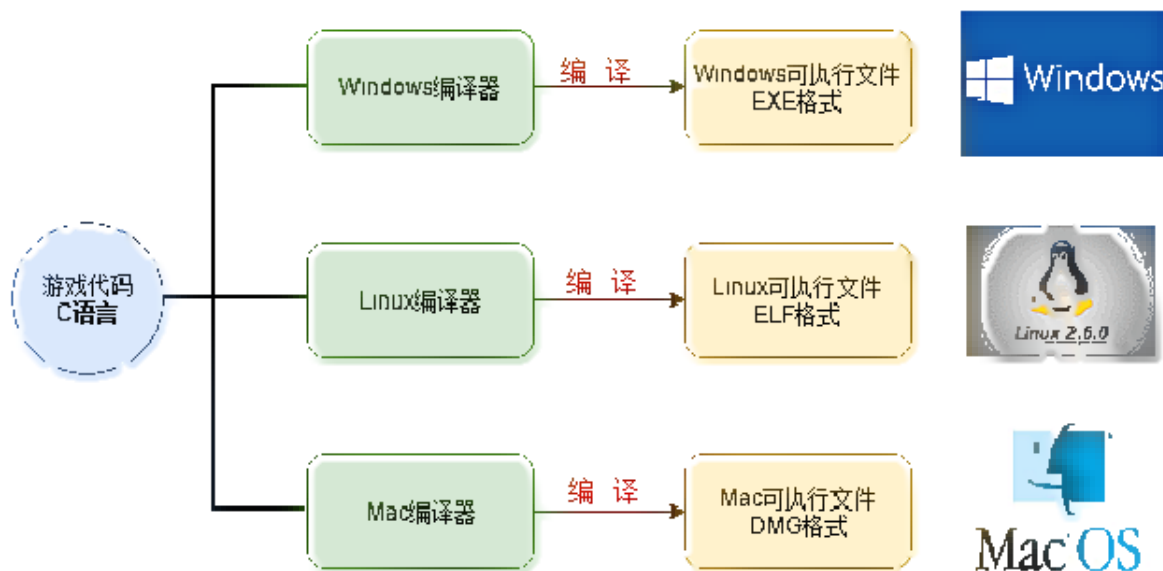
1.2.跨平台原理（了解）

1.2.1. 平台相关性（掌握）

Windows系统只能运行Windows的程序（exe），Linux系统只能运行Linux的程序，Mac系统只能运行Mac的程序。

Windows的可执行文件，不能直接运行在Linux系统中，反之亦然，就好比Android手机不能运行iPhone的App程序一样，我们把这种情况称之为平台相关性。

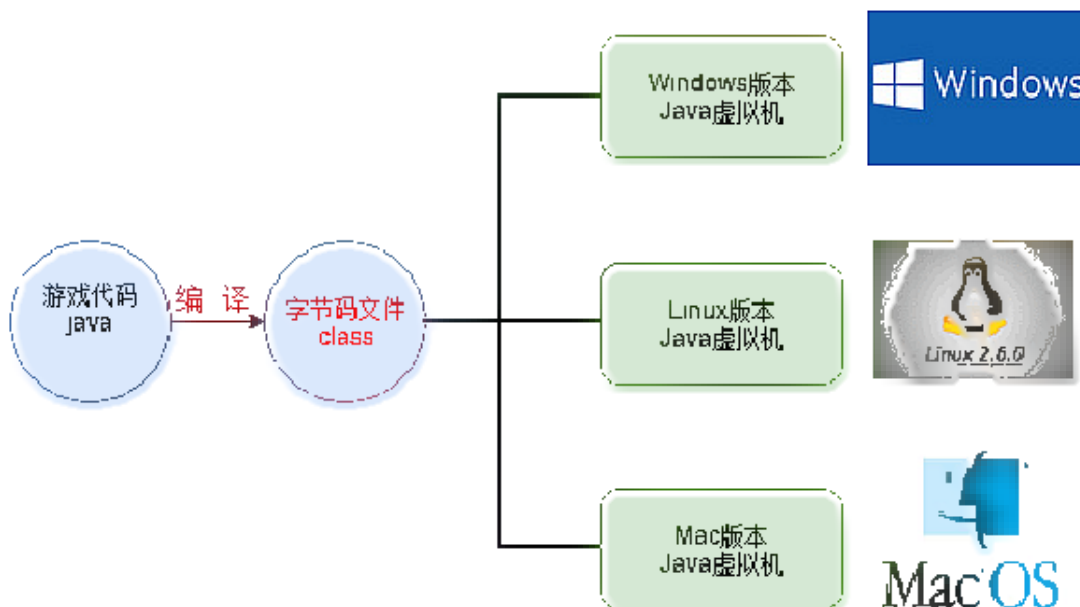
1.2.2. 跨平台性（掌握）



一份（源）代码，多次编译，到处运行

使用特定平台的编译器所编译的程序只能在对应的平台运行，此时会发现编译器是与平台相关的，编译后的文件也是与平台相关的。我们说的语言跨平台是编译后的文件跨平台，而不是源程序跨平台。

那语言的跨平台性如何实现？拿Java举例，我们可以对Java程序进行编译操作，编译后生成一种和平台系统无关的文件——字节码文件。但是此时Windows、Linux是不能执字节码文件的，只有Java虚拟机（JVM）才能识别字节码文件，那么为了在Windows系统上运行该Java程序，我们就只能在Windows平台上安装Windows版本的JVM，如果要在Mac系统上运行，那么得安装Mac版本的JVM。

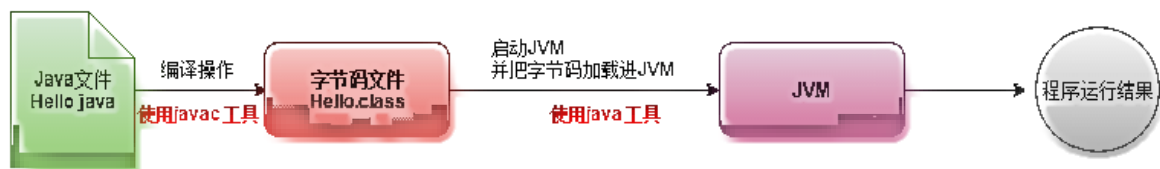


那么如此一来，Java就实现了跨平台，也就达到了“一次编译，到处运行”的效果。

Java之所以能跨平台有两个原因：

- Java文件经过编译后生成和平台无关的class文件
- Java虚拟机（JVM）是不跨平台的

在这里进行编译操作的工具叫做javac，启动JVM并把字节码加载进JVM的工具叫做java。

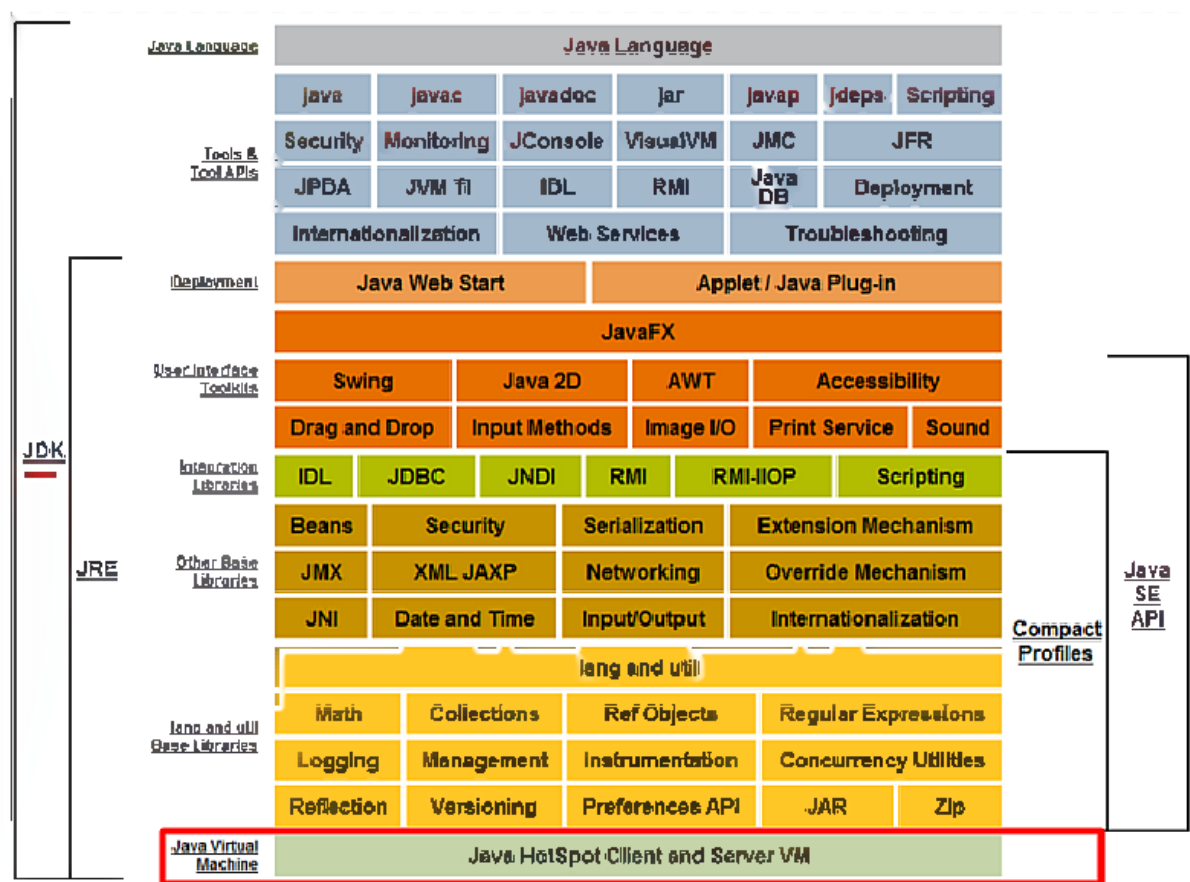


这里解释几个术语：

- 源代码：简称源码，是指还未编译的按照编程语言规范书写的代码，比如Java代码
- 源文件：存放源代码的文件，比如Java文件，拓展名是.java
- 字节码文件：经过编译器预处理过的一种文件,是JAVA的执行文件存在形式，拓展名是.class

1.3. Java开发和运行环境搭建（重点）

1.3.1. JDK和JRE概述（了解）



- JRE(Java Runtime Environment): Java运行环境，如果要运行Java程序，就需要JRE的支持，JRE里包含JVM，一般在只运行程序而不开发程序的服务器中安装。
- JDK(Java Development Kit): Java开发工具，包含开发Java程序的所有工具如javac和java等。JDK包含JRE，如果已经安装了JDK就不必安装JRE。
- JVM (Java Virtual Machine) :Java虚拟机，它是运行所有Java程序的虚拟计算机。JVM是不跨平台的，在Windows下装Windows版的JVM，在Linux下装Linux版的JVM。

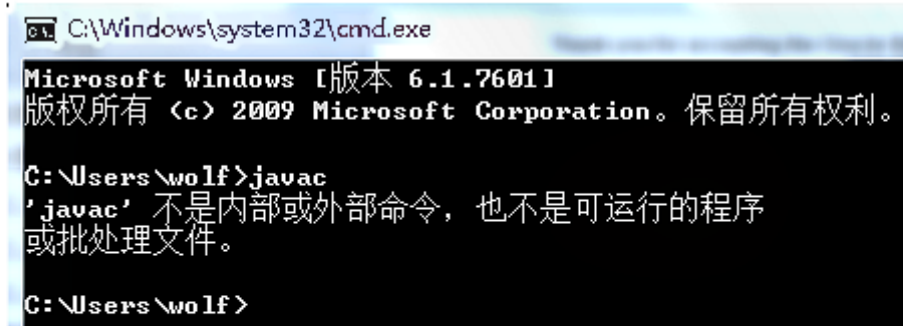
jvm:java虚拟机, 只能用于直接xx.class字节码文件
jre:java程序的运行环境, jre = jvm + 常用类库
jdk:java程序开发环境, 开发过程中的源代码需要编译后给jre运行。jdk = javac等开发工具 + jre

1.3.2. 安装JDK (掌握)

选择和操作系统一致版本的JDK, 如window 64位选择【jdk-8u251-windows-x64.exe】,

mac选择【jdk-8u251-macosx-x64.dmg】

当出现以下界面的时候, 说明当前系统不具备Java的开发环境, 就得去安装JDK和配置环境变量。



选择适合自己电脑的JDK后, 全称傻瓜式的下一步, 其中需要注意的是:

- JDK不要安装在C盘, 且不要带中文的路径, 路径中不包含空格字符
- 安装完JDK后, JRE不用安装
- 如果C:\WINDOWS\system32目录中出现名字以java开头的文件, 有, 删之

举例说明:

正例: D:\OpenSources\jdk1.8.0_45\bin

反例: D:\Java开发\jdk1.8.0_45\bin 出现中文

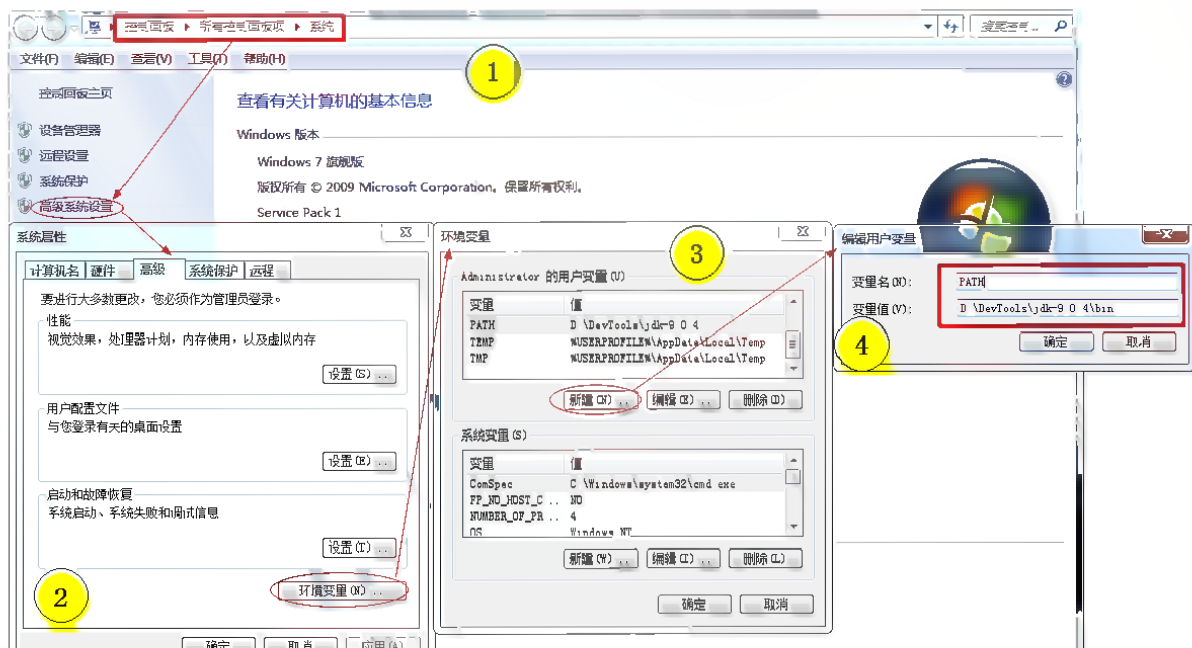
反例: D:\Program Files\jdk1.8.0_45\bin 路径中出现空格

注意: 在JDK安装目录下的bin目录中, 存在编译工具 (javac) 和运行工具 (java) 。

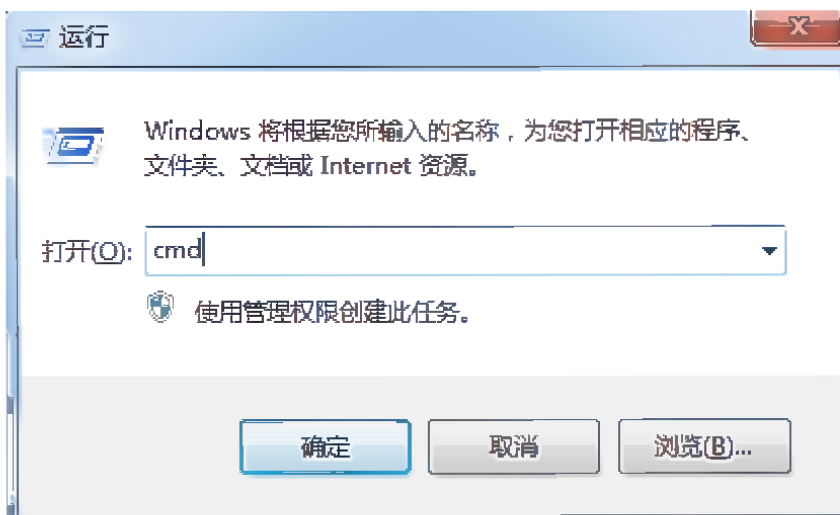
1.3.3. 配置PATH环境变量 (掌握)

配置PATH环境变量的目的是, 能够在任何地方使用编译工具 (javac) 和运行工具 (java) 。

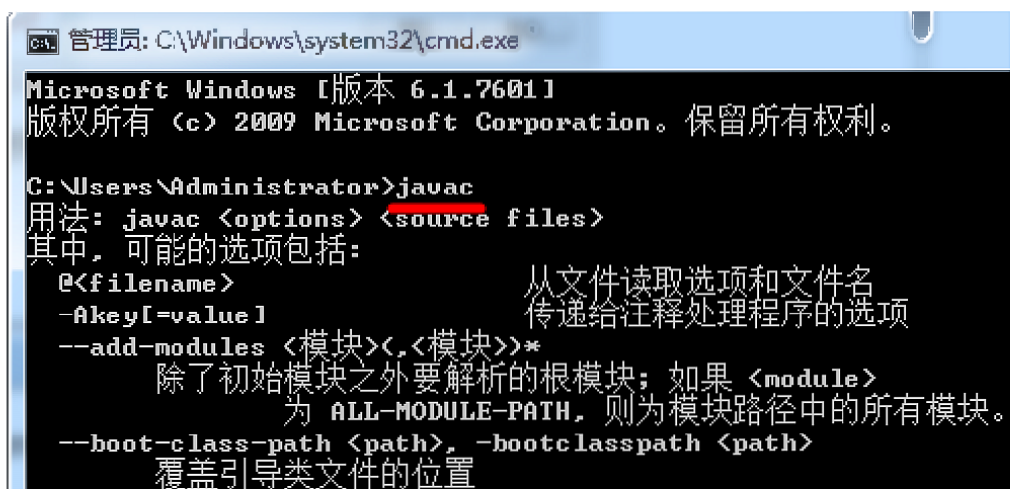
PATH环境变量的值就应该是javac和java工具所在的目录路径。



再点击开始菜单的运行工具（或者使用快捷键Win+R），弹出运行工具界面，输出cmd再敲回车：



此时会弹出一个黑色界面的控制台界面，在里面输入javac，并回车。



如果看到上述界面说明PATH配置成功，此时就可以开发Java程序了。

常见问题：

- 忘记配置PATH环境变量
- PATH环境变量没有指向JDK下面的bin目录

- 出现javac不是内部或者外部命令的错误，原因是path配置错误

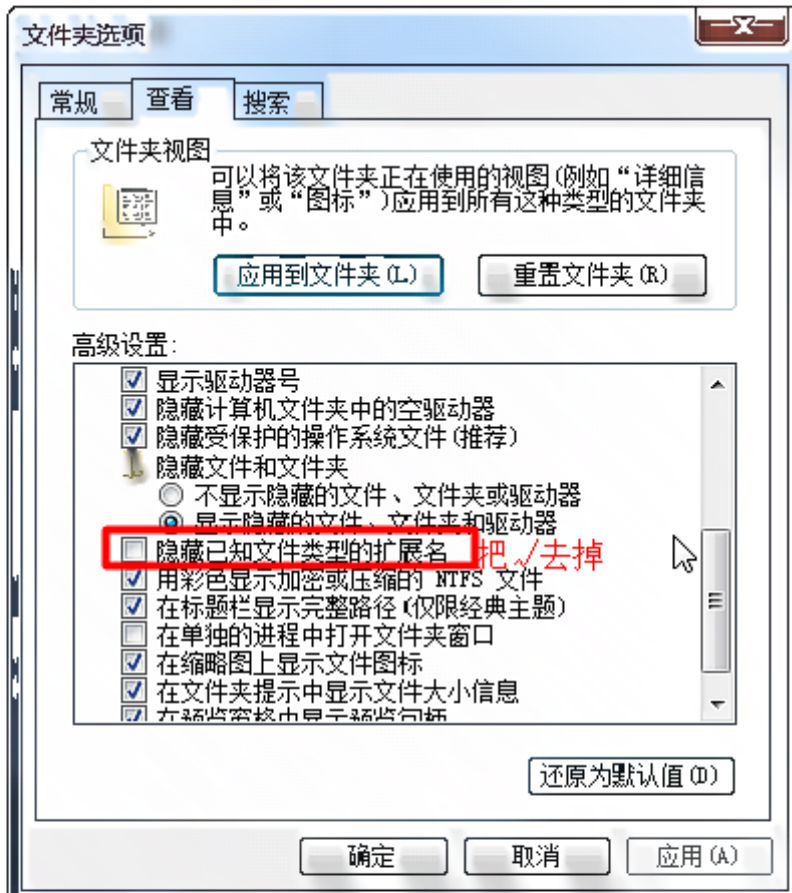
1.4. Java的第一个程序（重点）

一般而言，编写的第一个程序，习惯叫做HelloWorld程序。

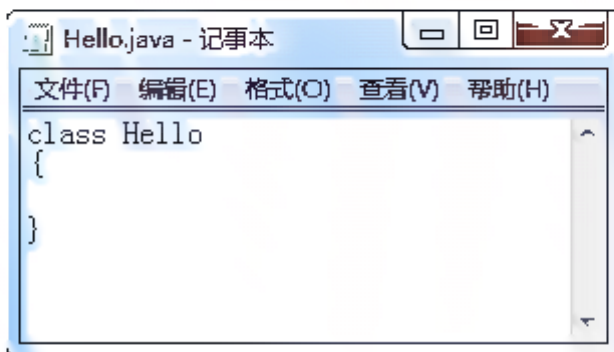
需求：在控制台界面，输出：你好师姐

1.4.1. HelloWorld程序开发（重点）

1. 新建文本文档，起名为Hello.java，此时需要在文件夹选项中去掉隐藏文件类型的拓展名。



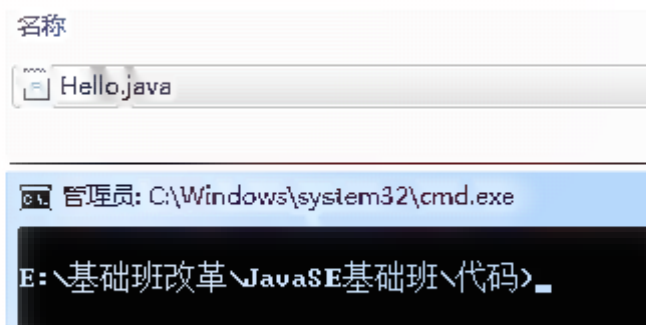
2. 使用记事本或Editplus或Notepad++等文本编辑器，打开Hello.java文件，输入以下代码



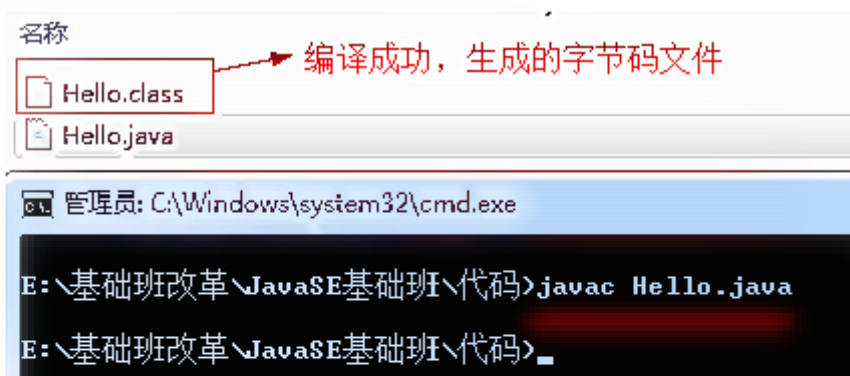
3. 在当前源文件所在目录，按住Shift，鼠标右键点击在次数打开命令窗口



4. 然后，进入类似如下界面



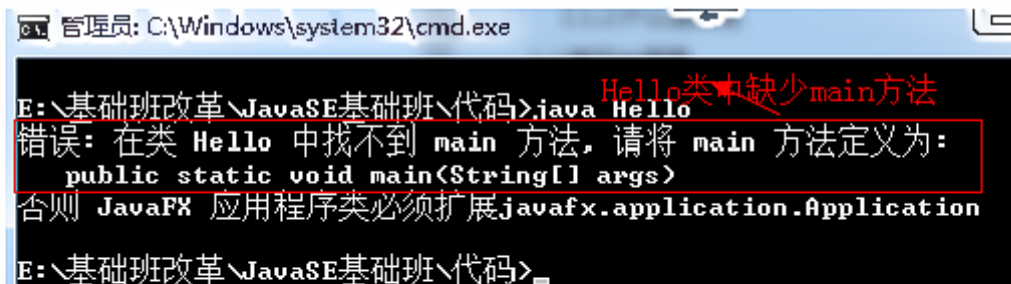
5. 此时使用javac工具，对Hello.java文件做编译操作，语法是：javac 文件名.java



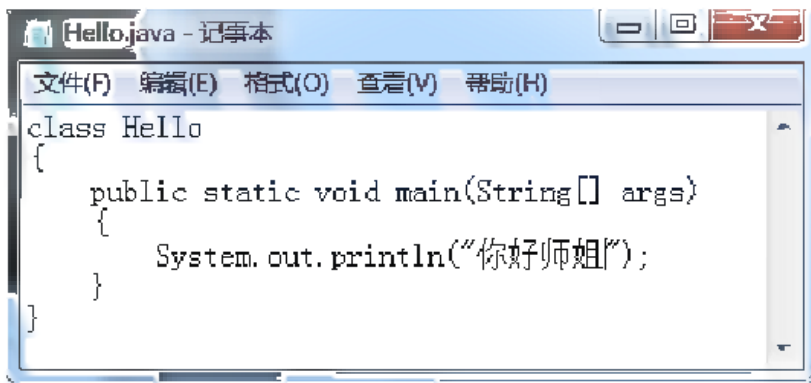
6. 编译成功之后，会生成Hello.class文件（字节码文件），如果编译失败，不会生成字节码文件。

当有字节码文件后，此时可以启动VM，并加载Hello这份字节码并执行，语法是：java 类名

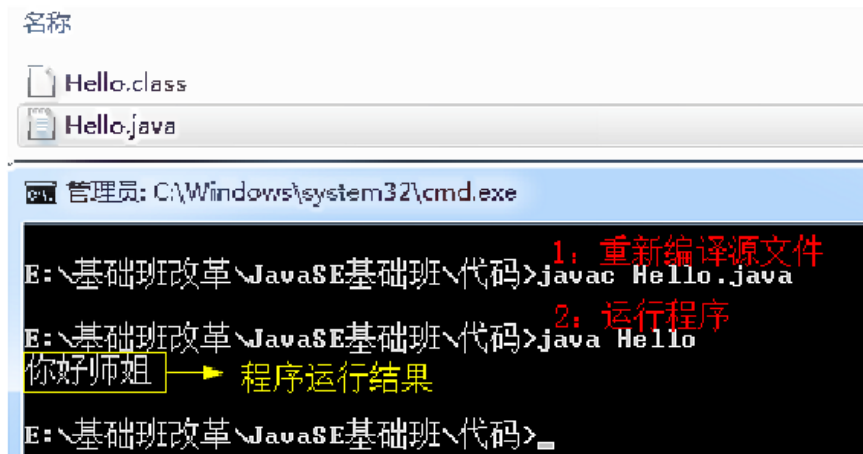
- 注意1：运行的命令是java Hello ,而不是 java Hello.class
- 注意2：被运行的类Hello中必须有main方法，否则报错



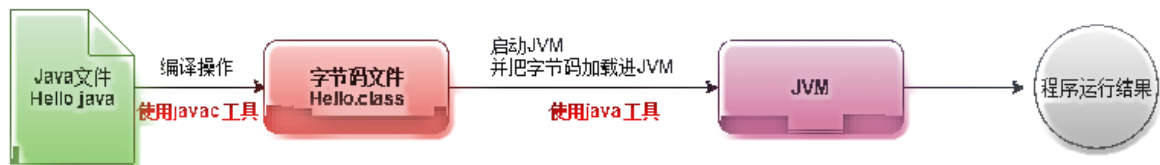
7. 此时重新修改Hello.java文件，在类中添加main方法的代码，如下：



8. 修改源文件之后，需要重新对源文件做编译操作，编译后生成新的字节码文件会覆盖旧的字节码文件



1.4.2. Java的编译和运行机制（重点）



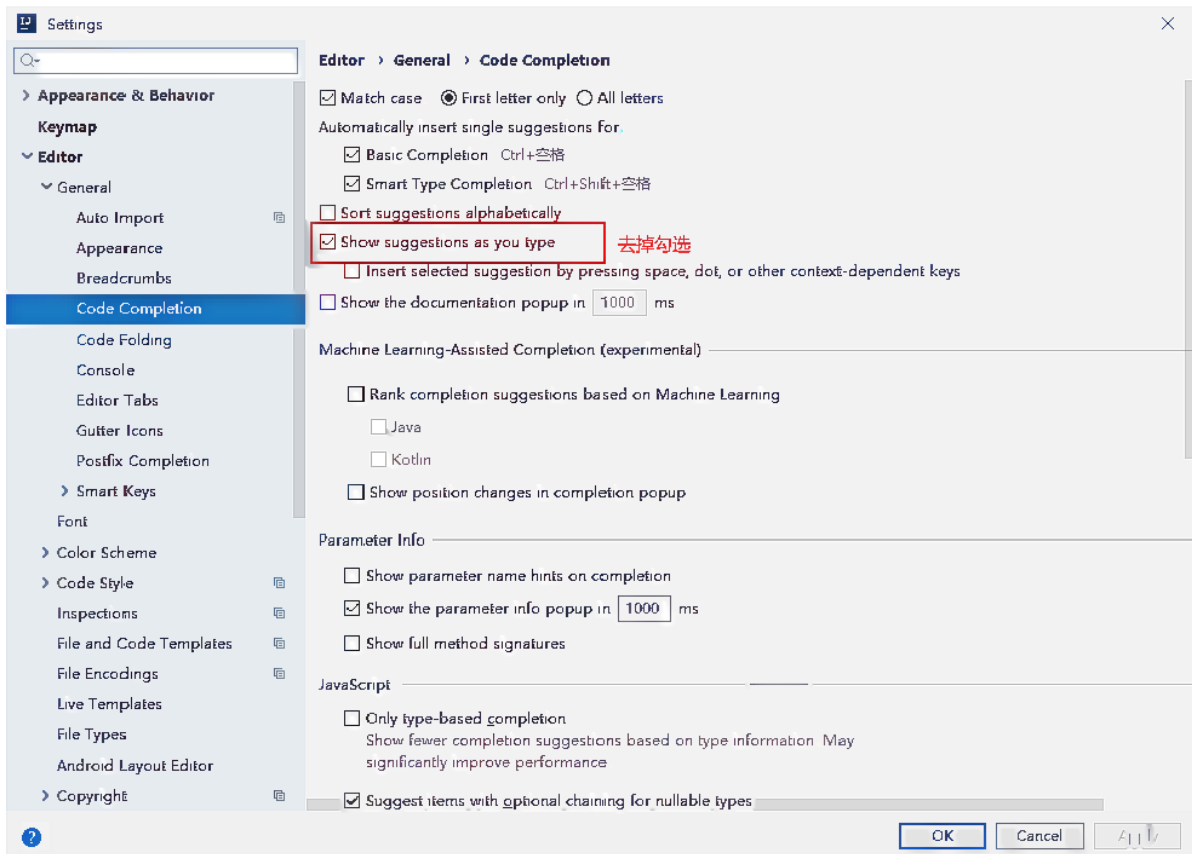
- 编写源文件（Java文件），源文件中包含源代码（Java代码）。
- 使用javac工具对源文件做编译操作，语法：javac 源文件.java 语法检查
- 生成字节码后，使用java工具启动JVM，运行程序，语法：java 拥有主方法的类名
- 在控制台输出，显示结果

1.5 开发工具-IDEA

IDEA 全称 IntelliJ IDEA，是java编程语言开发的集成环境。IntelliJ在业界被公认为最好的java开发工具，尤其在智能代码助手、代码自动提示、重构、JavaEE支持、各类版本工具(git、svn等)、JUnit、CVS整合、代码分析、创新的GUI设计等方面的功能可以说是超常的。IDEA是JetBrains公司的产品，这家公司总部位于捷克共和国捷克共和国/418555)的首都布拉格，开发人员以严谨著称的东欧程序员为主。它的旗舰版本还支持HTML，CSS，PHP，MySQL，Python等。免费版只支持java等少数语言。

根据文档《IDEA安装配置文档.pdf》安装配置好IDEA，在后面的课程中，我们将选择使用IDEA完成教学。

为了形成良好的指感，早期学习建议大家不使用智能提示！！！！



1.6. Java的基本语法（重点）

1.6.1. 基本语法（必须记住）

- **Java语言严格区分大小写。**好比main和Main、system和System是完全不同的概念。
- 一个Java源文件里可以定义多个Java类，但其中最多只能有一个类被定义成public类。若源文件中包括了public类，源文件必须和该public类同名。
- 一个源文件中包含N个Java类时，成功编译后会生成N份字节码文件，即每个类都会生成一份单独的class文件，且字节码文件名和其对应的类名相同。
- 若一个类必须运行，则必须拥有main方法（主方法），因为main方法是程序的入口。

1.6.2. 注释符号（了解）

1.6.2.1 三种注释方式（掌握）

Java的注释信息是给程序员看的，编译器(javac)在编译的时候会忽略掉源文件中的注释信息。

Java提供3种注释类型：

- 单行注释：`// 内容`, `//`后面这一行的内容被注释
- 多行注释：`/* 内容 */`; `/*` 和 `*/` 之间的所有内容被注释
- 文档注释：`/**内容 */`; `/**`和`*/`之间内容被注释，此外，还可以专门生成文档信息API

注意：多行注释之间彼此都不能交叉嵌套，以下就是错误的例子。

```
/*
/* 内容 */
*/
```

因为`/`会找举例自己最近的`*/`符号，组成一个注释语句块，上图中单独的`*/`符号就不能被编译器识别了。

1.6.2.2 标准的代码案例（必须这么去做）

后续代码的编写，要严格都要写需求、开发步骤、小结。

需求：在控制台输出内容：你好师姐

```
/*
    开发步骤：
    1: 先定义一个类Hello
    2: 在Hello类中定义一个main方法
    3: 在main方法中使用系统输出打印一句话
    4: 编译和运行程序

    写完代码，写下小结：我在写代码过程的得与失
    1: 如果不写main方法，程序没法运行，因为main方法是程序的入口。    mian
    2: Java严格区分大小写，System不能写出system，其他也要注意。
    3: 巨坑，分号必须使用英文状态下的，不能使用中文(中文的；英文的;)
*/
public class Hello {
    public static void main(String[] args) {
        System.out.println("你好师姐");
    }
}
```

1.6.3. 关键字和保留字（了解）

关键字：在编程语言中事先定义的，有着特殊含义和用途的单词。

保留字：和关键字一样是编程语言事先定义好的，只是说现在暂时没有特殊的功能，但说不定以后某天会突然被赋予功能，因此被保留下来的单词。比如goto和const就是保留字。

abstract	do	implement	private	this
boolean	double	import	protected	throw
break	else	instanceof	public	throws
byte	extends	int	return	transient
case	false	interface	short	true
catch	final	long	static	try
char	fianlly	native	strictfp	void
class	float	new	super	volatile
continue	for	null	switch	while
default	if	package	enum	synchronized
assert				

小结：注意关键字和保留字都是由小写组成，关键字不要去记，我们学一个记一个。

注意：java 无sizeof、goto、const 关键字

有人认为：true, false, null属于字面量（直接量），不属于关键字，无所谓，不要纠结这些学术性。

1.6.4.分隔符和标识符 (记住)

1.6.4.1.语言分隔符 (了解)

分号 (;)：语句的分割，表示一句话结束，好比咱们使用的句号。

花括号 ({})：表示一个代码块，是一个整体，花括号要成对使用。

方括号 ([])：定义数组和访问数组元素时使用。

圆括号 (())：使用很广泛，具体用到细讲。

圆点 (.)：类和对象访问它的成员时使用。

空格 ()：把一整条语句分割成几段，空格的次数不限制，好比一句英文里单词都要分开写一样。

注意：必须都是半角状态下的英文符号，写代码时确保搜狗输入是这个样子的。



小技巧写代码时，全部使用英文的符号，不要使用中文的符号，修改搜狗输入法配置。



1.6.4.2.标识符 (了解)

在写代码的时候为了增强代码的阅读性，我们会自定义很多名字如：类名、方法名、变量名等。

在编程的里，我们把这种为了增强程序阅读性而自定义的名称，称为标识符。

标识符命名规则

- [1] 由字母、数字、下划线、\$组成，但不能以数字开头（注：此处的字母可以是中文、日文等）
- [2] 大小写敏感（类名首字母大写，其他的首字母小写，如果是多个单词，后面的单词首字母都大写）
- [3] 不得使用java中的关键字和保留字
- [4] 不用java中内置的类名作为自己的类名。

总结：（记死）：

1. 标识符必须是字母、数字、下划线、\$构成的。
2. 数字不能开头
3. 不能使用java的关键字和保留字

小结：记住使用一个非关键字和保留字，且有意义的英文单词就很OK了。见名知义

注意：类名必须使用名词，且首字母必须是大写的。

2. Java入门基础下

2.1 数据类型、常量、变量

2.1.1. 常量（掌握）

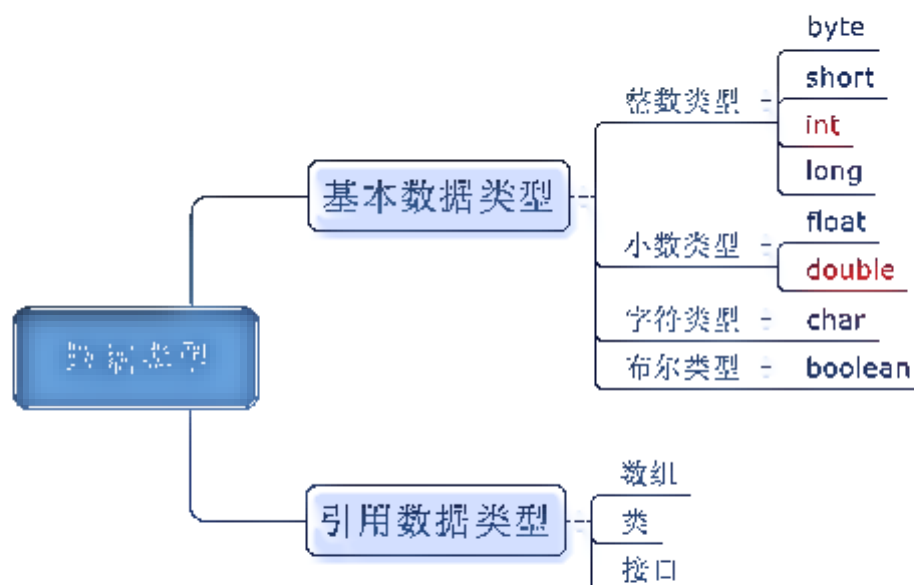
常量，程序运行过程中固定不变化的值。

常量分类：

- 字面量：就表示直接给出的一个值（可以是整数、小数等），也有人称之为直接量。如整数常量1, 2, 3, 小数常量3.14等。。
- 使用final定义的变量（后讲）

2.1.2. 数据类型（重点）

生活中，数据都是有类型这个概念的，比如张三18岁，18这个数字就是整型的，买了2.5斤菜，2.5就是小数类型的，在Java中每一个数据也有数据类型。



常用的整数类型是int和long，byte和short基本不用，常用的小数类型是double，float基本不用。因为double是不精确的，在实际开发中表示精确的小数我们使用BigDecimal类（后面学）。

8种基本数据类型范围和占内存大小（了解）：

NO.	数据类型	占位(字节)	数据范围	默认值
1	byte	1	$[-128, 127]$	0
2	short	2	$[-2^{15}, 2^{15}-1]$	0
3	int	4	$[-2^{31}, 2^{31}-1]$	0
4	long	8	$[-2^{63}, 2^{63}-1]$	0L
5	char	2	$[0, 2^{16}-1]$	'\u0000'
6	float	4	$[-3.4E38(-3.4*10^{38}), 3.4E38(-1.7*10^{38})]$	0.0F
7	double	8	$[-1.7E308(-1.7*10^{308}), 1.7E308(-1.7*10^{308})]$	0.0D
8	boolean	1位	false、true	false

- 整数类型默认是int类型，小数类型默认是double类型
- 表示long类型常量，要加L或者l，建议加L
- 表示float类型常量，要加F或者f，建议加F
- 字符表示Unicode（万国码）编码表中的每一个符号，每个符号使用单引号引起来，其中前128个符号和ASCII表相同，如下图。
- boolean类型适用于逻辑运算，一般用于流程控制。
- 引用类型先不管，先记住String这个类，表示字符串类型就可以了，所谓字符串就是多个字符合在一起，每个字符串使用双引号引起来。

ASCII 字符代码表

ASCII 非打印控制字符		ASCII 打印字符																																															
高位位	低位位	0000				0001				0010				0011				0100				0101				0110				0111																			
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_	`	{		}	~	Back space	
0000	0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_	`	{		}	~		
0001	1																																																
0010	2																																																
0011	3																																																
0100	4																																																
0101	5																																																
0110	6																																																
0111	7																																																
1000	8																																																
1001	9																																																
1010	10																																																
1011	11																																																
1100	12																																																
1101	13																																																
1110	14																																																
1111	15																																																

这张表要记住的几个符号，A在码表的顺序是65，a在码表的顺序是97。

不同数据类型的常量：

- 整数常量，所有整数，如1、2、3、100、200等
- 小数常量，所有小数，如1.2、2.7、3.14等
- 字符常量，0~65535之间的整数或用单引号括起来的符号如，'A'、'a'、'龙'等
- 布尔常量，只有true和false，分别表示对与错
- 字符串常量，使用双引号括起来的内容如：“Will”、“wolfcode”等

需求：定义每一种数据类型的常量

```
public class TypeDemo{
    public static void main(String[] args) {
```



```
//声明范围是一个byte类型常量
System.out.println(20);

//short类型常量
System.out.println(20);

//int类型常量
System.out.println("十进制" + 20);
System.out.println("二进制" + 0b00010100);
System.out.println("八进制" + 024);
System.out.println("十六进制" + 0x14);

//long类型常量,使用L后缀
System.out.println(20L);

//float类型常量,使用F后缀
System.out.println(3.14F);

//double类型常量
System.out.println(3.14);

//char类型常量
System.out.println(65);
System.out.println('A');

//boolean类型常量
System.out.println(true);
System.out.println(false);

//String类型常量
System.out.println("你好");
}
}
```

#####