

# 第一次课 初识 Python

## 一 了解 Python

Python是时下最流行、最火爆的编程语言之一，具体原因如下：

### 1. 简单、易学，适应人群广泛



#### 刚毕业，未来迷茫

大学/高中刚毕业，迷茫群体，看不到未来的方向，期待学一门有前景的技术



#### 跨专业转行

非计算机专业迫切要转行群体，期待学一门靠谱、有前景、易学的技术



#### 无基础 逻辑思维能力强

逻辑思维能力很强，想通过学一门技术来获得工作能力



#### 数学/统计学/物理专业

学过数学、大数据收集或分析、统计学、物理学等，是学这门课的合适人选



#### 传统运维转开发

如果你之前从事的是运维工作遇到瓶颈想转开发岗位，那Python将帮助你成功转型



#### 转型做Web全栈开发

如果你未来职业生涯致力于做Web全栈开发人才，Python会带你成功转型

### 2. 免费、开源

### 3. 应用领域广泛



目前最新版本是 Python3.8.

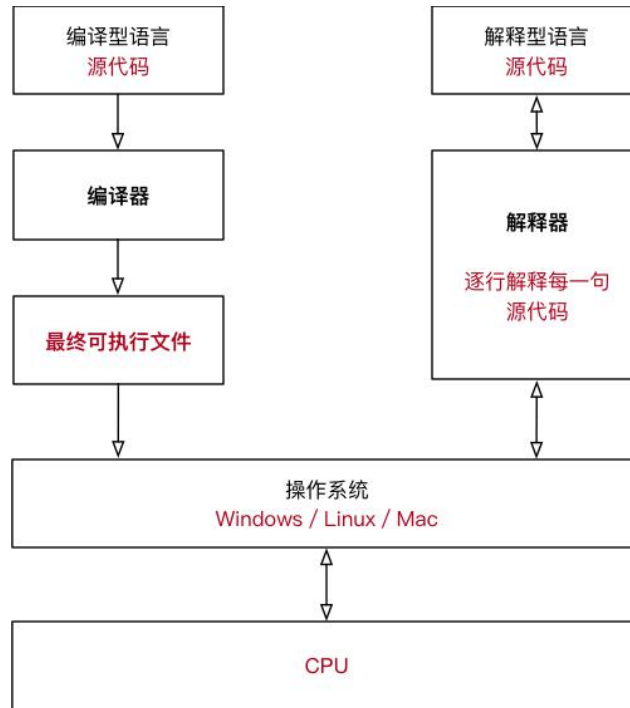
## 二 Python 课程安排

章目	讲课	上机	合计
第 1 章 Python 概述	2 学时	1 学时	3 学时
第 2 章 Python 语法基础	3 学时	1 学时	4 学时
第 3 章 Python 常用语句	4 学时	2 学时	6 学时
第 4 章 字符串	4 学时	2 学时	6 学时
第 5 章 列表元组和字典	3 学时	2 学时	5 学时
第 6 章 函数	5 学时	2 学时	7 学时
第 7 章 高级函数	2 学时	1 学时	3 学时
第 8 章 Python 文件操作	3 学时	2 学时	5 学时
第 9 章 异常	3 学时	1 学时	4 学时
第 10 章 模块	2 学时	1 学时	3 学时
第 11 章 面向对象（上）	4 学时	1 学时	5 学时
第 12 章 面向对象（下）	2 学时	1 学时	3 学时
第 13 章 飞机大战	4 学时	1 学时	5 学时
合计	41 学时	18 学时	59 学时

## 三 Python 解释器

计算机不能直接理解任何除机器语言以外的语言，所以必须要把程序员所写的程序语言翻译成机器语言，计算机才能执行程序。将其他语言翻译成机器语言的工具，被称为**编译器**。

编译器翻译的方式有两种：一个是**编译**，另外一个**解释**。两种方式之间的区别在于翻译时间点的不同。当编译器以**解释方式**运行的时候，也称之为**解释器**。



**编译型语言：**程序在执行之前需要一个专门的编译过程，把程序编译成为机器语言的文件，运行时不需要重新翻译，直接使用编译的结果就行了。程序执行效率高，依赖编译器，跨平台性差些。如 C、C++

**解释型语言：**解释型语言编写的程序不进行预先编译，以文本方式存储程序代码，会将代码一句一句直接运行。在发布程序时，看起来省了道编译工序，但是在运行程序的时候，必须先解释再运行

**Python 的解释器** 如今有多个语言的实现，包括（了解即可）：

- CPython—— 官方版本的 C 语言实现
- Jython—— 可以运行在 Java 平台
- IronPython—— 可以运行在 .NET 和 Mono 平台
- PyPy—— Python 实现的，支持 JIT 即时编译

目前我们使用的解释器是 **Python3.8**

## 四 初识 PyCharm

集成开发环境（IDE，Integrated Development Environment）—— 集成了开发软件需要的所有工具，一般包括以下工具：

图形用户界面  
代码编辑器（支持 代码补全 / 自动缩进）  
编译器 / 解释器  
调试器（断点 / 单步执行）  
.....

## PyCharm 介绍

- PyCharm 是 Python 的一款非常优秀的集成开发环境
- PyCharm 除了具有一般 IDE 所必备功能外，还可以在 Windows、Linux、macOS 下使用
- PyCharm 适合开发大型项目
  - 一个项目通常会包含 很多源文件
  - 每个 源文件 的代码行数是有限的，通常在几百行之内
  - 每个 源文件 各司其职，共同完成复杂的业务功能

下载地址: <https://www.jetbrains.com/pycharm/>

PyCharm 下载有两个版本 一个专业版和社区版，我们上课选择社区版即可（免费）。

收费版：购买费用为 199\$/年，或者 19.90\$ / 月

## 五 安装 Python 环境

见下发文件-

## 六 第一个 Python 项目

Python 源程序就是一个特殊格式的文本文件，可以使用任意文本编辑软件做 Python 的开发，Python 程序的文件扩展名 通常都是 **.py**。

```
print("hello python")  
  
print("hello world")
```

print 是 python 中我们学习的第一个 函数

`print` 函数的作用，可以把 `'''` 内部的内容，输出到屏幕上。

## 七 认识错误（BUG）

### 关于错误

编写的程序不能正常执行，或者执行的结果不是我们期望的俗称 **BUG**，是程序员在开发时非常常见的，初学者常见错误的原因包括：

1. 手误
2. 对已经学习过的知识理解还存在不足
3. 对语言还有需要学习和提升的内容
4. 在学习语言时，不仅要学会语言的语法，而且还要学会如何认识错误和解决错误的方法。

### 第一个演练中的常见错误

- 1>手误，例如使用 `pirnt("Hello world")`

```
NameError: name 'pirnt' is not defined
```

名称错误: `'pirnt'` 名字没有定义

- 2> 将多条 `print` 写在一行

```
SyntaxError: invalid syntax
```

语法错误: 语法无效

每行代码负责完成一个动作

- 3> 缩进错误

```
IndentationError: unexpected indent
```

缩进错误: 不期望出现的缩进

- Python 是一个格式非常严格的程序设计语言
- 目前而言，大家记住每行代码前面都不要增加空格

Python 2.x 默认不支持中文，具体原因等到介绍字符编码时给大家讲解。

开发过程中经常出现的单词列表：

- \* **error** 错误
- \* **name** 名字
- \* **defined** 已经定义
- \* **syntax** 语法
- \* **invalid** 无效
- \* **Indentation** 索引
- \* **unexpected** 意外的，不期望的
- \* **character** 字符
- \* **line** 行
- \* **encoding** 编码
- \* **declared** 声明
- \* **details** 细节，详细信息
- \* **ASCII** 一种字符编码