TOW Macson evenes Import types from watson.framework import events from watson.http.messages import Response from watson.common.imports import and an arrangement of the common and the common areas are as a second of the common areas are a second of the common areas are as a second of the common areas are a second of the common areas are as a second of the common areas are a second of the common areas ar from watson.common.contextmanagers

Python程序语言入门与应用



人生苦短,我用Python

def get_execute_method(self, **ksrps); @abc.abstractmethod And among the contract of



课程计划



部分	章节	授课教师
初识Python	第1章 Python语言概述与开发环境配置	郭长江
	第2章 Python程序实例解析	郭长江
深入Python	第3章 Python基本数据类型	郭长江
	第4章 Python程序控制结构	郭长江
	第5章 Python函数与模块	王海蛟
	第6章 Python组合数据类型	王海蛟
	第7章 Python文件操作	王海蛟
	第8章 Python程序设计方法	王海蛟



课程计划



部分	章节	授课教师
运用Python	第9章 Python科学计算和可视化	王海蛟
	第10章 Python爬虫与web开发	王海蛟
	第11章 Python图形用户界面 (GUI)	王海蛟
	第12章 Python手机应用开发	王海蛟
	第13章 Python办公自动化	郭长江
	第14章 Python机器学习	郭长江
	第15章 Python面向对象编程	郭长江
	第16章 Python其他应用场景	郭长江



课程考核



- 😍 能力和素质 (40%)
 - ❷ 考勤、上课提问、课堂讨论
 - ② 课后作业与实践
- ❖ 期末考试 (60%)
 - → 开卷笔试,以讲授的内容为主要考试内容范围



Python与中国教育





- Python取代VB成为浙江省信息技术教材 编程语言,将纳入高考内容之一
- ② 北京和山东也确定要把Python 编程基础 纳入信息技术课程和高考的内容体系
- 山东省出版的小学信息技术六年级教材加入了 Python 内容
- Python被列入全国计算机等级考试二级范围









Python程序语言入门与应用

第一章 Python语言概述与开发环境配置



郭长江 changjiangguo@xxmu.edu.cn

生命科学技术学院

新乡医学院



该二维码7天内(9月5日前)有效, 重新进入将更新





Why?

Everybody in this country should learn to program a computer... because it teaches you how to think

Steve Jobs, co-founder and CEO of Apple Inc. (1955 - 2011)





学习目标



- ♣ 基本要求
 - ፟ 掌握
 - Python语言开发和运行环境的配置方法
 - ♥ 理解
 - **Python语言的特点**
 - ♣ 编写程序的IPO方法
 - 令 了解
 - Python语言版本更迭过程和新旧版本的主要区别



本课概要



- 第1章 Python语言概述与开发环境
 - 1.1 计算机的概念
 - 1.2 程序设计语言
 - 1.3 Python语言概述
 - 1.4 Python语言开发环境配置
 - 1.5 程序的基本编写方法
 - 1.6 Python语言的版本更迭





1.1 计算机的概念



1.1 计算机的概念



计算机是根据指令操作数据的设备

- 功能性

对数据的操作,表现为数据计算、输入输出处理和结果存储等

- 可编程性

根据一系列指令自动地、可预测地、准确地完成操作者的意图



超级计算机



服务器



八八月小

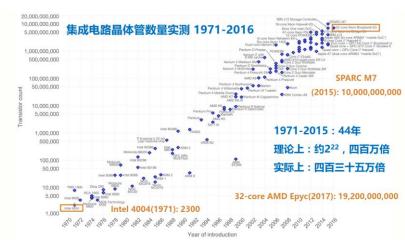


1.1 计算机的概念



• 计算机的发展

- 计算机硬件所依赖的集成电路规模参照摩尔定律发展
- 计算机运行速度接近几何级数快速增长
- 计算机高效支撑的各类运算功能不断丰富发展
- 程序设计语言从机器语言、汇编语言到高级语言发展



摩尔定律



编程语言









🝦 程序设计

计算机可编程性的体现

- 亦称编程,深度应用计算机的主要手段
- 已成为当今社会需求量最大的职业技能之一
- 很多岗位将被计算机程序接管,程序设计将是生存技能

▶ 为什么学习计算机编程?

训练思维(计算思维)、增进认识、带来乐趣、提高效率、利于就业、。。。。





- 聲程序设计语言
 - → 一种用于交互(交流)的人造语言
 - ② 亦称编程语言,程序设计的具体实现方式
 - ② 主要用于人类和计算机之间的交互
- 编程语言分类

机器语言 11010010 00111011 二进制代码

♣ 汇编语言 add 2, 3, result 助记符

相比人类语言,程序设计语言的结构在语法上十分精密,在语义上定义准确。



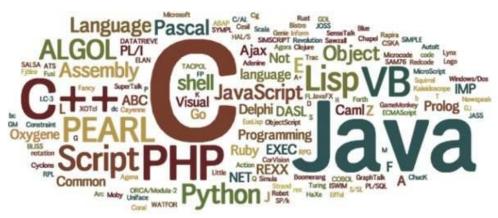


?

编程语言种类

种类很多,但生命力强劲的却不多

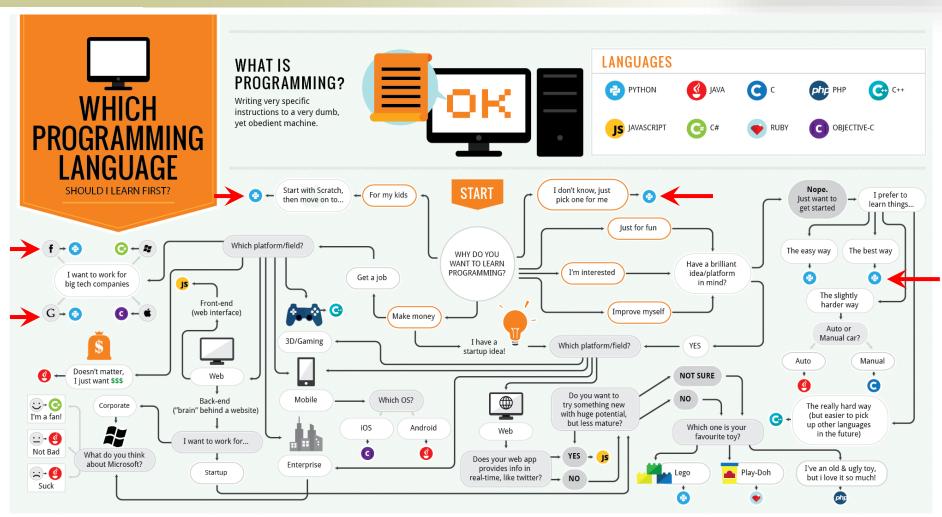
- 编程语言有超过600种,绝大部分都不再被使用
 C, C++, C#, Java, JavaScript, Go, PHP, HTML, R, VB
- C语言 (1972年) 是第一个被广泛使用的编程语言
- Python语言(1990年)是最流行最好用的编程语言



- ➤ 专用编程语言,如HTML
- ➤ 通用编程语言,如Python







编程语言学习选择宝典





• 编程语言的执行方式

编译 or 解释

源代码 采用某种编程语言编写的计算机程序,人类可读

😍 例如: result = 2 + 3

(高级语言代码)

目标代码 计算机可直接执行,人类不可读(专家除外)

例如: 11010010 00111011 (机器语言代码)

源代码 Source code

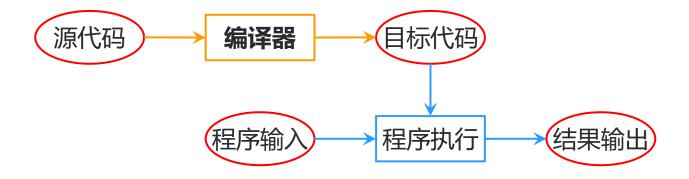


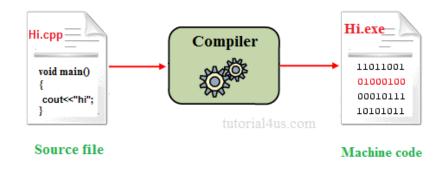
目标代码 Object code





- 编译 Compiler
 - ◆ 将源代码一次性转换成目标代码的过程



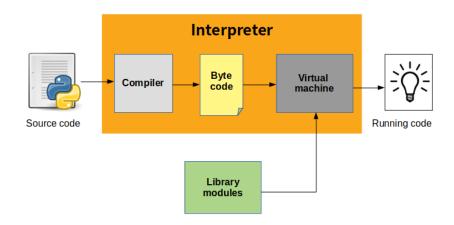






- 解释 Interpreter
 - ❷ 将源代码逐条转换成目标代码同时逐条运行的过程

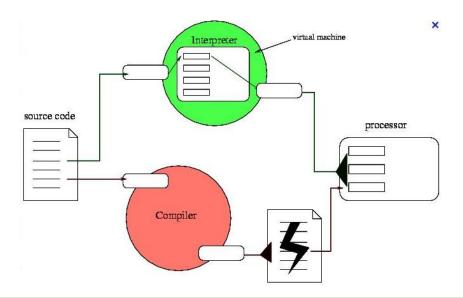








- 🤚 编译 vs 解释
 - 编译 一次性翻译,之后不再需要源代码(类似英文翻译)
 - 目标代码执行速度快,使用灵活(同类操作系统)
 - 解释 每次程序运行时随翻译随执行(类似实时的同声传译)
 - 便于纠错与维护,可移植性好







- 🝦 静态语言
 - ᇦ 编译执行,如C/C++语言、Java语言
- 脚本语言
 - ← 解释执行,如Python、JavaScript、PHP语言

- ♣ Python语言
 - 一种被广泛使用的高级通用脚本编程语言
 - ❷ 具有将解释器与编译器结合的新解释器





Q: 编程很难学吗? A: 掌握方法就很容易!

- 编程语言学习模式
 - ◎ 首先,掌握编程语言的语法,熟悉基本概念和逻辑
 - ❷ 其次,结合计算问题思考程序结构,会使用编程套路
 - ❷ 最后,参照案例多练习多实践,学会举一反三

学习计算机编程的重点在于练习。











- 🕏 Python [`paiθən],译为"蟒蛇"
 - 由编程牛人领导设计并开发的编程语言
 - 有开放、开源精神的编程语言
 - ◎ 应用于火星探测、搜索引擎、引力波分析等众多领域
- Python语言拥有者
 - Python Software Foundation(PSF)
 - PSF是非盈利组织,致力于保护Python语言开放、开源和发展

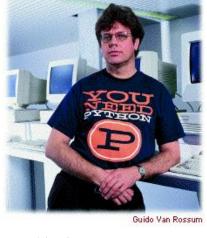






Python语言的诞生

- ◆ 1990年, 吉多·范罗苏姆 (Guido van Rossum) 设计开发
- ② 2000年, Python 2.x
- ② 2008年, Python 3.x



"终身仁慈独裁者"



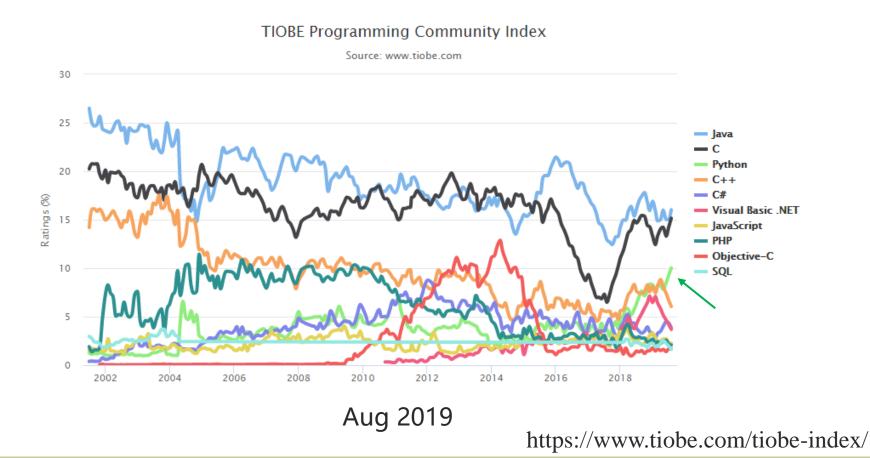
Python语言命名来源







Python语言排行 (TIOBE, 10.02%)

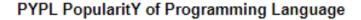


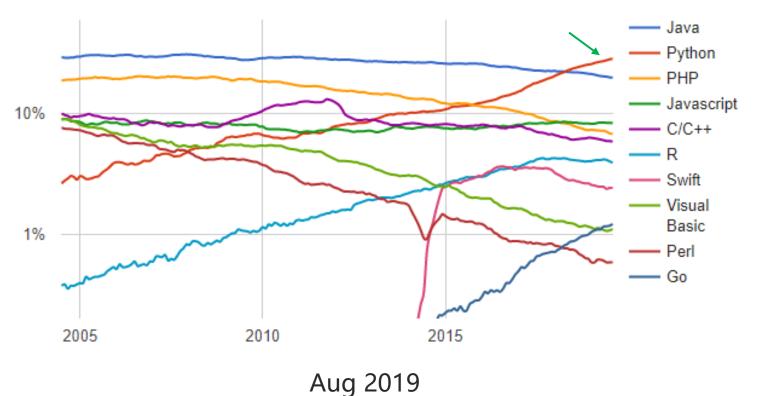






Python语言排行 (PYPL, 28.73%)





https://www.tiobe.com/tiobe-index/



Python是服务端解释器的、开源的、非编译语言,它能用于自身或其他框架

🦆 python" 用来做什么



Python的优点

Python广泛认同为最容易学的编程

大量令人惊讶的库和函数使得制作东

西极为容易。

于Python。







创始于 1991





















C的优点

开发运行在不同平台的软件。

C完全基于变量、宏命令、函数科

架构,他们不是特别多。因此,

现。学会就能使你方便的学会-

从冰箱到闹钟。

学会C学会一切:

切编程语言。

C工作市场

可以嵌入几乎所有现代微型处理器

C很小



C是多用途的、重要的计算机程序语言,支持结构化编程、词法变量范围和递

归。是开发固件或移动应用的理想语言。通常被用于编写系统软件。

欢迎度





C的缺点

C语言没有运行时检查机制。

不支持面向对象编程,这就是

不是最好学的编程语言也不是最

难的。适用于有其他编程基础的

运行时间

面向对象编程

什么创造C++

相当大的学习曲线:

创始于

0-0 1972



数据库

php



Web 开发





PHP

php用来做什么

PHP 是一种服务器端的脚本语言,专为网页开发而设计,但也是一门用途广泛的语言。





















\$89,000









IBM facebook









PHP趣事 php PHP起源于 Personal Home















PHP还能

连接数据



008

创建、打开、读写

关闭服务器文件























市面 上访问



JAVA

Java是一种开发者用来创造计算机应用的程序语言。 你下载程序需要Java运

行环境并确保它安装在你的系统上。Java也有一些web插件允许你在浏览器





Java开发者被大量需求

一种进化的语言

发者主要使用Java

lambda功能。

2015年Dice.com 报告表明Java

开发是需求量最高的软件开发技

谷歌的安卓移动平台是世界第一

的移动平台。编写安卓应用,开

这个平台在继续增加新功能,如











Java的缺点

使用大量内存 lava比C++使用更多内存 更大内存占用。

是学起来最简单的语言,但也 是最难的





Java工作市场





顶级雇主









Java趣事





C编写了。。。

平均薪水

\$102,000



工作数

38,456





顶尖工作位置

New York City

Washington D.C

San Francisco









amazon

C趣事

at&t 造并用于Unix操作系统的再列

很多后期语言是从C中直接或者间接的

C在1969年到1973年间在AT

贝尔实验室由Dennis Ritchie

Java编写了。。



Java的曾用名是Oak, 因为在 James Gosling的窗外有一棵巨 大的橡树 (Oak tree)

Java是安卓操作系统的基础















amazonkindle



avaScipic Ches Pytion















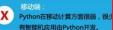
















Python工作市场

Python也许会成为物联网最受欢迎

的语言,如树莓派这样的新平台都基





顶级雇主

Python编写了。。

Instagram

You Tube

Spotify



(intel



amazon Dell

顶尖工作位置

New York City





tiacus

Python是谷歌的

官方语言





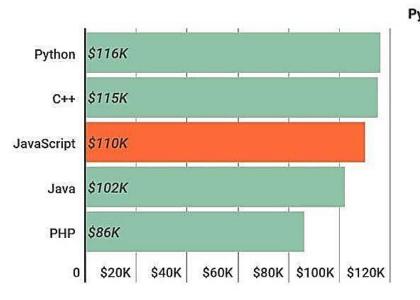


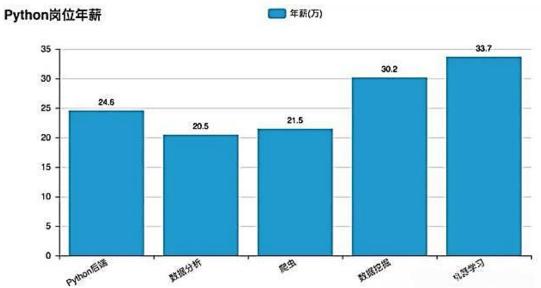












Python 116K 超过 C++、JS, 薪酬排行第一 (来自最大招聘网站Indeed.com数据)

2019年北上广深杭Python就业薪资





- Python语言的特点
 - 语法简洁、跨平台、粘性扩展、开源理念、通用灵活、强制可读、支持中文、模式多样、类库丰富
- Python之禅 (The Zen of Python)
 - 优美胜于丑陋,明了胜于晦涩
 - ◎ 简洁胜于复杂,复杂胜于凌乱
 - ◎ 扁平胜于嵌套,间隔胜于紧凑
 - **(**

Python设计哲学: 优雅、明确、简单。

Python开发哲学:用一种方法,最好是只有一种方法

来做一件事。





😍 Python代码示例

```
# 圆的半径是25
radius = 25
# π值
pi = 3.1415
# 圆的面积公式
area = pi * radius * radius
# 输出面积
print(area)
# 只输出两位小数
print("%.2f" % area)
print("{:.2f}".format(area))
```

```
"Hello, World"
```

```
#include <stdio.h>
int main(int argc, char ** argv)
{
    printf("Hello, World!\n");
}

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

now in Python
print "Hello, World!"
```

简洁性:实现同样功能, Python代码行数仅相当于C语言的1/10~1/5。



更少的程序错误、更快的开发速度、更好的可读性





Python语言重要性

- ፟ 广泛的应用领域
 - 科学计算、数据处理、图形处理、文本处理
 - ◆ 人工智能、Web开发、游戏开发、网络爬虫
 - 办公自动化、软件开发、系统运维、云计算

② 教育语言

- 便于入门,普及编程和人工智能教育
- 😍 培养计算思维,为其他语言学习奠定基础





1.4 Python语言开发环境配置



1.4 Python语言开发环境配置。



- 😍 Python解释器安装
 - ❷ Python语言主网站



- https://www.python.org/downloads/
- Anaconda/Miniconda



- ♣ 开源的Python包管理器
- https://www.anaconda.com/distribution/#downloadsection

Download the latest version for Windows

Download Python 3.7.4

Looking for Python with a different OS? Python for Windows, Linux/UNIX, Mac OS X, Other

Python 3.7 version

Download

64-Bit Graphical Installer (486 MB) 32-Bit Graphical Installer (418 MB)



1.4 Python语言开发环境配置



- 👶 Python解释器安装 (Windows10为例)
 - ♂ 64位安装程序下载链接:

https://www.python.org/ftp/python/3.7.4/python-3.7.4-amd64.exe

32位安装程序下载链接:

https://www.python.org/ftp/python/3.7.4/python-3.7.4.exe





😍 Python解释器安装 (Windows10为例)

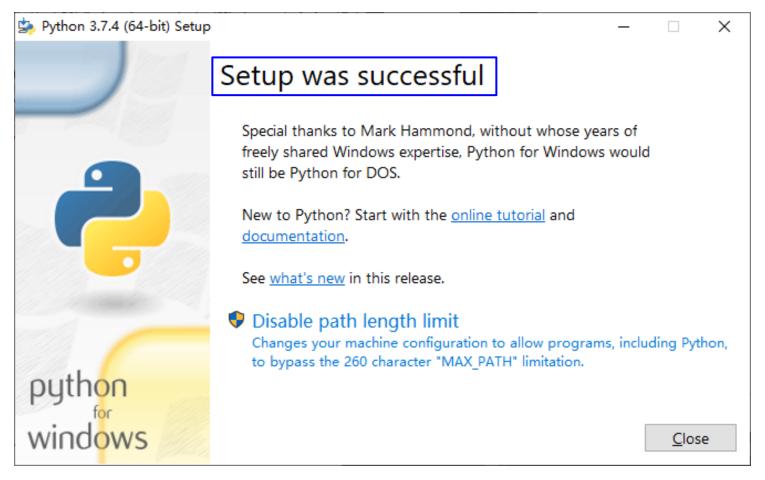


Python解释器安装程序引导过程的启动界面





😍 Python解释器安装 (Windows10为例)



Python解释器安装程序引导过程的成功页面





👶 运行程序

- ፟ 交互式
 - ♣ Python解释器即时响应每条代码,并给出输出结果
 - 一般用于调试少量代码

፟ 文件式

- 也称批量式,将Python程序写在文件中,启动Python解释器 批量执行文件中的代码
- **d** 最常用的Python编程方式





- 🕏 交互式 方法1
 - ∂ 启动Windows命令行工具 (cmd.exe)
 - ② 在 ">>>" 后输入代码: print("Hello World")
 - ② 按Enter键后输出结果: Hello World
 - ✿ 输入exit()或者quit()退出Python运行环境

```
EX C:\Windows\system32\cmd.exe - python

(c) 2019 Microsoft Corporation。保留所有权利。

C:\Users\iSynBio>python

Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 20:34:20) [MSC v.1916 64 bit (AMD64)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>> print("Hello World")

Hello World

>>>
```

运行环境提示符

Windows命令行下的Python交互运行模式





- 🕏 交互式 方法2
 - ♣ 启动Python IDLE
 - ② 在 ">>>" 后输入代码: print("Hello World")
 - ⇔ 按Enter键后输出结果: Hello World
 - ✿ 输入exit()或者quit()退出Python运行环境

Python IDLE下的Python交互运行模式





- 🕈 文件式 方法1
 - 令 将print("Hello World")保存到hello.py
 - ∂ 启动Windows命令行工具 (cmd.exe) , 并切换到程 序文件所在目录
 - 输入: python hello.py
 - ፟ 按Enter键后输出结果: Hello World

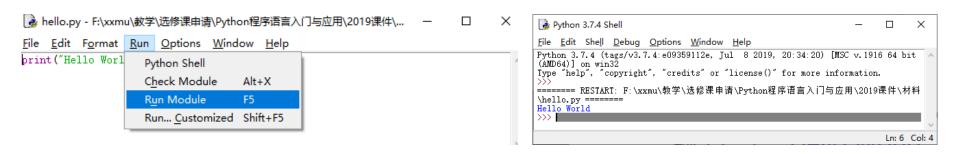


Windows命令行 (CLI) 下运行Python程序文件





- 🕏 文件式 方法2
 - Python IDLE (idle.pyw)
 - ⊕ 通过File菜单打开文件 hello.py
 - 在菜单中选择Run→Run Module选项运行程序文件, 输出结果: Hello World



Python IDLE下运行Python程序文件





Python IDE(集成开发环境)







Studio

Visual Studio Code







Sublime Text

Notepad++ **Atom**









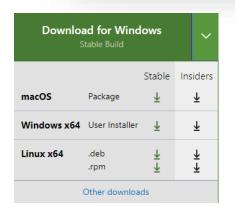
WingIDE

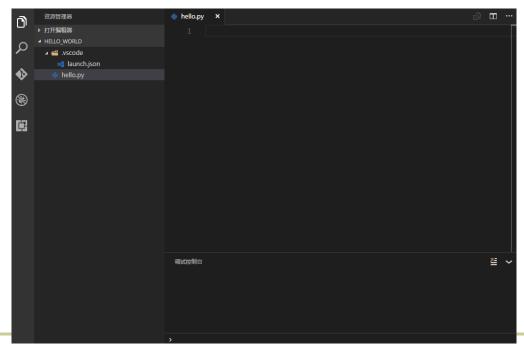




Visual Studio Code

- https://code.visualstudio.com/
- 启动快速,插件丰富,即插即用,集成了现代编辑器所具备的各种特性,包括语法高亮,可定制的热键绑定,括号匹配以及代码片段收集。









e

Python小程序运行实例1

圆面积的计算

IDLE交互环境

文件式内容 1.1CircleArea.py





Python小程序运行实例2

简单人名对话

```
>>> name = input("输入姓名: ")
输入姓名: 郭靖
>>> print("{}同学, 学好Python, 前途无量! ".format(name))
郭靖同学, 学好Python, 前途无量!
>>> print("{}大侠, 学好Python, 大展拳脚! ".format(name[0]))
郭大侠, 学好Python, 大展拳脚!
>>> print("{}哥哥, 学好Python, 人见人爱! ".format(name[1:]))
靖哥哥, 学好Python, 人见人爱!
```

```
name = input("输入姓名: ")
print("{}同学, 学好Python, 前途无量! ".format(name))
print("{}大侠, 学好Python, 大展拳脚! ".format(name[0]))
print("{}哥哥, 学好Python, 人见人爱! ".format(name[1:]))

1.2NameDialogue.py
```





Python小程序运行实例3

斐波那契数列计算

```
>>> a, b = 0, 1
>>> while a < 1000:
    print(a, end=',')
    a, b = b, a + b</pre>
0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987,
```

IDLE交互环境

增加缩进表示语句块的开始减少缩进表示语句块的退出

```
a, b = 0, 1

while a < 1000:

print(a, end=',')

a, b = b, a + b
```

文件式内容

1.3FibonacciSequence.py



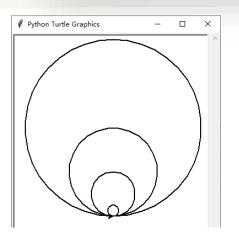




Python小程序运行实例4

同切圆绘制

```
>>> import turtle # 引用turtle库
>>> turtle.pensize(2) # 设置画笔宽度为2像素
>>> turtle.circle(10) # 绘制半径为10像素的圆
>>> turtle.circle(40) # 绘制半径为40像素的圆
>>> turtle.circle(80) # 绘制半径为80像素的圆
>>> turtle.circle(160) # 绘制半径为160像素的圆
```



IDLE交互环境

```
import turtle # 引用turtle库
turtle.pensize(2) # 设置画笔宽度为2像素
turtle.circle(10) # 绘制半径为10像素的圆
turtle.circle(40) # 绘制半径为40像素的圆
turtle.circle(80) # 绘制半径为80像素的圆
turtle.circle(160) # 绘制半径为160像素的圆
```

文件式内容

1.4CircleDraw.py







Python小程序运行实例5

日期和时间输出

IDLE交互环境

```
from datatime importdatetime# 引用datetime库now = datatime.now()# 获得当前的日期和时间信息print(now)# 输出其中的日期部分now.strftime("%x")# 输出其中的时间部分
```

文件式内容

1.5Datetime.py









?

IPO

- 🖰 I: Input 输入,程序的输入
- 😌 P: Process 处理,程序的主要逻辑
- ② O: Output 输出,程序的输出







Input 输入

- ₹ 程序的输入
 - 文件输入、网络输入、控制台输入、交互界面输入、 内部参数输入等
- 输入是一个程序的开始









python argparse_test.py -t 1
python argparse_test.py -t=1
python argparse_test.py --thread 1
python argparse_test.py --thread=1

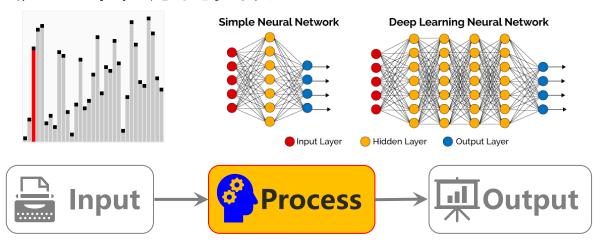






Process 处理

- 处理
 - 程序对输入数据进行计算产生输出结果的过程
- 处理方法
 - 统称为算法,它是程序最重要的部分
 - 算法是一个程序的灵魂



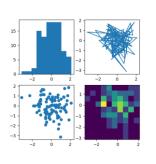




Output 输出

- ♣ 程序的输出
 - 空 控制台输出、图形输出、文件输出、网络输出、操作系统内部变量输出等
- 输出是程序展示运算结果的方式















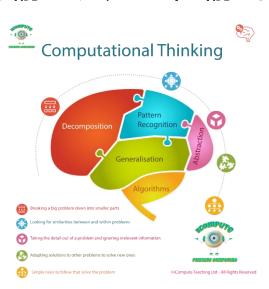


问题的计算部分

一个待解决问题中,可以用程序辅助完成的部分

- 计算机只能解决计算问题,即问题的计算部分
- 一个问题可能有多种角度理解,产生不同的计算部分
- 问题的计算部分一般都有输入、处理和输出过程









编程解决问题的步骤

分析问题:分析问题的计算部分,想清楚

炒 划分边界:划分问题的功能边界,规划IPO

② 设计算法:设计问题的求解算法,关注算法

编写程序:编写问题的计算程序,编程序

⇨ 调试测试:调试程序使正确运行,运行调试

分 升级维护:适应问题的升级维护,更新完善

3个精简步骤:确定IPO→编写程序→调试程序





1.6 Python语言的版本更迭

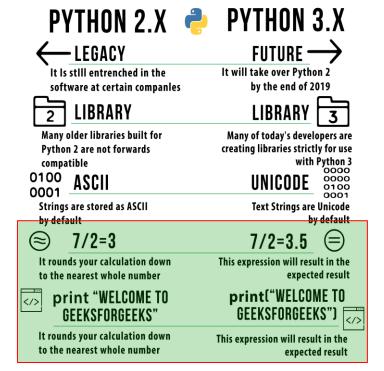


1.6 Python语言的版本更迭



- Python 2.x与Python 3.x
 - ② 2010年, Python 2.x最后一个版本2.7
 - ② 2008年, Python 3.x第一个主版本3.0
- 版本差异

Python 2.7 将于2020年1月1 日终止支持。



Python 3.x 不完全向下兼容 Python 2.x程序!



1.6 Python语言的版本更选



- ♣ 版本选择建议
 - ♂ 若无特殊情况,请学习Python 3.x版本
 - ♦ 特殊需要可以使用Anaconda虚拟环境管理功能搭建 Python 2.x开发环境





本章要点



- 程序设计语言的执行方式包括编译和解释两种。
- 伊 Python语言是─个语法简洁、跨平台、可扩展的开源通用脚本语言。
- ② IDLE是轻量级Python语言开发环境,支持交互式和批量式两种编程方式。
- 程序的基本编写方法为IPO方法。
- ❷ Python 3.x版本是Python语言的现在和未来。



程序练习题



1.1 字符串拼接

- 如: [某人]想去[某地]看看。
- 接受用户输入的2个字符串,将它们组合后输出。
- 1.2 整数序列求和

用户输入一个正整数N, 计算从1到N相加之和。

1.3 九九乘法表输出

工整打印输出常用的九九乘法表,格式不限。

- 1.4 计算1!+2!+3!+...+10!
- 1.5 五角星的绘制 绘制一个红色的五角星。



作业的代码文件打包(.zip)发送到邮箱python_xxmu@163.com

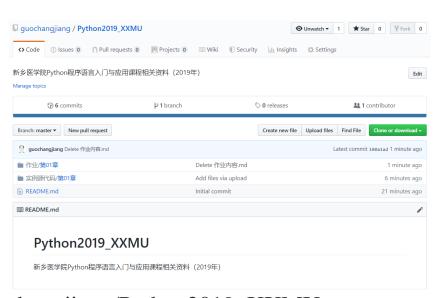


课程材料地址

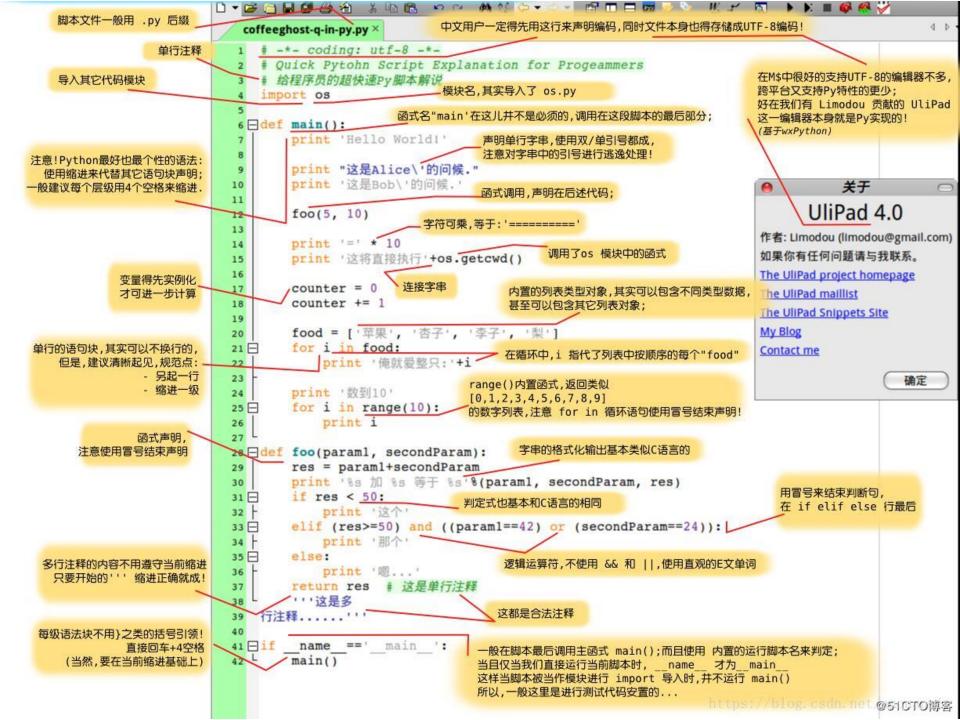


- GitHub
 - https://github.com
 - 一个面向开源及私有软件项目的托管平台
 - 可以十分轻易地找到海量的开源代码





https://github.com/guochangjiang/Python2019_XXMU 包含实例代码、课程作业、作业代码等课程资料





下周课程



- * 第2章 Python程序实例解析
 - Python语言基本语法
 - 😌 实例1: 温度转换
 - 😊 实例2: Python蟒蛇绘制

编程辣么好,还等什么?开始学习吧!





Programing is an Art