

## 北京 python 培训超全大纲文档

### 第一篇 python 语言基础知识

#### 第 1 章 python 概述

##### 1.1python 简介

##### 1.2python 的特征

##### 1.3 第一个 python 程序

##### 1.4 搭建开发环境

##### 1.4.1python 的下载和安装

##### 1.4.2 交互式命令行的使用方法

##### 1.5python 的开发工具

##### 1.5.1pythonwin 的使用方法

##### 1.5.2eclipse ide 集成开发环境

##### 1.5.3editplus 编辑器环境的配置

##### 1.6 不同平台下的 python

##### 1.7 小结

#### 第 2 章 python 的基本语法

##### 2.1python 的文件类型

##### 2.2python 的编码规则

##### 2.2.1 命名规则

##### 2.2.2 代码缩进与冒号

- 2.2.3 模块导入的规范
- 2.2.4 使用空行分隔代码
- 2.2.5 正确的注释
- 2.2.6 语句的分隔
- 2.3 变量和常量
  - 2.3.1 变量的命名
  - 2.3.2 变量的赋值
  - 2.3.3 局部变量
  - 2.3.4 全局变量
  - 2.3.5 常量
- 2.4 数据类型
  - 2.4.1 数字类型
  - 2.4.2 字符串类型
- 2.5 运算符与表达式
  - 2.5.1 算术运算符和算术表达式
  - 2.5.2 关系运算符和关系表达式
  - 2.5.3 逻辑运算符和逻辑表达式
  - 2.5.4 运算符的优先级
- 2.6 小结



## 第 3 章 python 的控制语句

### 3.1 结构化程序设计

#### 3.2 条件语句

##### 3.2.1if 语句

##### 3.2.2if...elif...else 语句

##### 3.2.3if 语句的嵌套

##### 3.2.4 实现 switch 语句的功能

#### 3.3 循环语句

##### 3.3.1while 循环

##### 3.3.2for 循环

##### 3.3.3break 和 continue 语句

### 3.4 结构化程序示例

#### 3.5 小结

## 第 4 章内置数据结构

### 4.1 元组

#### 4.1.1 元组的创建

#### 4.1.2 元组的访问

#### 4.1.3 元组的遍历

### 4.2 列表

#### 4.2.1 列表的创建

#### 4.2.2 列表的使用方法

#### 4.2.3 列表的查找、排序与反转

#### 4.2.4 用列表实现堆栈和队列

### 4.3 字典

#### 4.3.1 字典的创建

#### 4.3.2 字典的访问

#### 4.3.3 字典的方法

#### 4.3.4 字典的排序与复制

#### 4.3.5 全局字典——sys.modules 模块

### 4.4 序列

### 4.5 小结



## 第 5 章 模块与函数

### 5.1 python 程序的结构

### 5.2 模块

#### 5.2.1 模块的创建

#### 5.2.2 模块的导入

#### 5.2.3 模块的属性

#### 5.2.4 模块的内置函数

### 2.5 自定义包

### 5.3 函数

#### 5.3.1 函数的定义

#### 5.3.2 函数的参数

#### 5.3.3 函数的返回值

#### 5.3.4 函数的嵌套

#### 5.3.5 递归函数

5.3.6 lambda 函数

5.3.7 generator 函数

5.4 小结

## 第 6 章字符串与正则表达式

6.1 字符串的操作

6.1.1 字符串的格式化

6.1.2 字符串的转义符

6.1.3 字符串的合并

6.1.4 字符串的截取

6.1.5 字符串的比较

6.1.6 字符串的反转

6.1.7 字符串的查找和替换

6.1.8 字符串与日期的转换

6.2 正则表达式

6.2.1 正则表达式简介

6.2.2 使用 sys.re 模块处理正则表达式

6.3 小结

## 第 7 章文件的处理

7.1 文件的基本操作

7.1.1 文件的打开或创建

- 7.1.2 文件的读取
- 7.1.3 文件的写入
- 7.1.4 文件的删除
- 7.1.5 文件的复制
- 7.1.6 文件的重命名
- 7.1.7 文件内容的查找和替换
- 7.1.8 文件的比较
- 7.1.9 配置文件的访问
- 7.2 目录的基本操作
  - 7.2.1 目录的创建与删除
  - 7.2.2 目录的遍历
- 7.3 文件和流
  - 7.3.1 python 的流对象
  - 7.3.2 模拟 java 的输入、输出流
- 7.4 文件处理示例——文件属性浏览程序
- 7.5 小结

## 第 8 章 面向对象编程

- 8.1 面向对象编程概述
- 8.2 类和对象
  - 8.2.1 类和对象的区别
  - 8.2.2 类的定义

- 8.2.3 对象的创建
- 8.2.4 python 对象的体系结构
- 8.3 属性和方法
  - 8.3.1 类的属性
  - 8.3.2 类的方法
  - 8.3.3 内部类的使用
  - 8.3.4 \_\_init\_\_ 方法
  - 8.3.5 \_\_del\_\_ 方法
  - 8.3.6 垃圾回收机制
  - 8.3.7 类的内置方法
  - 8.3.8 方法的动态特性
- 8.4 继承
  - 8.4.1 使用继承
  - 8.4.2 抽象类的模拟
  - 8.4.3 多态性
  - 8.4.4 多重继承
  - 8.4.5 更好的继承方式
- 8.5 运算符的重载
- 8.6 python 与设计模式
  - 8.6.1 设计模式 简介
  - 8.6.2 设计模式 示例——python 实现工厂方法
- 8.7 小结

## 第 9 章异常处理与程序调试

### 9.1 异常的处理

#### 9.1.1 python 中的异常

#### 9.1.2 try...except 的使用方法

#### 9.1.3 try...finally 的使用方法

#### 9.1.4 使用 raise 抛出异常

#### 9.1.5 自定义异常

#### 9.1.6 assert 语句的使用方法

#### 9.1.7 异常信息

### 9.2 使用 pythonwin 调试程序

### 9.3 使用 eclipse for python 调试程序

#### 9.3.1 新建工程

#### 9.3.2 配置调试

#### 9.3.3 设置断点

### 9.4 小结



## 第 10 章 python 的其他数据库 编程

### 10.1python 环境下的其他数据库 编程

#### 10.1.1 通过 odbc 访问其他数据库

#### 10.1.2 使用 dao 对象访问其他数据库

#### 10.1.3 使用 activex data object 访问其他数据库

#### 10.1.4python 连接其他数据库 的专用模块

### 10.2python 的持久化模块

#### 10.2.1 用 dbhash 模块读写数据

#### 10.2.2shelve 模块读写数据

### 10.3 嵌入式其他数据库 sqlite

#### 10.3.1sqlite 的命令行工具

#### 10.3.2 使用 sqlite3 模块访问 sqlite 其他数据库

### 10.4 小结

## 第二篇 python 的 gui 程序设计

### 第 11 章 wxpython 的开发环境

#### 11.1python 所支持的 gui 开发库

##### 11.1.1tkinter

##### 11.1.2pygtk

##### 11.1.3pyqt

#### 11.2wxpython 介绍

##### 11.2.1 下载和安装 wxpython

11.2.2 第一个 wxpython 程序

11.2.3 wxpython 的封装：pythoncard

11.3 wxpython 的开发工具

11.3.1 使用 wxglade 开发 wxpython 程序

11.3.2 使用 wxformbuilder 开发 wxpython 程序

11.3.3 使用 boa constructor 开发 wxpython 程序

11.4 小结

第 12 章 wxpython 框架初步

12.1 wxpython 的程序结构

12.1.1 wxpython 应用程序的组成

12.1.2 wxpython 窗口的组成

12.1.3 wxpython 应用程序的运行过程

12.2 wxpython 中的事件驱动编程

12.2.1 事件驱动的编程模型

12.2.2 事件的触发

12.2.3 事件的绑定和处理

12.3 wxpython 的常用组件

12.3.1 对话框

12.3.2 工具栏

12.3.3 状态栏

12.4 小结

## 第 13 章 wxpython 库中的基本组件

### 13.1 静态文本框

### 13.2 文本框

#### 13.2.1 单行文本框

#### 13.2.2 多行文本框

### 13.3 按钮控件

#### 13.3.1 普通按钮

#### 13.3.2 位图按钮

### 13.4 单选框

#### 13.4.1 单选框的选择

#### 13.4.2 单选框的分组

### 13.5 多选框

#### 13.5.1 多选框

#### 13.5.2 多选框事件

### 13.6 列表控件

#### 13.6.1 列表框

#### 13.6.2 下拉列表

#### 13.6.3 可编辑的下拉列表

### 13.7 使用 sizers 布局组件

#### 13.7.1 sizers 布局管理器

#### 13.7.2 grid sizer 布局

13.7.3flex grid sizer 布局

13.7.4grid bag sizer 布局

13.7.5box sizer 布局

13.8 小结

第 14 章 wxpython 库中的菜单、窗口与对话框控件

14.1 菜单

14.1.1 创建菜单

14.1.2 菜单的事件

14.1.3 多级菜单

14.1.4 菜单的快捷键

14.1.5 位图菜单

14.1.6 上下文菜单

14.2 窗口

14.2.1frame 窗口

14.2.2 带工具栏和状态栏的 frame 窗口

14.2.3mdi 窗口

14.2.4miniframe 窗口

14.3 对话框

14.3.1 对话框的创建

14.3.2 提示对话框

14.3.3 文本输入对话框

14.3.4 特殊对话框

14.3.5 窗口与对话框的交互

14.4 小结

第 15 章 wxpython 库中的高级控件

15.1 表格控件

15.1.1 表格控件的创建

15.1.2 通过 pygridtable 创建表格

15.1.3 表格控件的单元格

15.1.4 处理单元格的事件

15.2 高级列表控件

15.2.1 列表控件的创建

15.2.2 列表控件的排序

15.2.3 带位图的列表控件

15.3 树型控件

15.3.1 树型控件的创建

15.3.2 树型控件的事件

15.3.3 树表控件

15.4 定时器控件

15.5 小结

第 16 章 wxpython 库中的高级功能

## 16.1 wxpython 下的 html 显示

### 16.1.1 html 的显示

### 16.1.2 基于 htmlwindow 的简单浏览器

## 16.2 wxpython 中的 xrc

### 16.2.1 使用 xrc

### 16.2.2 xrc 文件

## 16.3 剪贴板的操作

### 16.3.1 wxpython 中的剪贴板对象

### 16.3.2 处理剪贴板对象

### 16.3.3 剪贴板操作实例

## 16.4 小结

## 第三篇 python 的 web 开发

## 第 17 章 python 的 html 应用

### 17.1 html 介绍

#### 17.1.1 html 的历史

#### 17.1.2 sgml、html 和 xhtml 三者的关系

#### 17.1.3 html 的标签

#### 17.1.4 html 的框架组成

### 17.2 url 的处理

#### 17.2.1 统一资源定位符 url

#### 17.2.2 url 的解析

- 17.2.3url 的拼合
- 17.2.4url 的分解
- 17.2.5url 的编码与解码
- 17.2.6 中文的编码与解码
- 17.2.7 查询参数的编码
- 17.3cgi 的使用
  - 17.3.1cgi 介绍
  - 17.3.2 获取 cgi 环境信息
  - 17.3.3 解析用户的输入
- 17.4 获取 html 资源
  - 17.4.1 使用 urlopen 和 urlretrieve 获取 http 资源
  - 17.4.2 分析返回资源的相关信息
  - 17.4.3 自定义获取资源方式
  - 17.4.4 使用 urllib2 模块获取资源
  - 17.4.5 使用 httplib 模块获取资源
- 17.5html 文档的解析
  - 17.5.1 使用 htmlparser 模块
  - 17.5.2sgmlib 的 html 文档处理
  - 17.5.3html 文档元素的处理
  - 17.5.4 使用 htmllib 处理 html 文档
- 17.6 小结



千锋Python培训班 时代风口 耀世起航

# 打造颠覆性Python全栈工程师

报名即送价值500元树莓派设备

- ✓ 源于清华来自微软首席培训师
- ✓ 23周超长课程体系融会贯通
- ✓ 易入门前景广拿高薪不再是梦

 千锋教育™  
www.mobiletrain.org

## 第 18 章 python 和 xml

### 18.1xml 介绍

#### 18.1.1xml 的演进历史

#### 18.1.2xml 的优点和限制

#### 18.1.3xml 技术的 python 支持

### 18.2xml 文档概览和验证

#### 18.2.1xml 文档的基本概念

#### 18.2.2xml 文档的结构良好性验证

#### 18.2.3xml 文档的有效性验证

### 18.3xml 文档的结构

#### 18.3.1xml 的元素和标签

#### 18.3.2 元素的属性

#### 18.3.3xml 的文字

#### 18.3.4 字符实体

#### 18.3.5cdata 段

18.3.6 注释

18.3.7 处理指令

18.3.8xml 定义

18.4 使用 sax 处理 xml 文档

18.4.1sax 介绍

18.4.2sax 处理的组成部分

18.5 使用 dom 处理 xml 文档

18.5.1dom 介绍

18.5.2xml.dom 模块中的接口操作

18.5.3 对 xml 文档的操作

18.6 小结

第 19 章 python 的 web 开发——django 框架的应用

19.1 常见的 web 开发框架

19.1.1zope

19.1.2turbogears

19.1.3django

19.1.4 其他的 web 开发框架

19.1.5 各种开发框架的选择

19.2mvc 模式

19.2.1mvc 模式介绍

19.2.2mvc 模式的优点和缺点

19.2.3django 框架中的 mvc

19.3django 开发环境的搭建

19.3.1django 框架的安装

19.3.2 其他数据库 的配置

19.4django 框架的应用

19.4.1web 应用的创建

19.4.2django 中的开发服务器

19.4.3 创建其他数据库

19.4.4 生成 django 应用

19.4.5 创建数据模型

19.4.6url 设计

19.4.7 创建视图

19.4.8 模板系统

19.4.9 发布 django 项目

19.5django 框架的高级应用

19.5.1 管理界面

19.5.2 生成其他数据库 数据

19.5.3session 功能

19.5.4 国际化

19.6 小结



## 第 20 章敏捷方法学在 python 中的应用——测试驱动开发

### 20.1 测试驱动开发

#### 20.1.1 测试驱动开发 模式 (tdd)

#### 20.1.2tdd 的优势

#### 20.1.3tdd 的使用步骤

### 20.2unittest 测试框架

#### 20.2.1unittest 模块介绍

#### 20.2.2 构建测试用例

#### 20.2.3 构建测试固件

#### 20.2.4 组织多个测试用例

#### 20.2.5 构建测试套件

#### 20.2.6 重构代码

#### 20.2.7 执行测试

### 20.3 使用 doctest 模块进行测试

#### 20.3.1doctest 模块介绍

20.3.2 构建可执行文档

20.3.3 执行 doctest 测试

20.4 小结

第四篇 python 的其他应用

第 21 章 python 中的进程和线程

21.1 进程和线程

21.1.1 进程和线程的概念

21.1.2 python 中对于进程和线程处理的支持

21.2 python 下的进程编程

21.2.1 进程的运行环境

21.2.2 创建进程

21.2.3 终止进程

21.3 使用 subprocess 模块管理进程

21.3.1 使用 popen 类管理进程

21.3.2 调用外部系统命令

21.3.3 替代其他进程创建函数

21.4 进程间的信号机制

21.4.1 信号的处理

21.4.2 信号的使用规则

21.5 多线程概述

21.5.1 什么是多线程

21.5.2 线程的状态

21.5.3 python 中的线程支持

21.6 生成和终止线程

21.6.1 使用 thread 模块

21.6.2 使用 threading.thread 类

21.7 管理线程

21.7.1 线程状态转移

21.7.2 主线程对子线程的控制

21.7.3 线程中的局部变量

· x viii ·

21.8 线程之间的同步

21.8.1 临界资源和临界区

21.8.2 锁机制

21.8.3 条件变量

21.8.4 信号量

21.8.5 同步队列

21.8.6 线程同步小结

21.9 小结

第 22 章基于 python 的系统管理

22.1 增强的交互式环境 ipython

22.1.1 ipython 介绍

- 22.1.2ipython 的安装
- 22.1.3ipython 的启动
- 22.1.4ipython 的环境配置
- 22.1.5ipython 的退出
- 22.2 和 ipython 的简单交互
  - 22.2.1ipython 中的输入与输出
  - 22.2.2 输出提示符的区别
  - 22.2.3 输出提示符不同的原因
  - 22.2.4ipython 中的 in 和 out
- 22.3ipython 中的 magic 函数
  - 22.3.1magic 函数的使用和构造
  - 22.3.2 目录管理
  - 22.3.3 对象信息的收集
- 22.4ipython 适合于系统管理的特点
  - 22.4.1tab 补全
  - 22.4.2 历史记录功能
  - 22.4.3 执行外部的系统命令和运行脚本文件
  - 22.4.4 对象的查看和自省
  - 22.4.5 直接编辑代码
  - 22.4.6 设置别名和宏
- 22.5 使用 python 进行文件管理
  - 22.5.1 文件的比较

22.5.2 文件的归档

22.5.3 文件的压缩

22.6 使用 python 定时执行任务

22.6.1 使用休眠功能

22.6.2 使用 sched 来定时执行任务

22.7 小结

· x viii ·

23 章 python 和网络编程

23.1 网络模型介绍

23.1.1osi 简介

23.1.2tcp/ip 简介

23.2socket 应用

23.2.1socket 基础知识

23.2.2socket 的工作方式

23.3 服务器端和客户端通信

23.3.1 服务器端的构建

23.3.2 客户端的构建

23.4 异步通信方式

23.4.1 使用 fork 方式

23.4.2 使用线程方式

23.4.3 异步 io 方式

23.4.4 使用 asyncore 模块

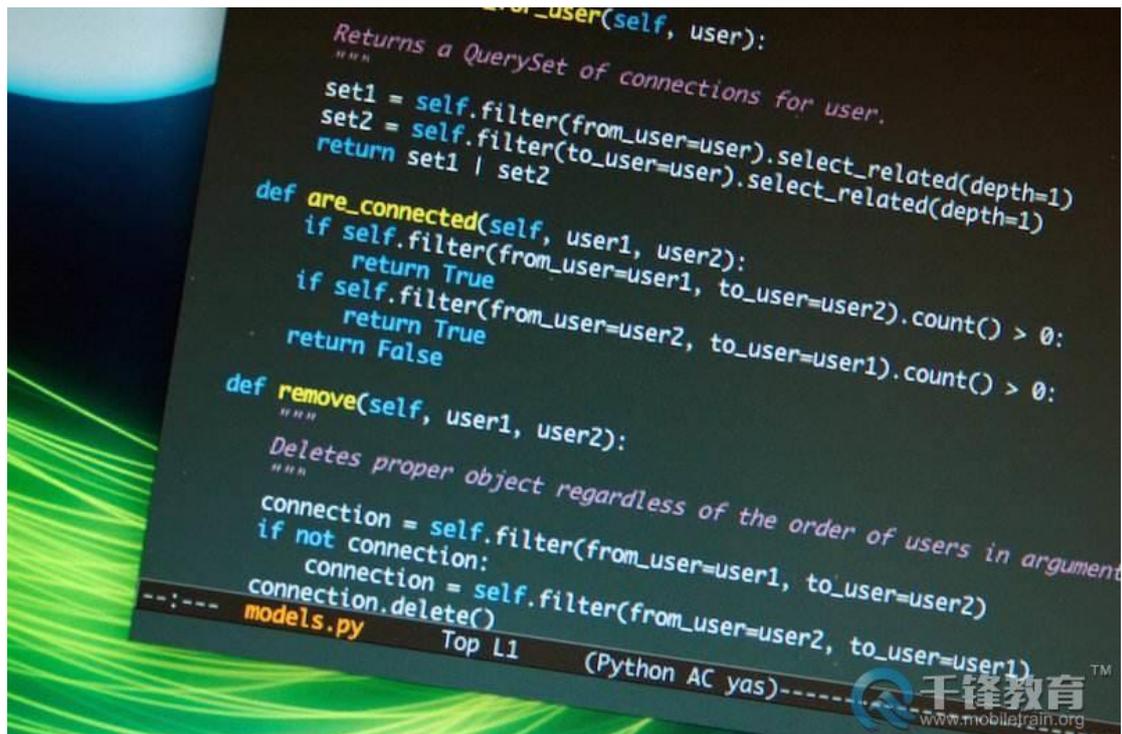
## 23.5twisted 网络框架

### 23.5.1 认识 twisted 框架

### 23.5.2twisted 框架下服务器端的实现

### 23.5.3twisted 框架下服务器端的其他处理

## 23.6 小结



## 24 章常见的 python 网络应用

### 24.1 使用 ftp 传输文件

#### 24.1.1ftp 的工作原理和 python 库的支持

#### 24.1.2ftp 的登录和退出

#### 24.1.3ftp 的数据传输

### 24.2 使用 pop3 获取邮件

#### 24.2.1pop3 协议介绍

24.2.2poplib 模块的使用方法

24.3 使用 smtp 发送邮件

24.3.1smtp 协议介绍

24.3.2smtp 模块的使用

24.4 使用 telnet 协议远程登录

24.4.1telnet 协议介绍和 python 库的支持

24.4.2telnetlib 模块的使用方法

24.5 使用 snmp 管理网络

24.5.1snmp 协议的组成

24.5.2pysnmp 框架介绍及其使用方法

24.6 网络分析

24.6.1 网络分析概述

24.6.2 使用 scapy 在网络中抓包分析

24.7 小结

第 25 章图像处理 and 游戏开发

25.1 图像处理的基本概念

25.1.1python 下的图像处理包

25.1.2pil 支持的图像文件格式

25.1.3 图像处理中的其他概念

25.2 图像的基本处理

25.2.1 图像的读写操作

25.2.2 获取图像信息

25.2.3 图像文件格式的转换

25.2.4 图像的裁剪和合成

25.2.5 图像的变换

· x ix ·

25.3 图像处理的高级应用

25.3.1 图像的通道操作

25.3.2 图像的增强处理

25.3.3 pil 中的内置滤镜

25.4 使用 pygame 进行游戏开发

25.4.1 python 游戏开发 包: pygame

25.4.2 创建基本的 pygame 窗口

25.4.3 事件系统

25.5 小结

26 章 python 语言的扩展与嵌入

26.1 python 语言的扩展

26.1.1 python 扩展简介

26.1.2 一个 c 扩展的例子

26.1.3 模块方法表和初始化函数

26.1.4 编译和测试

26.2 python 语言的嵌入

26.2.1python 嵌入简介

26.2.2 一个 python 嵌入的例子

26.2.3 更好的嵌入

26.3 小结

第 27 章 windows 下的 python

27.1 组件对象模型

· x x ·

27.1.1 组件对象模型介绍

27.1.2com 结构

27.1.3com 对象的交互

27.2python 对 com 技术的支持

27.2.1python 中的 windows 扩展: pywin32

27.2.2 客户端 com 组件

27.2.3 实现 com 组件

27.3windows 下的常见 python 应用

27.3.1 对 word 的自动访问

27.3.2 对 excel 的自动访问

27.3.3 对 powerpoint 的自动访问

27.3.4 对 outlook 的自动访问

27.4 小结

以上为千锋 Python 学院课程大纲。