Tedu.cn
达内教育

Python开发入门

NSD PYTHON1 DAY01

内容

上午	09:00 ~ 09:30	Python概述
	09:30 ~ 10:20	环境准备
	10:30 ~ 11:20	
	11:30 ~ 12:00	Python起步
下午	14:00 ~ 14:50	
	15:00 ~ 15:50	数据类型概述
	16:10 ~ 17:00	
	17:10 ~ 18:00	总结和答疑

python概述

Tedu.cn
达内教育

Python简介

python起源

- 贵铎·范·罗萨姆（Guido van Rossum）于1989年底始创了python
- 1991年初，python发布了第一个公开发行版
- 为了更好的完成荷兰的CWI（国家数学和计算机科学研究院）的一个研究项目而创建

知识讲解

Python版本

- Python2.x
  - 目前所有系统默认安装的版本
- Python3.x
  - 2009年2月13日发布
  - 在语法和功能上有较大调整
  - Python的发展趋势

知识讲解

Python的特点

- 高级：有高级的数据结构，缩短开发时间与代码量
- 面向对象：为数据和逻辑相分离的结构化和过程化编程添加了新的活力
- 可升级：提供了基本的开发模块，可以在它上面开发软件，实现代码的重用
- 可扩展：通过将其分离为多个文件或模块加以组织管理

知识讲解

Python的特点（续1）

知识讲解

- 可移植性：python是用C写的，又由于C的可移植性，使得python可以运行在任何带有ANSI C编译器的平台上
- 易学：python关键字少、结构简单、语法清晰
- 易读：没有其他语言通常用来访问变量、定义代码块和进行模式匹配的命令式符号
- 内存管理器：内存管理是由python解释器负责的

环境准备

Tedu.cn
达内教育

安装与配置

获取python3源码

- 官方站点
  - http://www.python.org
- 选择正确的系统
- 选择正确的版本

知识讲解

安装python3

- 安装依赖包

```
# yum install -y gcc gcc-c++ zlib-devel openssl-devel readline-devel \
 libffi-devel sqlite-devel tcl-devel tk-devel
```

- 安装python3

```
# tar xzf Python-3.6.7.tar.gz
# cd Python-3.6.7
# ./configure --prefix=/usr/local
# make && make install
```

知识讲解

设置pycharm

知识讲解

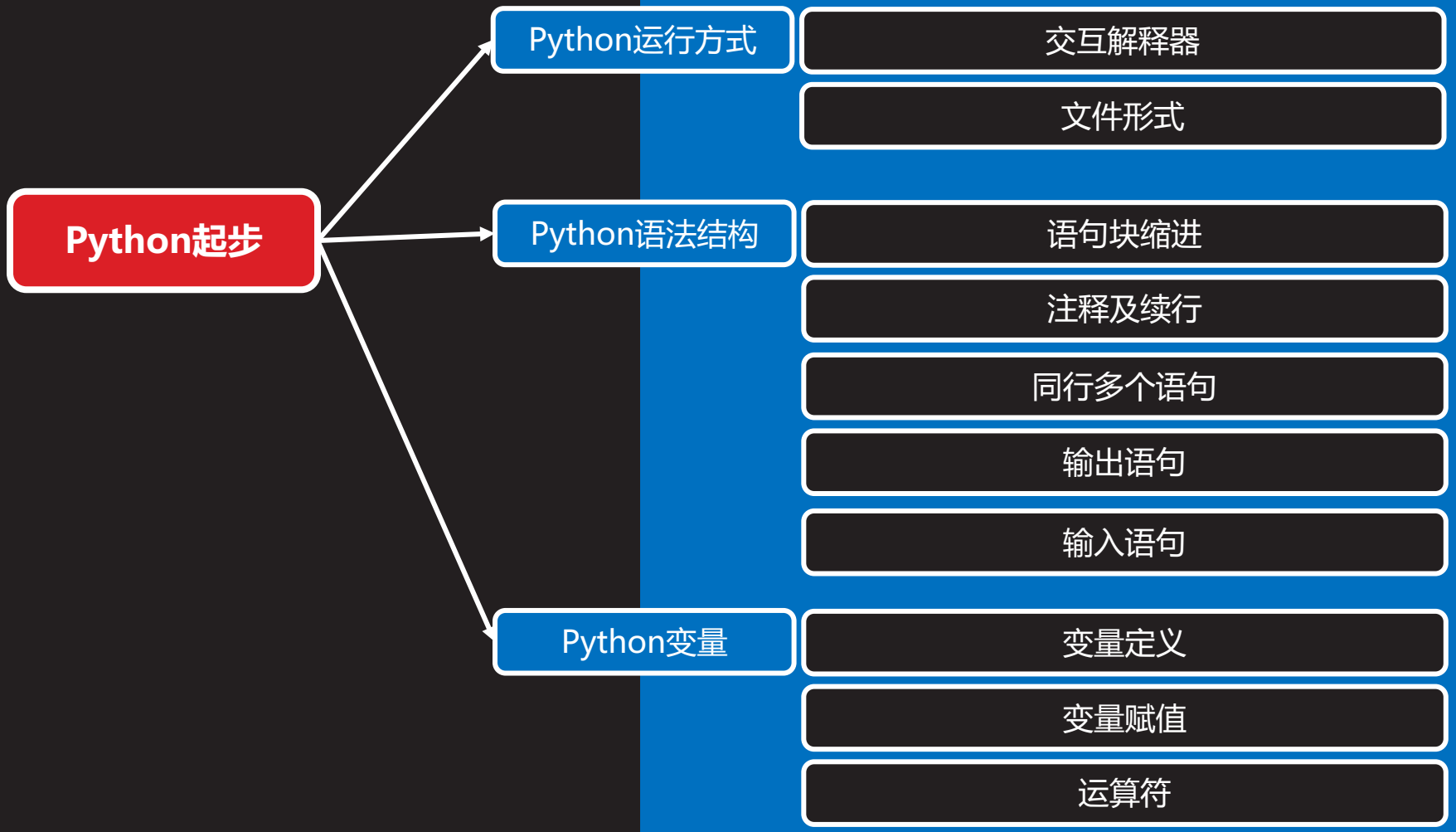
- Pycharm是由JetBrains打造的一款Python IDE
- 支持的功能有：
  - 调试、语法高亮
  - Project管理、代码跳转
  - 智能提示、自动完成
  - 单元测试、版本控制
- 下载地址：
  https://www.jetbrains.com/pycharm/download
- 分为收费的专业版和免费的社区版

案例1：准备python开发环境

1. 下载最新版本的python3
2. 下载pycharm社区版
3. 安装python3，使其支持Tab键补全
4. 配置pycharm，使其符合自己的习惯

课堂练习

Python起步


Python起步
Python运行方式
交互解释器
文件形式
Python语法结构
语句块缩进
注释及续行
同行多个语句
输出语句
输入语句
Python变量
变量定义
变量赋值
运算符

Python运行方式

交互解释器

- 进入交互解释器

```
[root@zzghost1 bin]# python3
Python 3.6.3 (default, Oct 13 2017, 11:38:12)
[GCC 4.8.5 20150623 (Red Hat 4.8.5-4)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

- 退出交互解释器

```
>>> exit()
或
>>> ctrl + d
```

知识讲解

文件形式

- 明确指定解释器

```
[root@zzghost1 day01]# python3 hello.py
```

- 赋予python文件可执行权限

```
[root@zzghost1 day01]# chmod +x hello.py
[root@zzghost1 day01]# ./hello.py
```

知识讲解

Tedu.cn
达内教育

Python语法结构

语句块缩进

- python代码块通过缩进对齐表达代码逻辑而不是使用大括号
- 缩进表达一个语句属于哪个代码块
- 缩进风格
  - 1或2：可能不够，很难确定代码语句属于哪个块
  - 8至10：可能太多，如果代码内嵌的层次太多，就会使得代码很难阅读
  - 4个空格：非常流行， 范·罗萨姆支持的风格

知识讲解

语句块缩进（续1）

- 缩进相同的一组语句构成一个代码块，称之代码组
- 首行以关键字开始，以冒号：结束，该行之后的一行或多行代码构成代码组
- 如果代码组只有一行，可以将其直接写在冒号后面，但是这样的写法可读性差，不推荐

知识讲解

注释及续行

- 首要说明的是：尽管Python是可读性最好的语言之一，这并不意味着程序员在代码中就可以不写注释
- 和很多UNIX脚本类似，python注释语句从#字符开始
- 注释可以在一行的任何地方开始，解释器会忽略掉该行#之后的所有内容
- 一行过长的语句可以使用反斜杠\分解成几行

知识讲解

同行多个语句

- 分号；允许你将多个语句写在同一行上
- 但是些语句不能在这行开始一个新的代码块
- 因为可读会变差，所以不推荐使用

知识讲解

输出语句

- 获取帮助

```
>>> help(print)
```

- 使用方式

```
>>> print('Hello World!')
>>> print('Hello' + 'World!')
>>> print('Hello', 'World!')
>>> print('Hello', 'World!', sep='***')
>>> print('Hello', 'World!', sep='***', end='')
```

知识讲解

输入语句

- 获得帮助

```
>>> help(input)
```

- 使用方式（注意，返回值一定是字符类型）

```
>>> num = input("Number: ")
Number: 20
>>> num + 10
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  TypeError: must be str, not int
```

知识讲解

案例2：模拟用户登陆

1. 创建名为login.py的程序文件
2. 程序提示用户输入用户名
3. 用户输入用户名后，打印欢迎用户

课堂练习

Tedu.cn
达内教育

Python变量

变量定义

- 变量名称约定
  - 第一个字符只能是大小写字母或下划线
  - 后续字符只能是大小写字母或数字或下划线
  - 区分大小写
- python是动态类型语言，即不需要预先声明变量的类型

知识讲解

变量定义（续1）

- 推荐采用的全名方法
  - 变量名全部采用小写字母
  - 简短、有意义
  - 多个单词间用下划线分隔
  - 变量名用名词，函数名用谓词（动词+名词）
  - 类名采用驼峰形式

知识讲解

变量赋值

知识讲解

- 变量的类型和值在赋值那一刻被初始化
- 变量赋值通过等号来执行

```
>>> counter = 0
>>> name = 'bob'
```

- python也支持增量赋值

```
>>>  n += 1      #等价于n = n + 1
>>>  n *= 1      #等价于n = n * 1
>>> i++
  File "<stdin>", line 1
    i++
      ^
SyntaxError: invalid syntax
```

运算符

- 标准算术运算符

  + - * / // % **

- 比较运算符

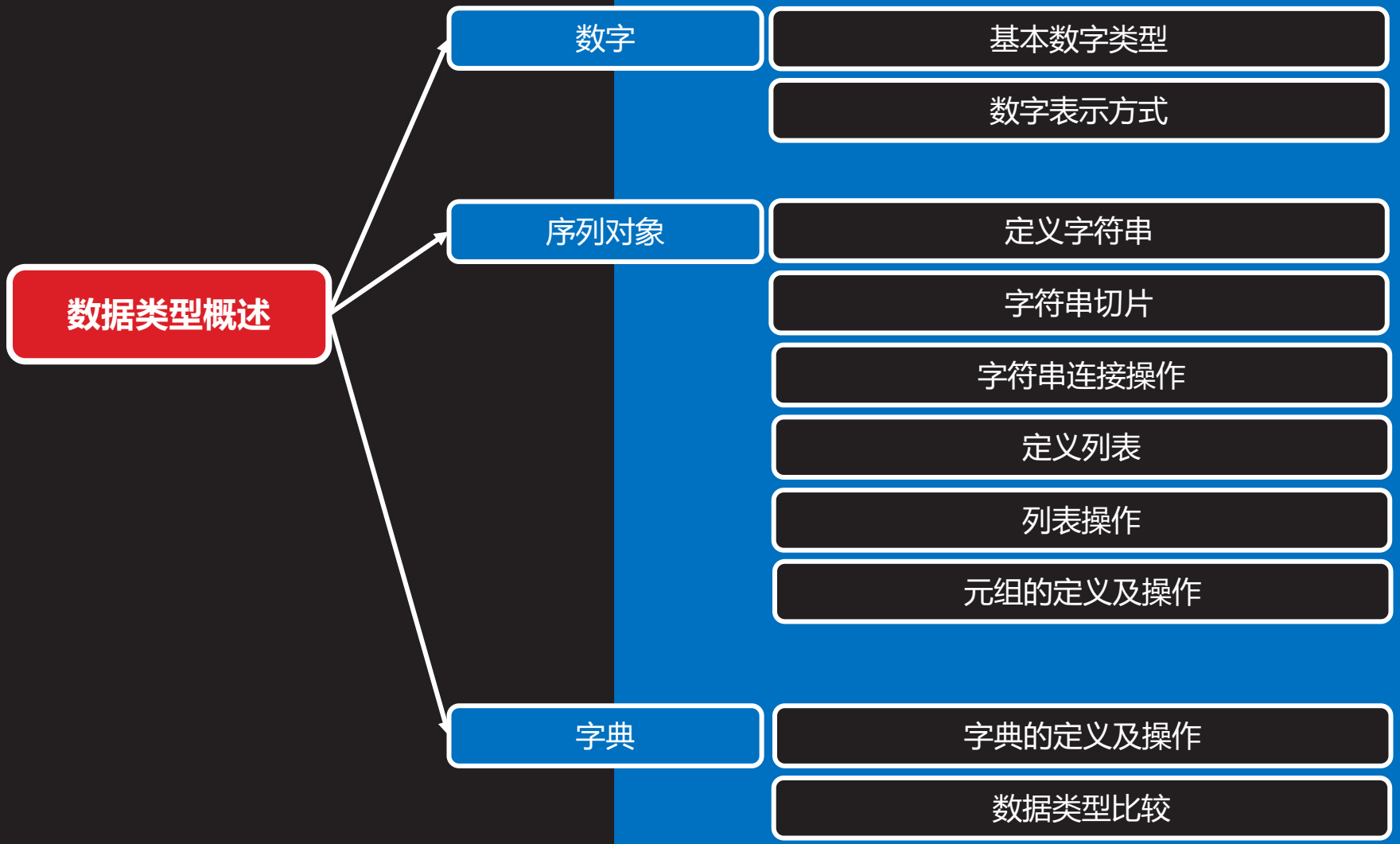
  < <= > >= == !=

- 逻辑运算符

  and not or

知识讲解

数据类型概述


数据类型概述
数字
基本数字类型
数字表示方式
序列对象
定义字符串
字符串切片
字符串连接操作
定义列表
列表操作
元组的定义及操作
字典
字典的定义及操作
数据类型比较

Tedu.cn
达内教育

数字

基本数字类型

- int：有符号整数
- bool：布尔值
  - True：1
  - False：0
- float：浮点数
- complex：复数

知识讲解

数字表示方式

- python默认以十进制数显示
- 数字以0o或0O开头表示为8进制数
- 数字以0x或0X开头表示16进制数
- 数字以0b或0B开头表示2进制数

知识讲解

Tedu.cn
达内教育

字符串

定义字符串

- python中字符串被定义为引号之间的字符集合
- python支持使用成对的单引号或双引号
- 无论单引号，还是双引号，表示的意义相同
- python还支持三引号（三个连续的单引号或者双引号），可以用来包含特殊字符
- python不区分字符和字符串

知识讲解

字符串切片

- 使用索引运算符[ ]和切片运算符[：]可得到子字符串
- 第一个字符的索引是0，最后一个字符的索引是-1
- 子字符串包含切片中的起始下标，但不包含结束下标

```
>>> py_str = 'python'
>>> py_str[0]
'P'
>>> py_str[-2]
'o'
>>> py_str[2:4]
'th'
>>> py_str[2:]
'thon'
>>> py_str[:4]
'Pyth'
```

知识讲解

字符串连接操作

- 使用+号可以将多个字符串拼接在一起
- 使用*号可以将一个字符串重复多次

```
>>> py_str = 'python'
>>> is_cool = 'is Cool‘
>>> print py_str + ' ' + is_cool
python is Cool
>>> py_str * 2
'pythonpython'
```

知识讲解

定义列表

- 可以将列表当成普通的“数组”，它能保存任意数量任意类型的python对象
- 像字符串一样，列表也支持下标和切片操作
- 列表中的项目可以改变

```
>>> alist = [1, "tom", 2, "alice"]
>>> alist[1] = 'bob‘
>>> alist[2:]
```

知识讲解

列表操作

- 使用in或not in判断成员关系
- 使用append方法向列表中追加元素

```
>>> alist = [1, "tom", 2, "alice"]
>>> 'tom' in alist
True
>>> 'alice' not in alist
False
>>> alist.append(3)
>>> alist[5] = 'bob'
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  IndexError: list assignment index out of range
```

知识讲解

元组的定义及操作

- 可以认为元组是“静态”的列表
- 元组一旦定义，不能改变

```
>>> atuple = (1, "tom", 2, "alice")
>>> 'tom' in atuple
True
>>> atuple[0] = 3
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  TypeError: 'tuple' object does not support item assignment
```

知识讲解

字典

字典的定义及操作

- 字典是由键-值(key-value)对构成的映射数据类型
- 通过键取值，不支持下标操作

知识讲解

```
>>> user_dict = {'name':'bob', 'age':23}
>>> use_dict['gender'] = 'male'
>>> 'bob' in user_dict
False
>>> 'name' in user_dict
True
>>> user_dict[0]
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  KeyError: 0
```

数据类型比较

- 按存储模型分类
  - 标量类型：数值、字符串
  - 容器类型：列表、元组、字典
- 按更新模型分类：
  - 可变类型：列表、字典
  - 不可变类型：数字、字符串、元组
- 按访问模型分类
  - 直接访问：数字
  - 顺序访问：字符串、列表、元组
  - 映射访问：字典

知识讲解

Tedu.cn
达内教育

总结和答疑