

# Python程序语言入门与应用



python

Life is short, use Python  
人生苦短，我用Python



# 课程回顾



## 🔗 第13章 Python办公自动化

🔗 Office办公自动化

🔗 python-docx与Word办公自动化

🔗 openpyxl与Excel办公自动化

🔗 python-pptx与PPT办公自动化



# 上周练习题



## 13.1 请假条word文档模板化生成

要包含如下信息：申请表标题、姓名、学号、院系、年级、请假理由、请假时间、请假天数、辅导员意见和签字。

新乡医学院请假条

填表日期：{{year}}年{{month}}月{{day}}日

姓名	{{name}}	学号	{{sno}}	院系	{{college}}
年级	{{grade}}		联系方式	{{phone}}	
请假时间	{{syear}}年{{smonth}}月{{sday}}日至{{eyear}}年{{emonth}}月{{eday}}日		请假天数	{{dno}}天	
请假事由	{{reason}}		本人签字：_____ 年 月 日		
辅导员意见	{{reply}}		辅导员签字：_____ 年 月 日		

模板

新乡医学院请假条

填表日期：2019年12月13日

姓名	郭德纲	学号	2017001	院系	生命科学技术学院
年级	2017级		联系方式	18903730000	
请假时间	2019年12月20日至2019年12月日			请假天数	5天
请假事由	世界那么大，我想去看看。 本人签字：_____ 年 月 日				
辅导员意见	同意 辅导员签字：_____ 年 月 日				

生成示例



# 上周练习题



## 13.1 请假条word文档模板化生成

要包含如下信息：申请表标题、姓名、学号、院系、年级、请假理由、请假时间、请假天数、辅导员意见和签字。

```
from docxtpl import DocxTemplate
from datetime import datetime

tpl = DocxTemplate('请假条模板.docx')
context = {
    'name': '郭德纲', 'sno': '2017001', 'college': '生命科学技术学院',
    'grade': '2017级', 'phone': '18903730000', 'syear': 2019, 'smmonth': 12,
    'sday': 16, 'eyear': 2019, 'emmonth': 12, 'sday': 20, 'dno': 5,
    'reason': '世界那么大,我想去看看。', 'reply': '同意',
    'year': datetime.now().year, 'month': datetime.now().month,
    'day': datetime.now().day,
}
tpl.render(context)
tpl.save('{}的请假条.docx'.format(context['name']))
```

Python代码示例



# 上周练习题



## 13.2 Excel图表绘制

使用openpyxl库自由选择数据绘制任意形式的图表。

```
from openpyxl import Workbook
from openpyxl.chart import LineChart, Reference

wb = Workbook()
ws = wb.active
rows = [
    ['品牌', '华为', 'OPPO', 'vivo', '小米', '苹果'],
    ['Q1', 24.2, 18.9, 16.3, 15.1, 11.3],
    ['Q2', 27.2, 20.2, 19.0, 14.2, 8.0],
    ['Q3', 24.6, 20.4, 21.7, 14.0, 7.4],
    ['Q4', 29.0, 19.6, 18.8, 10.0, 11.5],
]
chart_loc = ['A8', 'H8', 'A21', 'H21']

for row in rows:
    ws.append(row)
```

```
for stl in range(1, 5):
    c1 = LineChart()
    c1.title = "2018年手机市场占有率 (中国)"
    c1.style = stl
    c1.y_axis.title = '市场占有率 (%)'
    c1.x_axis.title = '季度'
    data = Reference(ws, min_col=2, min_row=1,
max_col=6, max_row=5)
    c1.add_data(data, titles_from_data=True)
    c1.width = 12
    c1.height = 6
    ws.add_chart(c1, chart_loc[stl-1])
wb.save("13.2charts.xlsx")
```

Python代码示例

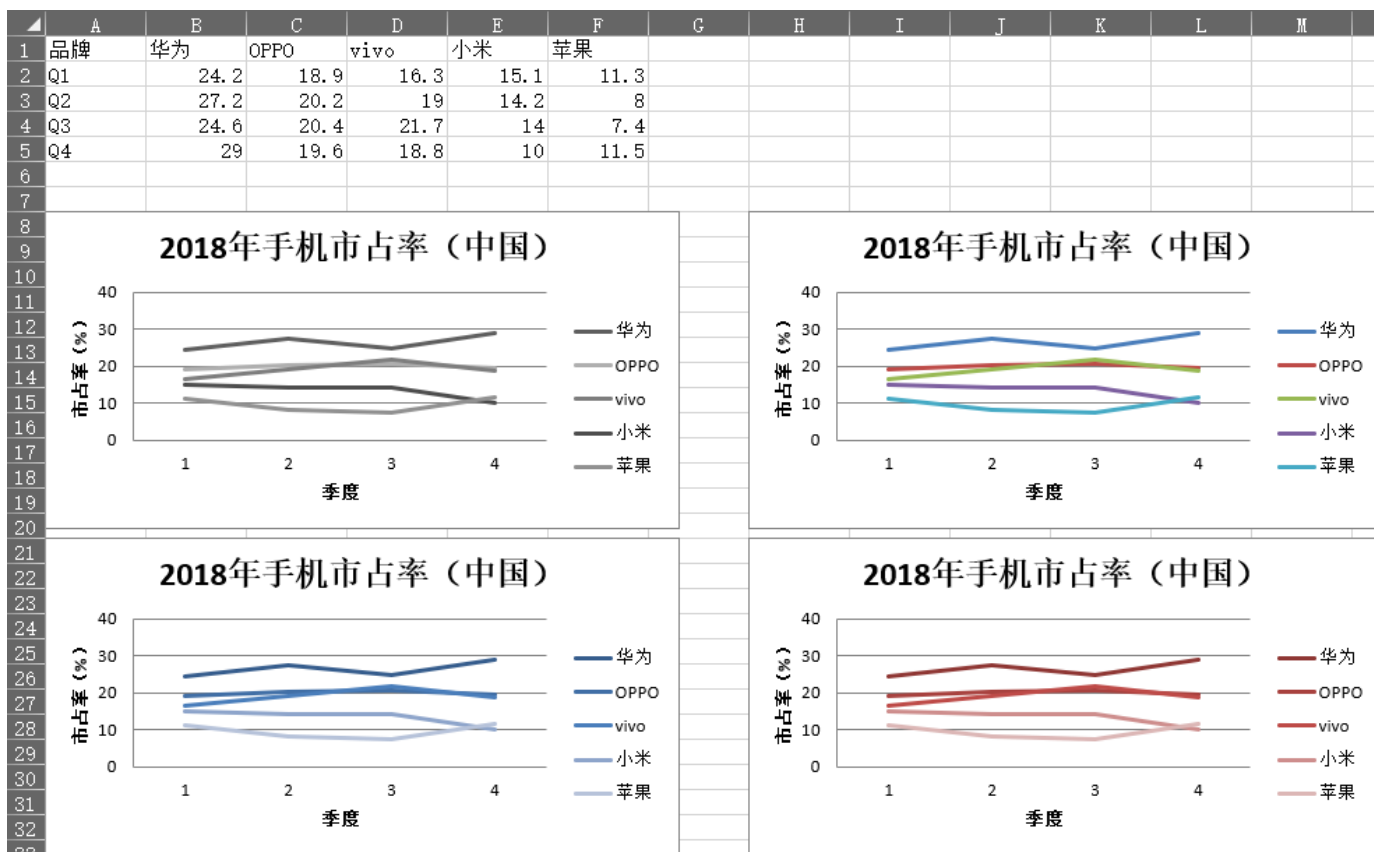


# 上周练习题



## 13.2 Excel图表绘制

使用openpyxl库自由选择数据绘制任意形式的图表。





新乡医学院

## Python程序语言入门与应用

# 考试大纲和题型解析



郭长江

changjiangguo@xxmu.edu.cn

生命科学技术学院

新乡医学院





# 1. 考试大纲





# 1. 考试大纲



## 考试大纲

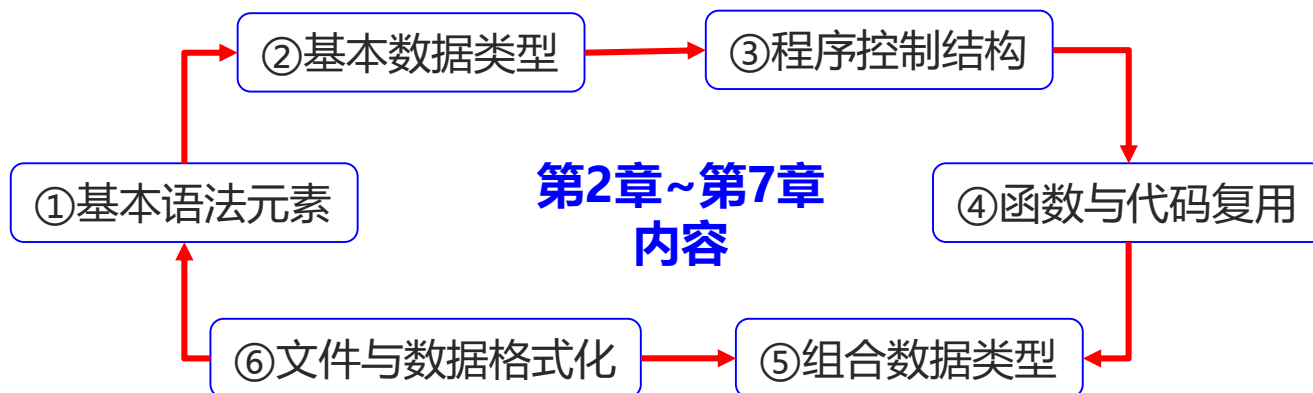
参考全国计算机等级考试二级Python语言考试大纲

考纲

Python基本语法

Python计算生态

以“基本语法”考核为主，包括Python过程式编程





# 1. 考试大纲



## ❖ 考试大纲

### ❖ Python计算生态

❖ Python计算生态覆盖标准库和第三方库，包括程序考核和名称考核。

#### ❖ 名称考核

✓ 重点考核库名称与功能类别的对应关系。



# 1. 考试大纲



## ❖ 1.1 Python基本语法元素（3项内容）

❖ 考点①：程序的**格式框架**、**缩进**、**注释**、**变量**、**命名**、**保留字**、**数据类型**、**赋值语句**与**引用**。

❖ **格式框架**：表达语句**从属关系**，由**缩进**表达，使用if/elif/else/for/while/def等保留字在**行尾加冒号**实现。体现代码可读性，缩进一般采用**4个空格**，亦可使用**相同数量**的**空白字符**。

❖ **注释**：不被解释器执行，**单行注释使用#引导**，**多行注释用'''或'''括起来**

❖ **变量**：用户定义用于保存和表示数据值，**无需预先声明**（随时命名、随时赋值、随时使用）



# 1. 考试大纲



## ❖ 1.1 Python基本语法元素（3项内容）

❖ 考点①：程序的格式框架、缩进、注释、变量、命名、保留字、数据类型、赋值语句与引用。

❖ 命名：给变量等元素关联名称，使用**大小写字符**、**数字**、**\_**、**汉字**等字符及其组合进行命名，但**首位不能是数字**，且不能使用空格、连字符、标点符号、引号或其他字符、**大小写敏感**且长度无限制

❖ 保留字：内部定义并保留使用的标识符，如**if/elif/else**、**for/in/while**、**break/continue/pass**、**def/class/return**、**not/and/or**、**True/False**、**from/import/as**、**with**、**try/except/raise/finally**、**lambda**、**global**等




# 1. 考试大纲

## 1.1 Python基本语法元素 (3项内容)

🔗 考点①：程序的格式框架、缩进、注释、变量、命名、保留字、数据类型、赋值语句与引用。

数据类型：表达数据含义，消除计算机对数据理解的二义性，主要有**整数、浮点数、复数、字符串、列表、字典**6种类型。

 **赋值：** 将等号（=）右侧表达式的结果赋给左侧变量。

 引用：调用其他功能库，使用import，分为4种方式：

- ✓ import <库名> (全命名空间引用) <库名>.<函数名>()
- ✓ from <库名> import <函数> (具体函数引用) <函数名>()
- ✓ from <库名> import \* (全函数引用) <函数名>()
- ✓ import <库名> as <别名> (库别名引用) <库别名>.<函数名>()



# 1. 考试大纲



## ❖ 1.1 Python基本语法元素（3项内容）

❖ 考点②：基本输入输出函数：input()、eval()、print()

❖ input()函数从控制台获得输入并以字符串类型返回

`<变量>=input(<"提示性文字">)`

❖ eval()函数去掉字符串外侧引号并执行字符串内容

`<变量>=eval(<字符串>)`

❖ print()函数用于输出(运算)结果，有3种用法：

✓ 单变量输出：`print(<变量>)`

✓ 多变量输出：`print(<变量1>, <变量2>, ..., <变量n>)`

✓ 混合输出（槽格式）：`print(<字符串模板>.format(<变量1>, <变量2>, ..., <变量n>))`



# 1. 考试大纲



## ❖ 1.1 Python基本语法元素（3项内容）

### ❖ 考点③：代码书写风格与语言特点

❖ Python之禅显示方法：`import this`

❖ 主要风格：优美、明了、简洁、不凌乱、扁平、间隔、可读

❖ 3个最主要的特点：`通用脚本语言`、`语法简洁`、`生态高产`

❖ 其他特点：`跨平台`、`强制可读`、`支持中文`、`模式多样`、`类库便捷`

❖ 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。

```
Beautiful is better than ugly.  
Explicit is better than implicit.  
Simple is better than complex.  
Complex is better than complicated.  
Flat is better than nested.  
Sparse is better than dense.  
Readability counts.  
.....
```



# 1. 考试大纲



## ❖ 1.2 Python基本数据类型（5项内容）

### ❖ 考点①：数字类型：整数、浮点数和复数

❖ **整数**：没有取值范围限制，有4种进制表示形式，默认采用**十进制**，其他进制需要使用引导符号：

✓ 二进制(0b/0B)、十进制、八进制(0o/0O)、十六进制(0x/0X)

❖ **浮点数**：数值范围和小数精度受系统限制，有2种表示方法

✓ 十进制的一般表示和科学计数法表示  $\langle a \rangle E \langle b \rangle = a * 10^b$

✓ 浮点数运算存在**不确定尾数**问题，可通过**round(x,d)**控制。

❖ **复数**：表示方法a+bj，实数部分(real属性)+虚数部分(imag属性)（不考核）





# 1. 考试大纲



## ❖ 1.2 Python基本数据类型 (5项内容)

### ❖ 考点②：数字类型的运算

#### ❖ 数值运算操作符

- ✓  $+$ (和),  $-$ (差),  $*$ (积),  $/$ (商),  $//$ (整除),  $%$ (取余/模运算),  $**$ (次幂),  $-$ (负值)
- ✓ 二元运算操作符与 $=$ 相连:  $x \text{ op} = y \rightarrow x = x \text{ op} y$

#### ❖ 数值运算函数

- ✓  $\text{abs}()$  (绝对值),  $\text{divmod}()$  (整除, 取余),  $\text{pow}()$  (幂运算),  $\text{round}()$  (四舍五入),  $\text{max}()$  (最大值),  $\text{min}()$  (最小值)



# 1. 考试大纲



## ❖ 1.2 Python基本数据类型（5项内容）

### ❖ 考点③：字符串类型及格式化

❖ 索引：单字符检索 <字符串>[正向序号/反向序号]

❖ 切片：子串或区间检索 <字符串>[N:M]

❖ 字符串格式化：**format方法**实现 <模板字符串>.format()

- ✓ 槽{}：包括参数序号和格式化控制信息{<序号>:<控制标记>}
- ✓ 序号可省略，也可使用字典形式
- ✓ 控制标记：引导符号(:)、填充字符、对齐方式(<,>^)、输出宽度、千分位分隔符、浮点数精度(.N)、数据类型(s,d,f,e,%)



# 1. 考试大纲



## ❖ 1.2 Python基本数据类型（5项内容）

### ❖ 考点④：字符串类型的操作

#### ❖ 字符串操作符

- ✓ + (连接), \* (复制), in (子串判断)

#### ❖ 字符串处理函数 <函数>(<字符串变量>)

- ✓ len() (长度), str() (转换为字符串), chr() (Unicode编码对应字符), ord() (字符对应Unicode编码), hex() (整数对应十六进制), oct() (整数对应八进制)

#### ❖ 字符串处理方法 <字符串变量>.<方法>(<方法参数>)

- ✓ .lower() (小写), .upper() (大写), .split() (拆分为列表), .count() (子串次数), .replace(替换), .strip() (去除首尾字符), .join() (列表连接为字符串)



# 1. 考试大纲



## ❖ 1.2 Python基本数据类型 (5项内容)

### ❖ 考点⑤：类型判断和转换

#### ❖ 类型判断

✓ `type(<变量>)`

#### ❖ 类型转换

✓ `int(x)`: 将浮点数或字符串x转换为整数

✓ `float(x)`: 将整数或字符串x转换为浮点数

✓ `str(x)`: 将整数或浮点数x转换为字符串



# 1. 考试大纲



## ❖ 1.3 Python程序控制结构（4项内容）

❖ 考点①：3种控制结构：顺序结构、分支结构、循环结构

❖ 考点②：分支结构：单/二/多分支结构

❖ if、if-else、if-elif...-else

❖ if-else简单形式：<表达式1> if <条件> else <表达式2>

❖ 考点③：循环结构

❖ 遍历循环：for ... in <字符串、文件、range()、组合数据>：

❖ 无限循环：while <条件>：

❖ 循环控制：break跳出当前循环，continue结束当前当次循环

❖ 考点④：异常处理：try-except



# 1. 考试大纲



## ❖ 1.4 Python函数和代码复用（3项内容）

### ❖ 考点①：函数的定义和使用

❖ **定义：** `def <函数名>(<参数列表>):`  
    <函数体>  
    `return <返回值列表>`

❖ **使用/调用：** `<函数名>(<赋值参数列表>)`

### ❖ 考点②：函数的参数传递：可选参数、参数名称、返回值

❖ **可选参数：** `def <函数名>(<可选参数>=<默认值>):`

❖ **参数名称：** `<函数名>(<参数名>=<实际值>)`

❖ **返回值：** `return <返回值列表>`



# 1. 考试大纲



## ❖ 1.4 Python函数和代码复用（3项内容）

### ❖ 考点③：变量作用域

#### ❖ 局部变量

- ✓ 函数内部使用，仅在函数内部有效，函数退出后不再存在

#### ❖ 全局变量

- ✓ 函数之外定义，在程序执行全过程有效
- ✓ 使用global声明：global <全局变量>



# 1. 考试大纲

## 1.5 Python组合数据类型 (5项内容)

## 🌀 考点①：组合数据类型的基本概念与表示方法

- 集合类型：元素集合，无序、唯一（集合{}
- 序列类型：元素向量，有序可重复（字符串"、元组()、列表[]
- 映射类型：“键-值”数据项组合（字典{:}

## 🔄 考点②：列表类型

- 定义：0个或多个元素的有序序列，用[]表示或list()转换
- 索引：ls[N] 空列表：[]、list()
- 切片：ls[N:M:K]，如 ls[::-1]（列表反转）





# 1. 考试大纲



## ❖ 1.5 Python组合数据类型（5项内容）

### ❖ 考点③：列表的操作

#### ❖ 操作函数 <函数>(<列表变量>)

- ✓ `len()`（元素个数）、`min()`（最小元素）、`max()`（最大元素）

#### ❖ 操作方法 <列表变量>.<方法>(<方法参数>)

- ✓ `.append(x)`（末尾增加元素）、`.extend(lt)`（列表lt元素增加到ls中）、`.insert(x)`（指定位置添加元素）、`.clear()`（删除所有元素）、`.pop(i)`（取出指定元素并删除，默认末尾元素）、`.remove(x)`（删除第一个出现的x）、`.reverse()`（元素反转）、`.copy()`（复制所有元素）



# 1. 考试大纲



## 1.5 Python组合数据类型（5项内容）

### 考点④：字典类型

- 定义：键-值对元素组合，使用方式{<键>:<值>,...}
- 索引：<字典>[<键>]      空字典：{}、dict()

### 考点⑤：字典的操作

- 操作函数 <函数>(<字典变量>)
  - ✓ len() (元素个数)、min() (最小键名)、max() (最大键名)
- 操作方法 <字典变量>.<方法>(<方法参数>)
  - ✓ .keys() (所有键)、.values() (所有值)、.items() (所有键值对)、.get(k,default) (键存在返回值，否则返回默认值)、.pop(k,default) (存在删除)、.popitem() (随机取出键值对)、.clear() (清空字典)



# 1. 考试大纲



## ❖ 1.6 文件与数据格式化 (5项内容)

### ❖ 考点①：文件的使用

#### ❖ 打开文件

- ✓ open()函数, 形式: `<变量>=open(<文件路径>,<打开模式>)`
- ✓ 打开模式  
r(只读), w(覆盖写), x(创建写), a(追加写), b(二进制), t(文本默认), + (模式叠加)

#### ❖ 关闭文件

- ✓ close()方法, 形式: `<变量>.close()`



# 1. 考试大纲



## ❖ 1.6 文件与数据格式化 (5项内容)

### ❖ 考点①：文件的使用

#### ❖ 读取文件 <变量>.<方法>(<方法参数>)

- ✓ `.readall()` (读入整个文件内容) , `.read(size=-1)` (读入整个或前size长度的文件内容) , `.readline(size=-1)` (读入一行或该行前size长度) , `.readlines(hint=-1)` (读入所有/前hint行形成列表) , `.seek(offset)` (改变指针位置, 0开头1当前2结尾)

#### ❖ 写入文件 <变量>.<方法>(<方法参数>)

- ✓ `.write(s)` (写入字符串)
- ✓ `.writelines(lines)` (写入字符串列表)



# 1. 考试大纲



## ❖ 1.6 文件与数据格式化（5项内容）

### ❖ 考点②：数据组织的维度

- ❖ 数据组织：一维/二维/多维/高维数据
- ❖ 一维数据：由对等关系的有序或无序数据构成，采用线性方式组织，对应于数学中的集合和数组。
- ❖ 二维数据：也称为表格数据，由关联关系数据构成，采用二维表格方式组织，对应于数学中的矩阵。
- ❖ 多维数据：由二维数据扩展而来
- ❖ 高维数据：由键值对构成



# 1. 考试大纲



## ❖ 1.6 文件与数据格式化（5项内容）

### ❖ 考点③：一维数据的处理

- ❖ 表示：线性结构，主要以列表形式表示，无序的可采用集合
- ❖ 存储：采用特殊字符分隔
  - ✓ 空格、制表符、逗号、换行及其他特殊字符
- ❖ 处理
  - ✓ 将各元素连接为字符串：.join()方法
  - ✓ 将字符串分解为列表：.split()方法
  - ✓ 遍历：for ... in ...



# 1. 考试大纲



## ❖ 1.6 文件与数据格式化 (5项内容)

### ❖ 考点④：二维数据的处理

❖ 表示：一维数据的组合，采用二维列表表示，每个元素对应于二维数据的一行（嵌套列表）

❖ 存储：二维表格

❖ 处理：遍历

```
for row in ls:  
    for item in row:  
        .....
```



# 1. 考试大纲



## ❖ 1.6 文件与数据格式化（5项内容）

### ❖ 考点⑤：采用CSV格式对一二维数据文件的读写

- ❖ **CSV格式**：逗号分隔，一种通用的、相对简单的文件格式
- ❖ **一维数据**：各元素采用逗号分隔，形成一行
- ❖ **二维数据**：每一行是一维数据，多行形成二维数据
- ❖ **读取&存储**：以二维数据为例

```
f=open('data.csv', 'r')  
ls = []  
for line in f:  
    ls.append(line.strip('\n').split(','))  
f.close()
```

读取CSV文件

```
f=open('data.csv', 'w')  
for row in ls:  
    f.write(','.join(row)+'\n')  
f.close()
```

存储为CSV文件





# 1. 考试大纲



## ❖ 1.7 Python计算生态（6项内容）

### ❖ 考点①：标准库：turtle库

❖ 海龟绘图，用于图形绘制

❖ 主要功能函数：

- ✓ 绘制状态： `pendown()/pd()`, `penup()/pu()`, `pensize()/width()`
- ✓ 颜色控制： `color()`, `pencolor()`, `begin_fill()`, `end_fill()`
- ✓ 运动控制： `forward()/fd()`, `backward()/bk()`, `right()/rt()`, `left()/lt()`, `setheading()/seth()`, `goto()`, `circle()`

**仅考核函数与功能对应关系，不涉及具体使用。**



# 1. 考试大纲



## ❧ 1.7 Python计算生态（6项内容）

### ❧ 考点②：标准库：random库

❧ 用于产生随机数，9个随机函数：

✓ `seed()`, `random()`, `randint()`, `getrandbits()`, `randrange()`,  
`uniform()`, `choice()`, `shuffle()`, `sample()`

### ❧ 考点③：其他标准库的功能

❧ `math`：数值计算

❧ `datetime`：日期和时间表达

❧ `json`：处理JSON格式

❧ `threading`：多线程

❧ `subprocess`：多进程



# 1. 考试大纲



## 🔗 1.7 Python计算生态（6项内容）

### 🔗 考点④：基本内置函数

🔗 共68个内置函数，其中考核：

✓ abs(), chr(), dict(), eval(), float(), input(), int(), len(), list(),  
max(), min(), open(), ord(), pow(), print(), range(), round(),  
sorted(), str(), sum(), type()

### 🔗 考点⑤：第三方库的获取、安装和卸载

🔗 安装： `pip install <拟安装第三方库名>`

🔗 卸载： `pip uninstall <已安装第三方库名>`



# 1. 考试大纲



## ❧ 1.7 Python计算生态（6项内容）

### ❧ 考点⑥：第三方库及应用场景

- ❧ pyinstaller：脚本程序转变为可执行程序
- ❧ jieba：中文文本分词
- ❧ wordcloud：根据文本生成词云
- ❧ PIL：图像归档和处理
- ❧ numpy：多维数组科学计算
- ❧ matplotlib：数据绘图
- ❧ beautifulsoup4：解析HTML网页格式
- ❧ python-docx：创建和编辑word文档
- ❧ openpyxl：excel读写与图表绘制



# 1. 考试大纲



## ❖ 1.7 Python计算生态（6项内容）

### ❖ 考点⑥：第三方库及应用场景

- ❖ 网络爬虫：requests、scrapy、pyspider
- ❖ 数据分析：numpy、pandas、scipy
- ❖ 文本处理：beautifulsoup4、pdfminer
- ❖ office办公：python-docx、openpyxl、python-pptx
- ❖ 数据可视化：matplotlib、seaborn、Plotly
- ❖ 图形用户界面：Tkinter、PyQt5、wxPython
- ❖ 机器学习：tensorflow、pytorch、scikit-learn、keras
- ❖ Web开发：Django、flask
- ❖ 游戏开发：pygame、Panda3D、cocos2D



## 2. 题型详解



## 2. 题型详解



### ❖ 二级考试

考试时间120分钟，满分100分。

#### ❖ 单项选择题（40分）

❖ 公共基础知识：10题（10分）

❖ Python知识：30题（30分）

#### ❖ 编程操作题（60分）

❖ 基础编程题：3题（15分）

❖ 简单应用题：2题（25分）

❖ 综合应用题：1题（20分）



## 2. 题型详解



### 本课程考试

考试时间80分钟，满分100分。

#### 单项选择题（75分）

 Python知识：25题

#### 编程操作题（25分）

 基础编程题：5题







## 2. 题型详解



### Python知识单选题

-  下面不符合Python语言命名规则的是 ( **C** )  
A. monthly B. Monthly C. 3monthly D. \_Monthly3\_
-  关于Python程序格式框架的描述, 错误的是 ( **A** )  
A. 不采用严格的缩进表明程序框架  
B. 缩进可使用Tab键实现  
C. 单层缩进代码属于前最近邻的一行非缩进代码  
D. 判断、循环、函数等语法形式能够通过缩进包含一组代码



## 2. 题型详解



### Python知识单选题

 turtle库绘制直线的函数是 (**A**)

A. fd() B. pd() C. circle() D. seth()

 假设 $x=1$ ,  $x*=3+5**2$ 的运算结果是 (**B**)

A.  $x=125$  B.  $x=28$  C.  $x=1$  D.  $x=13$

 假设 $s="PYTHON"$ ,  $s.lower()$ 的操作结果是 (**D**)

A. "pYHTON" B. "PYTHOn" C. "Python" D. "python"

 下列表达式错误的是 (**C**)





A.  $'abc' < 'ad'$  B.  $'abc' < 'abcd'$  C.  $'4' + '5' = '9'$  D.  $'a' + 'b' = 'ab'$



## 2. 题型详解



### Python知识单选题

-  循环结构可以使用（**B**）语句实现？  
A. print B. while C. loop D. if
-  可以终结一个循环的保留字是（**B**）  
A. if B. break C. exit D. continue
-  Python中定义函数的关键字是（**A**）  
A. def B. define C. function D. defunc
-  下列不是使用函数的优点的是（**D**）  
A. 减少代码重复 B. 使程序更加模块化  
C. 使程序便于阅读 D. 为了展现智力优势



## 2. 题型详解




### Python知识单选题

 `type(12.34)`的输出结果是 ( **C** )

- A. `<class 'complex'>`    B. `<class 'int'>`  
C. `<class 'float'>`        D. `<class 'bool'>`

 `len()`函数的使用对象不包含 ( **A** )

- A. 整数    B. 字符串    C. 列表    D. 字典

 `ls=[[2,3], [3,5,7],25,[0,9]]`, `len(ls)`值是 ( **B** )

- A. 8    B. 4    C. 2    D. 3

 下列是正确的字典创建方式的是 ( **D** )




- A. `d={ [1,2]:1 }`    B. `d=[ 'a':1 ]`    C. `d=( 'a':1 )`    D. `d={ 'a':[1,2] }`



## 2. 题型详解



### Python知识单选题

-  下列不是Python对文件的读操作方法是（**D**）  
A. read B. readline C. readall D. readtext
-  以下选项中是Python中文分词的第三方库的是（**B**）  
A. turtle B. jieba C. PIL D. wordcloud
-  以下选项中不是Python数据分析第三方库的是（**A**）  
A. requests B. numpy C. scipy D. pandas





## 2. 题型详解



### Python编程操作题

-  不使用任何模块，书写Python代码求解下列数学表达式的值。

$$x = \sqrt{\frac{(3^4 + 5 \times 6^7)}{8}}$$

-  代码1: `print(( ( 3**4+5*6**7) / 8 ) ** 0.5)`
-  代码2: `print(pow( ( 3**4+5*6**7) / 8 ,0.5))`



## 2. 题型详解



### Python编程操作题

- 根据输入的正整数n输出一个宽度为20字符，右对齐，带千位分隔符的效果，并使用“-”填充。如n=1234，则输出如下： -----1,234

```
n = input("请输入整数： ")  
_____ # 可以多行
```


- 代码1： `print("{:->20,}".format(eval(n)))`
- 代码2： `print("{:->20,}".format(int(n)))`



## 2. 题型详解




### Python编程操作题

 补充如下代码，使程序能够计算a中元素与b逐项乘积的累加和。

```
a = [[1,2,3],[4,5,6],[7,8,9]]  
b = [3,6,9]  
____①____  
for c in a:  
    for j in ____②____:  
        s += c[j] + b[j]  
print(s)
```

 代码1: ① `s = 0` ② `range(len(c))`

 代码2: ① `s = 0` ② `range(len(b))`

 代码3: ① `s = 0` ② `range(3)`





## 2. 题型详解



### Python编程操作题

- 补充如下代码，实现将列表`ls=[23, 45, 78, 11, 67, 89, 13, 243, 56]`中的奇数去除，并输出去除奇数后列表`ls`的元素个数。

```
def isOdd(n):  
    if n % 2 != 0:  
        ①  
  
ls=[23, 45, 78, 11, 67, 89, 13, 243, 56]  
lt = ls.copy()  
    ②  
for i in lt:  
    if isOdd(i) != True:  
        ③  
print(len(ls))
```

- 代码：① `return True` ② `ls=[]`或`ls.clear()` ③ `ls.append(i)`



## 2. 题型详解



### 试题资源

#### Python 全国二级等级考试小知识

<https://python123.io/index/ncre>

#### Python 全国二级等级考试免费模拟考试

<https://python123.io/student/series/5/catalogs/1533901866519/modules/46>



# 练习题



将前13章未完成的作业  
继续完成。

截止时间：2019年12月31日

(已完成左右的同学也可以将所有作业按  
右侧目录组织进行打包发送到邮箱)

.py代码文件和结果文件打包(姓名  
+学号/章节号/作业代码)发送到  
python\_xxmu@163.com

./张三2019001/

├── 01

│ ├── 1.1.py

│ └── 1.2.py

├── 02

│ └── 2.1.py

├── 03

│ └── 3.1.py

├── 04

│ └── 4.1.py

├── 05

│ └── 5.1.py

├── 06

│ └── 6.1.py

├── 07

│ └── 7.1.py

├── 08

│ └── 8.1.py

├── 09

│ └── 9.1.py

├── 10

│ └── 10.1.py

├── 11

│ └── 11.1.py

├── 12

│ └── 12.1.py

├── 13

│ └── 13.1.py

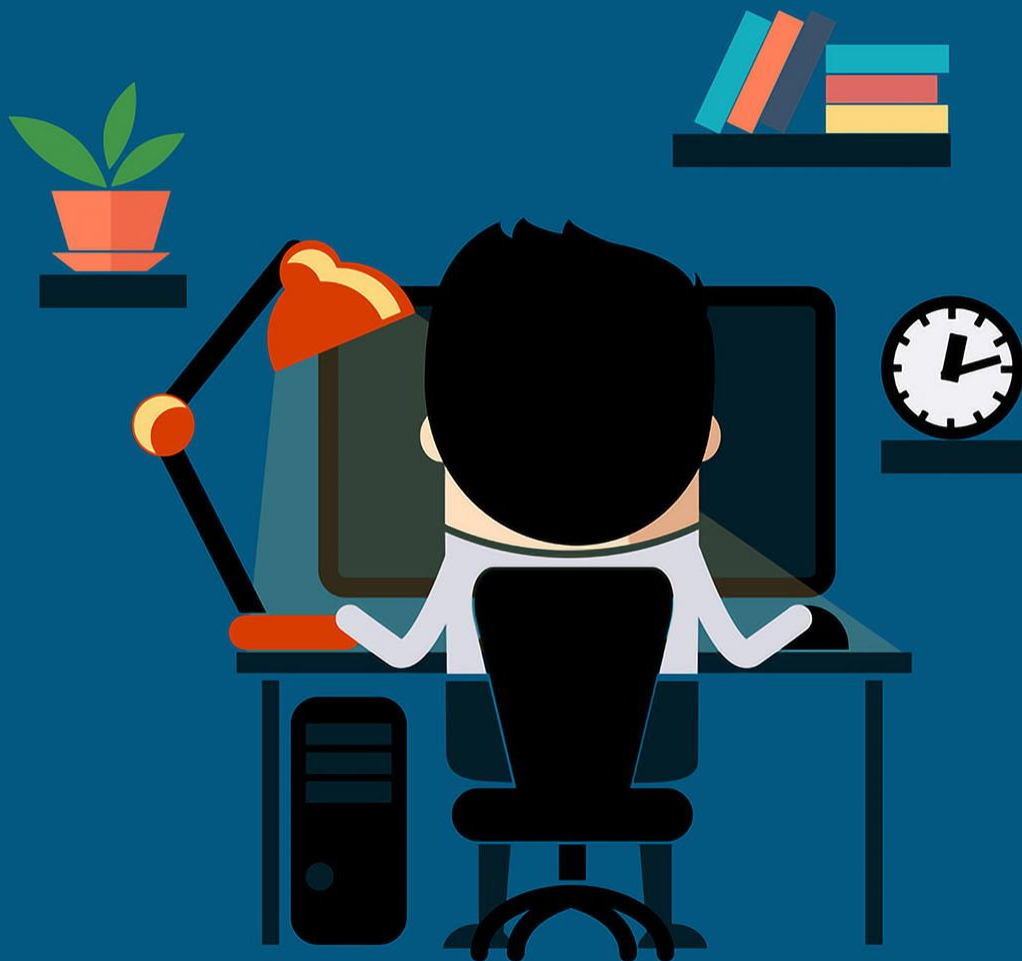


# 下周安排



- ❧ 期末考试 (60%)
  - ❧ 开卷笔试 (**仅限于**携带**纸质材料**)
  - ❧ 内容为二级Python语言考试大纲结合课堂内容
  - ❧ 考试时间: 2019年12月20日19:00~20:20

编程辣么好，还等什么？开始学习吧！



Programing is an Art