**实验1 开始python编程**

**目的和要求**

1. 了解什么是python?
2. 了解python的特性
3. 学习下载和安装python
4. 学习执行python命令和脚本文件的方法
5. 学习python语音的基本语法
6. 下载和安装Anaconda的方法
7. 学习使用python的集成开发环境Anaconda&spyder的方法

**实验准备**

了解python

（1）简单易学

（2）Python是开源的、免费的

（3）Python是高级语言

（4）高可移植性

（5）Python是解释型语言

（6）Python全面支持面向对象的程序设计思想

（7）高可扩展性

（8）支持嵌入式编程

（9）功能 强大的开发库

**实验内容**

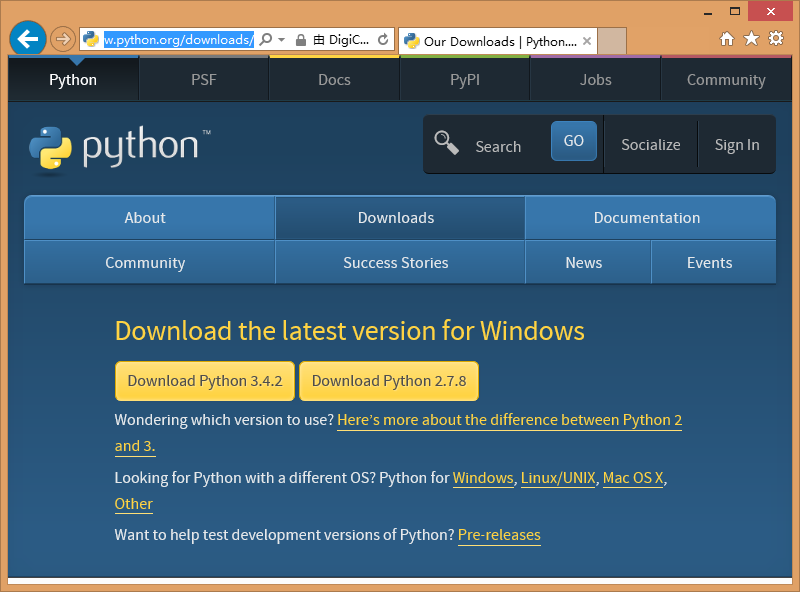
本实验主要包含以下内容

1. 练习下载python
2. 练习安装python
3. 练习执行python命令和脚本文件
4. 练习下载和安装pywin32
5. 练习使用python的文本编辑juper Notebook
6. 练习使用python的集成开发环境spyder
7. 下载python

* 访问如下网址：**https://www.python.org/downloads/**

选择下载python3.4系列最新版本

1. 安装python



* **在Windows 7中安装后，在开始菜单的所有程序中会出现一个Python2.7分组。单击其下面的Python 2.7 (command line - 32 bit)菜单项，就可以打开python命令窗口，如图1-5所示。也可以打开Windows命令窗口，然后运行python命令，来打开python命令窗口。**

1. **执行python命令和脚本**

* **创建一个文件MyfirstPython.py，使用记事本编辑它的内容如下：**

**# My first Python program**

**print('I am Python')**

* **保存后，打开命令窗口。切换到MyfirstPython.py所在的目录，然后执行下面的命令：**

**python MyfirstPython.py**

* **运行结果如下：**

**I am Python**

1. 下载和安装Pywin32

* **访问下面的网址可以下载Pywin32安装包。**
* **http://sourceforge.net/projects/pywin32/**

**5.使用python**文本编辑juper Notebook

输入以下脚本：

print （“Hello World!"）

print （"Hello Again"）

print （"I like typing this."）

print "This is fun."

print 'Yay! Printing.'

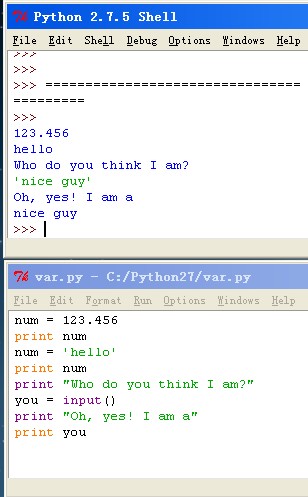
print "I'd much rather you 'not'."

print 'I "said" do not touch this.'

**熟悉input()函数**

python有一个接收命令行下输入的方法：  
input()

print （"Who do you think I am?"）  
input()  
print （"Oh, yes!"）

1. **练习：人机对话：要求根据姓名、性别、年龄……分别提问及回答。**
2. **完成如图所示的程序要求。  
   **
3. **编写一个猜年龄的小游戏。**

**4.编写程序，输入<人名 1>和<人名 2>，在屏幕上显示如下的新年贺卡**

**【源程序】**

**###################################**

**# 新年贺卡**

**# python0101.py**

**# 2015**

**###################################**

5.输入直角三角形两直角边a,b求斜边C,并输出。(from math import \*)

6.编写程序，输入球的半径，计算球的表面积和体积，半径为实数，用π，结果输出为浮点数，共10位其中2位有效数字。

**实验2 python语言基础**

**目的和要求**

1. **了解Python语言的基本语法和编码规范**
2. **掌握Python语言的数据类型、运算符、常量、变量、表达式和常用语句等基础知识，**
3. **学习Python常用语句**
4. **学习序列数据结构的方法。**

**实验准备**

* **练习使用常量和变量**
* **练习使用运算符和表达式**
* **练习使用常用语句**
* **练习使用序列数据结构**

1. 使用变量

参照下面的步骤练习使用变量

**【例2-3】 用id()函数输出变量地址的示例程序：**

**str1 = "这是一个变量";**

**print("变量str1的值是："+str1);**

**print("变量str1的地址是：%d" %(id(str1)));**

**str2 = str1;**

**print("变量str2的值是："+str2);**

**print("变量str2的地址是：%d" %(id(str2)));**

**str1 = "这是另一个变量";**

**print("变量str1的值是："+str1);**

**print("变量str1的地址是：%d" %(id(str1)));**

**print("变量str2的值是："+str2);**

**print("变量str2的地址是：%d" %(id(str2)));**

1. 使用运算符

参照下面的步骤练习使用运算符

**x =3**

**x += 3**

**print(x)**

**x -= 3**

**print(x)**

**x \*= 3**

**print(x)**

**x /= 3**

**print(x)**

1. 练习使用常用语句

**赋值语句**

**条件分支语句**

**循环语句**

**try-except语句**

**（1） a = 10;**

**a += 1;**

**print (a);**

**a\*= 10;**

**print (a);**

**a\*\*= 2;**

**print (a);**

（2）**参照【例2-13】、【例2-14】、【例2-15】练习使用if语句.**

**(3) 参照【例2-16】练习使用while语句.**

**(4) 参照【例2-17】练习使用for语句.**

**(5) 参照【例2-18】练习使用continue语句.**

**(6) 参照【例2-19】练习使用break语句.**

**(7) 参照【例2-16】练习使用try-except语句.**

**4.练习使用序列数据结构**

（1）**参照【例2-22】、【例2-23】、【例2-24】、【例2-25】【例2-28】……【例2-39】练习使用列表。**

（2）**参照【例2-40】、参照【例2-41】、……参照【例2-45】练习使用元组。**

（3）**参照【例2-46】、参照【例2-47】、……参照【例2-56】练习使用字典。**

（4）**参照【例2-57】、参照【例2-58】、……参照【例2-74】练习使用集合。**

**练习：**

**1.格式输出**

**print("%d %d %d"%(1,2,3))**

**print("%d %d %d"%(1.1,2.5,3.6))**

**print("%e %e %e"%(1.1,2.5,3.6))**

**print("%f %f %f"%(1.1,2.5,3.6))**

**print("%5.2f %5.3f %6.7f"%(1.1,2.5,3.6))**

**print("%10.2f %5.3f %6.7f"%(12345.12345,2.5,3.6))**

**2.判断闰年**

**3. 输入小明身高1.75，体重80.5kg。请根据BMI公式（体重除以身高的平方）帮小明计算他的BMI指数，并根据BMI指数：**

**低于18.5：过轻**

**18.5-25：正常**

**25-28：过重**

**28-32：肥胖**

**高于32：严重肥胖**

**4.利用下面的计算公式计算e的近似值，要求最后一项小于10-6**

**e=1+++……+**

**5.编写程序打印九九成法口诀表**

**6.输入两个整数，打印他们相除后的结果，若输入的不是整数或除数为0，进行异常处理。**

**实验3 python函数**

**目的和要求**

1. 了解函数的概念
2. 了解局部变量和全局变量的作用域
3. 学习声明和调用函数的方法
4. 学习在调试窗口查看变量的值
5. 学习使用函数的参数和返回值
6. 学习使用python的内置函数

**实验准备**

了解函数

（1）**函数（function）由若干条语句组成，用于实现特定的功能。函数包含函数名、若干参数和返回值。一旦定义了函数，就可以在程序中需要实现该功能的位置调用该函数，给程序员共享代码带来了很大方便。**

（2）**在Python语言中，除了提供丰富的系统函数（本书前面已经介绍了一下常用的系统函数）外，还允许用户创建和使用自定义函数。**

**（3）变量的作用域，在调试窗口中查看变量的值。**

（4）**在函数中传递参数并查看函数的返回值。**

**实验内容**

本实验主要包含以下内容

1. 练习声明和调用函数
2. 练习在调试窗口中查看变量的值
3. 练习使用函数参数和返回值
4. 学习使用python的内置函数
5. 声明和调用函数
6. 参照**【例3-1】、【例3-2】、【例3-3】练习创建python自定义函数**
7. 参照【**例3-4】、【例3-5】、【例3-6】练习创建调用函数**
8. 参照【**例3-7】练习使用局部变量和全局变量**
9. 在调试窗口中查看变量的值

参照下面的步骤在调试窗口中查看变量的值

1. 在IDLE中打开【**例3-7】的程序**
2. **设置断点**

**（3）例如在IDLE中打开例3-7.py，然后在菜单中选择Run / Python Shell，打开Python Shell窗口。在Python Shell的菜单中，选择Debug/ Debugger，Python Shell窗口中会出现下面文字，[DEBUG ON]同时打开Debug Control窗口**

3. 使用函数参数和返回值

参照下面的步骤练习使用函数的参数和返回值

1. 参照**【例3-8】练习在函数中按值传递参数**
2. 参照**【例3-9】练习打印形参和实参的地址**
3. 参照**【例3-10】练习使用列表作为函数的参数**
4. 参照**【例3-11】练习使用字典作为函数的参数**
5. 参照**【例3-12】练习在函数中修改列表参数**
6. 参照**【例3-13】练习在函数中修改字典参数**
7. 参照**【例3-14】、【例3-15】练习使用参数默认值**

（8）参照**【例3-16】、【例3-17】、【例3-18】、【例3-19】练习可变长参数**

（9）参照**【例3-20】、【例3-21】练习使用函数的返回值**

4.使用python内置函数

参照下面的步骤练习使用内置函数

1. 参照**【例3-22】练习使用数学运算函数**
2. **参照【例3-23】、【例3-24】、【例3-25】、【例3-26】、【例3-27】练习使用字符串处理函数**
3. **参照【例3-28】、【例3-29】练习使用help()**
4. **参照【例3-30】使用type()**

**练习：**

**实验4 python面向对象程序设计**

**目的和要求**

1. 了解面向对象的程序设计思想
2. 了解对象、类、封装、继承、方法、构造函数和析构函数等面向对象的程序设计的基本概念
3. 学习声明类
4. 学习静态变量、静态方法和类方法
5. 学习类的继承和多态
6. 学习复制对象的方法

**实验准备**

1. **面向对象编程是Python采用的基本编程思想，它可以将属性和代码集成在一起，定义为类，从而使程序设计更加简单、规范、有条理。**
2. 对象（Object）：面向对象程序设计思想可以将一组数据和与这组数据有关操作组装在一起，形成一个实体，这个实体就是对象。
3. 具有相同或相似性质的对象的抽象就是类。因此，对象的抽象是类，类的具体化就是对象。

**实验内容**

本实验主要包含以下内容

1. 练习声明类
2. 练习类的继承和多态
3. 练习复制对象
4. 声明类

参照下面的步骤练习声明类

1. 参照**【例4-2】练习定义类和使用对象**
2. 参照**【例4-3】练习定义类的成员变量**
3. 参照**【例4-4】、【例4-5】练习定义类的构造函数**
4. 参照**【例4-6】练习使用析构函数**
5. 参照**【例4-7】练习使用静态变量**
6. 参照**【例4-8】练习使用静态方法**
7. 参照**【例4-9】练习使用类方法**
8. 参照**【例4-10】练习使用instanceof关键字**
9. 类的继承和多态

参照下面的步骤练习类的继承和多态

1. 参照**【例4-11】练习类的继承**
2. 参照**【例4-12】、【例4-13】练习使用抽象类和多态**
3. 复制对象

参照下面的步骤练习复制对象

1. 参照**【例4-14】练习通过赋值复制对象**
2. 参照**【例4-15】、【例4-16】练习使用抽象类和多态**

**实验5 函数式编程**

**目的和要求**

1. **了解什么是函数式编程**
2. **函数式编程的优点**
3. **学习lambda表达式**
4. **学习使用map()函数**
5. **学习filter()函数**
6. **学习reduce()函数**
7. **学习zip()函数**
8. **学习闭包（closure）和递归函数**
9. **学习迭代器和生成器（Generator）**

**实验准备**

**函数式编程是一种典范。本章首先对函数式编程的基本概念进行介绍，然后介绍Python语言是如何实现函数式编程的。函数式编程一种编程的基本风格，也就是构建程序的结构和元素的方式。函数式编程将计算过程看作是数学函数，也就是可以使用表达式编程。在函数的代码中，函数的返回值只依赖传入函数的参数，因此使用相同的参数调用函数2次，会得到相同的结果。**

* **如果一个编程语言把函数视为头等函数，则可以称其拥有头等函数。拥有头等函数的编程语言可以将函数作为其他函数的参数，也可以将函数作为作为其他函数的返回值。可以把函数赋值给变量或存储在元组、列表、字典、集合和对象等数据结构中。有的语言还支持匿名函数。**
* **在拥有头等函数的编程语言中，函数名没有任何特殊的状态，而是将函数看作是function类型的二进制类型。**

**实验内容**

本实验主要包含以下内容

1. 练习使用python函数式编程常用的函数。
2. 练习使用闭包和递归函数。
3. 练习使用迭代器和生成器。
4. 使用python函数式编程常用的函数

参照下面的步骤练习python函数式编程常用的函数。

1. 参照**【例6-1】、【例6-2】和【例6-3】练习使用Lambda表达式。**
2. 参照**【例6-6】练习使用filter()函数。**
3. 参照**【例6-8】、【例6-9】和【例6-10】练习使用zip()函数。**
4. 参照**【例6-11】、【例6-12】练习使用比较普通编程方式和函数式编程。**
5. 使用闭包和递归函数
6. 参照**【例6-13】练习使用闭包**
7. 参照**【例6-14】练习使用递归函数**
8. 使用迭代器和生成器
9. 参照**【例6-15】练习使用使用iter()函数获取序列的迭代器对象。**
10. 参照**【例6-16】练习使用使用enumerate ()函数可以将列表或元组生成一个有序号的序列。**
11. 参照**【例6-17】、【例6-18】练习使用生成器。**

**实验6** python模块及I/O编程

**目的和要求**

1. 了解什么是模块
2. 学习使用标准库中的模块
3. 学习创建和使用自定义模块
4. 了解I/O编程的基本含义
5. 学习输入和显示数据的基本方法
6. 学习目录编程的基本方法

**实验准备**

**模块是Python语言的一个重要概念，它可以将函数按功能划分到一起，以便日后使用或共享给他人。可以使用Python标准库中的模块，也可以下载和使用第三方模块。**

**I/O是Input/Output的缩写，即输入输出接口。I/O接口的功能是负责实现CPU通过系统总线把I/O电路和外围设备联系在一起。I/O编程是一个程序设计语言的基本功能，常用的I/O操作包括通过键盘输入数据、在屏幕上打印信息和读写硬盘等。**

**实验内容**

1. **练习使用SYS模块**
2. **platform模块**
3. **练习使用 与数学有关的模块**
4. **练习使用 time模块**
5. **练习自定义和使用模块**
6. **练习输入和显示数据**
7. **练习文件操作**
8. **练习目录编程**
9. **使用SYS模块**

**参照【例5-1】——【例5-5】练习打印当前操作系统的名称、版本号、命令行参数及路径等**

1. **使用platform模块**

**参照【例5-6】——【例5-18】练习**

1. **练习使用 与数学有关的模块**

**参照【例5-19】——【例5-26】练习使用random模块**

**参照【例5-27】、【例5-28】练习使用dcimal模块**

**参照【例5-29】、【例5-30】练习使用fractions模块**

**参照【例5-31】——【例5-34】练习使用time模块**

1. **练习自定义和使用模块**

**参照【例5-35】、【例5-36】练习自定义和导入模块**

1. **输入和显示数据**

**参照【例7-1】——【例7-9】练习使用input()和print(\_)函数**

1. **文件操作**
   1. **——【例7-27】练习使用文件操作有关函数**

**7.目录编程**

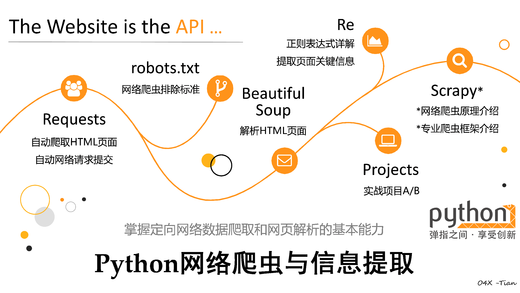
**参照【例7-28】——【例7-31】练习使用目录编程**

**实验7** Python网络爬虫与信息提取

**目的和要求**

1. 了解利用Python语言爬取网络数据并提取关键信息的技术和方法
2. 学习和掌握定向网络数据爬取和网页解析的基本能力。
3. 了解Python计算生态中最优秀的网络数据爬取和解析技术

**实验准备**



**实验内容**

（1）Python第三方库Requests，通过HTTP/HTTPS协议自动从互联网获取数据并向其提交请求的方法；

  （2）Robots协议，网络爬虫排除标准，礼貌合法获取信息的规范；

  （3）Python第三方库Beautiful Soup，从所爬取HTML页面中解析完整Web信息的方法；

  （4）Python标准库Re，从所爬取HTML页面中提取关键信息的方法。

**参照下面的实例来练习**

实验8 Python数据分析与展示

**目的和要求**

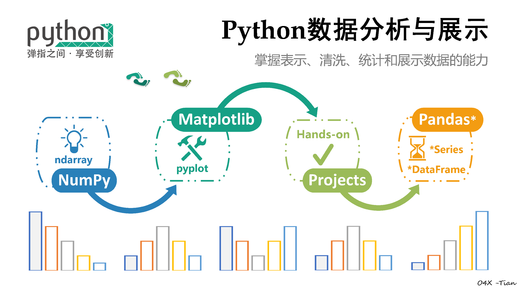
（1）利用Python语言表达N维数据并结合数据特点合理展示数据的技术和方法，

（2）帮助学习者掌握表示、清洗、统计和展示数据的能力。

**（3）运用数据是精准刻画事物、呈现发展规律的主要手段，分析数据展示规律，把思想变得更精细**

（4）培养学习者运用当代最优秀第三方专业资源，快速分析和解决问题的能力。

**实验准备**



**实验内容**

（1）Python第三方库Pandas，强大的专业级数据分析和处理第三方库，介绍并讲解对时间序列数据的基本分析方法以及对缺失数据的基本处理方法。

（2）Python第三方库NumPy，讲解N维数据的表达及科学计算的基本概念和运算方法；

（3）Python第三方库Matplotlib，讲解绘制坐标系、雷达图、等高线等直观展示数据趋势和特点的方法。

**按照下面步骤来练习**