

# 《Python 语言（实验）》课程实验指导书

修订日期： 2019 年 2 月

课程名称 Python 语言（实验）

课程代码

面向专业 社会科学实验班（信息管理）

学时数：48

学分：2

编写者：范昊

实验项目列表：

序号	实验项目名称	学时数
1	实验环境搭建与熟悉	3
2	基本数据类型与运算符的使用	3
3	程序控制结构	3
4	组合数据类型的应用	3
5	函数和代码复用	3
6	基础拓展模块的使用	3
7	文件读写和数据格式化	3
8	函数和代码复用：进阶	3
9	期中综合实验	3
10	正则表达式	3
11	文件和数据格式化：进阶	3
12	Jieba 库使用与词频分析	3
13	文本分析综合实验	3
14	科学计算与可视化	3
15	数据分析综合实验（一）	3
16	数据分析综合实验（二）	3

## 实验一 实验环境搭建与熟悉（3 学时）

### 一、实验目的

#### 1、掌握 Python 开发环境的搭建

Python 是一种结合了解释性、编译性、互动性的面向对象的计算机程序设计语言。最初它被设计用于编写自动化脚本，随着版本的不断更新和语言新功能的添加，越来越多的人用它来开发独立、大型的项目。Python 是一种跨平台的编程语言，能够运行在所有主要的操作系统中。

本实验的目的是以 Windows 系统为例，让学生掌握并实践 python3 以及相关编程软件的安装。

#### 2、掌握 IDLE、Jupyter Notebook 等软件的基本操作

集成开发环境（IDE，Integrated Development Environment）是用于提供程序开发环境的应用程序。IDE 集成了众多专门为软件开发而设计的工具，一般包括代码编辑器、编译器、调试器和图形用户界面等工具，集成了代码编写功能、分析功能、编译功能、调试功能等一体化的开发软件服务套件。

IDLE 是唯一一个 Python 标准发行版中包含的 Python IDE。当系统中安装好 Python 以后，IDLE 就已经自动安装好了，在任何能运行 Python 的环境下用户都能运行 IDLE。。它是采用 Python 的图形界面库 Tkinter 开发的，操作界面较为简单，主要以纯文本输入代码的方式为主。

Jupyter Notebook 一个开源 Web 应用程序，由两个组件构成：网页应用和笔记本（Notebook）文档。网页应用是一种基于浏览器的交互式文档创作平台，提供了编写解释文本、数学公式、交互计算和其他形式富文本（rich text）的诸多功能。

本实验的目的是通过演示和实践 IDLE 和 Jupyter Notebook 的基本操作，让学生掌握上述两个常用 python 集成开发环境的基本操作。

#### 3、熟练使用 IDLE、Jupyter Notebook 进行基本 Python 程序的编写、运行

本实验的目的是通过在 IDLE 和 Jupyter Notebook 上编写和运行 python 程序，让学生熟练使用 IDLE、Jupyter Notebook 进行基本 Python 程序的编写和运行。

### 二、实验要求

#### 1、自行搭建 Python 编程环境（Python 3.x 安装、Jupyter Notebook 软件安装）

2、安装 Jupyter Notebook，在 Jupyter Notebook 中编写程序实现对朋友的问候，同时以 Markdown 方式下编写一段语句。

#### 3、编写 MyFirstPython.py 文件，在终端窗口运行程序，输出结果：

```
Hello world!
```

```
This is my first Python code.
```

#### 4、编写程序完成下列问题：

（1）初三年级 9 个班，每班 45 人，请问初三年级共有多少人？请分别使用交互式运解器器文件和文件的方式运行代码并展示结果。

- (2) 在交互解释器中使用 `print` 语句打印 “I am a student!”。
- (3) 使用 Jupyter Notebook 运行以上两题中出现的代码。
- (4) 请使用 `pip` 命令安装 `matplotlib` (Python 绘图第三方库)。

## 实验二 基本数据类型的使用（3 学时）

### 一、实验目的

#### 1、掌握 Python 语言的基本语法，包括缩进、变量、命名等。

每一种编程语言都有着自身独特的语言规范，对于程序开发者而言，了解并学习这些语言规范以使其编写的程序代码具有可读性和规范性十分重要。Python 语言的语言规范是包含程序格式规范、变量与常量的命名规范等等。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握 Python 语言的基本语法。

#### 2、掌握 Python 的基本数据类型的概念和使用

Python 中包含的基本数据类型主要有数值类型、逻辑值类型和字符类型，其中，数值类型包含整数类型、浮点数类型和复数类型，逻辑值通常使用常量 True 和 False 来表示，字符类型由一系列字符表示，既可以是中文字符，也可以是英文字符。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握 Python 的基本数据类型的概念和使用。

#### 3、掌握 Python 运算符与表达式的基本用法。

Python 语言的运算符包括算术运算符、逻辑运算符、关系运算符、赋值运算符、成员运算符、位运算符、成员运算符以及身份运算符等，能够实现基本的计算功能。表达式则是由数字、运算符、数字分组符号（括号）、自由变量和约束变量等要素构成的，按照既定的、有意义的方式进行排列，并能求得返回值（结果）的组合。表达式是构成程序代码的重要组成部分，能够表达程序的基本计算语句。表达式可以按照连接运算数的运算符进行分类，分成算术表达式、逻辑表达式、关系表达式等。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握 Python 运算符与表达式的基本用法。

#### 4、掌握 Python 字符串类型的基本操作

字符串是一种 Python 数据类型，由一系列字符构成，既可以是中文字符，也可以是英文字符。Python 提供了诸多的字符串操作方法，其基本操作包括字符串的引号使用、合并、复制与转义等。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握 Python 字符串类型的基本操作。

### 二、实验要求

#### 1、创建两个数值变量分别命名为 a、b 并赋值，然后输出变量的值。

2、分别对题 1 中的两个变量进行算术运算（加、减、乘、除、幂运算）、比较运算（大于、小于、等于、大于等于、小于等于）、逻辑运算（与、或、非）、位运算，并使用 print 语句输出结果。

#### 3、分别对题 1 中的两个变量进行复数类型的转换，并使用 print 语句输出结果。

#### 4、创建两个字符串变量并将其合并，两个字符串中间留有空格。

#### 5、移除“128\*6”的引号，并输出计算结果。

6、参考教材 3.4.5 节内容编写格式化输出代码，等待用户输入名字、爱好，根据用户的名字和爱好进行任意显示。例如：xxx 喜欢 xxx。

## 实验三 程序控制结构（3 学时）

### 一、实验目的

#### 1、掌握程序的顺序、分支、循环等控制结构

程序控制结构是指以某种顺序执行的一系列动作，其目的是用于解决某个特定问题而设计的指令流程。理解程序控制结构是学习一门语言的重要组成部分，控制结构规定了程序语句执行的顺序，使其能根据用户的需求来执行指令。无论多复杂的算法均可通过顺序结构、选择结构、循环结构三种基本的控制结构来实现。

顺序结构的程序，就是指程序的语句是按照其出现的先后顺序来执行的程序结构，这是结构化程序中最简单的结构。计算机按照程序的语句顺序逐条执行语句，当一条语句执行完毕，会自动地转到下一条语句开始执行。这就是顺序结构的执行方式。

当程序执行到分支结构的语句时，首先要进行表达式的条件判断，根据条件判断的值选择相应分支中的语句块进行执行，而同时另一分支中的语句块就会被放弃执行。分支结构根据分支的数目可分为单分支、双分支和多分支三种形式，以及嵌套的分支结构。Python 采用 if、else、elif 等语句实现分支结构的程序控制。

循环结构是指对有某些语句或者功能模块反复地执行，构成循环的程序结构。循环结构由条件表达式和循环体构成，根据条件表达式，可以判断程序是继续执行循环体中的语句块还是退出循环。Python 中的循环语句有 while 循环和 for 循环两种。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握程序的分支、循环等控制结构。

#### 2、了解程序的异常处理及用法；

异常是指程序在执行过程中引发的错误事件，该事件会影响程序的正常执行。Python 提供特定的语法来实现异常处理，可以让程序具有更高的容错性，在异常情况下也能够正确地加以处理，给用户提供更加友好的提示，引导程序回到正常运行的状态之下。

本实验的目的是通过编写和运行相关 python 程序，让学生了解程序的异常处理及用法。

### 二、实验要求

1、绘制流程图并编写 Python 程序实现如下操作：求数的阶乘，输入一个正整数，计算并输出该数的阶乘。

2、分别用 for 循环和 while 循环实现 1~100 之间所有偶数的求和。

3、编写如下程序，按用户的输入计算苹果的总价，要求用户输入苹果单价和个数。如果是非数字输入，允许用户重新输入，直到输入正确并计算出结果。

4、编写代码完成一个猜数字游戏：系统随机生成的一个 1 到 100 的数字，每个玩家共有 5 次猜测机会，每次猜测之后，系统会产生一次提示信息，说明猜测的数目太大还是太小，如果猜测正确，则游戏成功并退出；若连续 5 次未猜中，则游戏失败。

5、编写一个小程序，模拟幼儿园老师分苹果，要求每个人至少要分到一个苹果。如果不够，显示出错提示。

## 实验四 组合数据类型的应用（3 学时）

### 一. 实验目的

#### 1、了解 3 类基本组合数据类型。

Python 中常用的组合数据类型有三种：序列类型、集合类型和映射类型。Python 序列是一维元素的向量，在一个序列中元素类型可以相同，也可以不同，序列中各成员元素间由序号引导，通过下标访问序列的特定元素。Python 的内置序列类型：字符串(str)、元组(tuple)和列表(list)等。

集合数据类型往往是由无序的、不重复的元素所组成的数据组合。Python 语言的集合有两种不同的类型，分别是可变集合类型(set)和不可变集合类型(frozenset)。可变集合类型可以添加或删除元素，但其存储的数据元素不能执行哈希操作；而不可变集合类型frozenset的对象是不能添加或删除元素的，但其中存储的数据元素是可以执行哈希操作。

Python 中映射数据类型是由“键-值”数据项构成的组合，并且也是可迭代的。在进行迭代时，映射类型以任意顺序提供其数据项。映射对象键和值是一对多的关系，即“键”是不允许重复的，但“值”可以重复。字典是 Python 中唯一的映射类型，提供了存取数据项及其键、值的方法。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握 python 中 3 类基本组合数据类型。

#### 2、理解列表概念并掌握 Python 中列表的使用。

列表(list)是 Python 语言提供的最常用的序列数据类型，是由一系列任意类型的对象按特定顺序排列而成的。列表用方括号“[]”加以界定，其内的各元素间用逗号分隔。同一列表中可以包含多个不同类型的数据对象，不同的列表元素之间的数据类型也可以不同；任何种类的数据对象，包括数字、字符串、元组、集合甚至其他列表，都可以成为列表对象。列表的常用操作包括修改、删除、插入、修改等。

本实验的目的是通过编写和运行相关 python 程序，让学生理解列表概念并掌握列表的使用。

#### 3、理解字典概念并掌握 Python 中字典的使用。

字典(dict)是映射的体现，通过“键-值”对来进行数据索引的扩展，只有可进行哈希运算的数据对象才可用作字典的键，例如字符串、元组等不可变序列，而列表、集合、字典等可变序列则不能作为字典的键。字典中每个键关联的值实际上是一个对象的引用，可以引用任意类型的对象，包括字符串、元组、列表、字典、集合、函数等，都可以作为字典的值元素。字典的常用操作包括读取、修改、删除等。

本实验的目的是通过编写和运行相关 python 程序，让学生理解字典概念并掌握字典的使用。

### 二、实验要求

1、有 5 名学生 xiaoyun、xiaohong、xiaoteng、xiaoyi 和 xiaoyang，其 QQ 号分别是 88888、555555、11111、12341234 和 1212121，用字典将这些数据组织起来。编程实现以下功能：

用户输入某一个学生的姓名后输出其 QQ 号, 如果输入的姓名不在字典中则输出字符串“Not Found”。

2、编写程序, 求出 1000 以内的所有完数。一个数如果恰好等于它的因子之和, 这个数就称为完数。例如 6 的因子是 1,2,3, 而且  $6=1+2+3$ , 所以 6 就是一个完数。

3、编写程序实现以下操作: 首先生成包含 1000 个随机字符的字符串, 然后统计每个字符的出现次数。

4、应用列表和元组将一下电影按票房由高到低进行排列:

《哪吒之魔童降世》, 票房: 49.34 亿

《疯狂的外星人》, 票房: 21.83 亿

《流浪地球》, 票房: 46.18 亿

《我和我的祖国》, 票房: 29.64 亿

《烈火英雄》, 票房: 16.76 亿

《中国机长》, 票房: 28.46 亿



## 实验五 函数和代码复用（3 学时）

### 一. 实验目的

#### 1、掌握函数的调用方法

函数是一组可以被重复使用的代码的集合。使用函数时，只要按照函数定义的形式向函数传递必需的参数，就可以让函数完成所需的功能。Python 在调用函数的时候，需要使用函数名指定要调用的函数，然后在函数名后的圆括号中给出需要传递给函数参数的值。函数名其实就是指向一个函数对象的引用，也可以把函数名赋给一个变量。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握 python 函数的调用方法。

#### 2、掌握函数的参数传递的方法；

函数的参数传递方式有多种。定义函数的时候，可以把参数的名字和位置确定下来，函数的接口定义就完成了。在 Python 的函数除了正常定义的位置参数外，还可以使用默认参数、命名参数、可变长参数和关键字参数。在 Python 中定义函数时，以上几种参数都可以组合使用。但是请注意，参数定义的顺序必须是：位置参数、默认参数/命名参数、可变参数、关键字参数。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握参数传递的方法。

#### 3、掌握自定义函数的方法。

除 Python 本身提供的可直接调用的内置函数外，用户也可以根据需要灵活地编写自己的函数。在 Python 中，使用 def 自定义一个函数。完整的函数由函数名、参数列表以及函数语句组成。用户自定义函数内容主要涉及到函数的声明、函数的参数、函数的嵌套使用以及变量的作用域等相关内容。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握自定义函数的方法。

### 二、实验要求

1、“水仙花数”是指一个三位数，其各位数字的立方之和的结果等于该数本身。例如：153 是一个“水仙花数”，因为  $153 = 1^3 + 5^3 + 3^3$ 。

问题：编写一个函数，要求输出所有的“水仙花数”。

输入格式示例：无

输出格式示例：水仙花数有：

153, 370, 371, 407

2、回文数是指正读（从左往右）和反读（从右往左）都一样的一类数字。五位回文数指个位与万位相同、十位与千位相同的对称型五位数，如 12321 是回文数。

问题：编写一个函数，输入一个 5 位数，要求判断它是不是回文数，并输出判断结果。

输入格式示例：请输入一个 5 位整数：12321

输出格式示例：12321 是一个回文数

3、编写函数计算  $1^2 - 2^2 + 3^2 - 4^2 + \dots - 98^2 + 99^2$  的值。

4、编写一个函数，使用字典存储学生信息，学生信息包括学号和姓名，并分别根据学生学号升序、学生姓名首字母升序输出学生的信息。

5、编写一个函数 `Cacluat()`，它可以对接收的任意多个数返回一个元组，这个元组的第一个值为所有参数的平均值，第二个值为大于平均值的所有数。

## 实验六 基础扩展模块的使用（3 学时）

### 一、实验目的

#### 1、掌握 datetime 库的使用方法

datetime 模块提供了通过多种方式操作日期和时间的类，在支持日期时间数学运算的同时，更着重于如何能够更有效地解析其字段用于格式化输出和数据操作。datetime 模块包含以下类：表示日期的 date 类、表示时间的 time 类、表示日期和时间的 datetime 类、表示日期或时间间隔的 timedelta 类、表示时区的 tzinfo 类和 timezone 类。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握 datetime 库的使用方法。

#### 2、掌握算数模块的使用方法。

Python 提供了支持各种类型数据进行数学函数运算的模块。例如 math 模块提供了对于实数的 C 语言标准定义的数学函数的访问，cmath 模块支持复数运算，decimal 模块支持十进制定点和浮点运算，fractions 模块支持分数运算。另外，random 模块能够实现各种分布的伪随机数生成器，满足编程的需要。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握算数模块的使用方法。

#### 3、掌握制图模块的使用方法。

Python 中常用的制图模块是 turtle 模块，该模块使用 Tkinter 框架实现基本的图形绘制功能，其原理是想象在绘图区内有一只只会移动的机器海龟，起始位置在 xy 二维平面的(0, 0)点，以海龟走过的路线为轨迹，通过控制海龟的前进方向、前进距离以及对画笔风格等元素，绘制出想要的图形。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握制图模块的使用方法。

### 二、实验要求

1、请使用 datetime 模块求取用户年龄（用户生日由用户输入）并判断用户属于青年人、中年人还是老年人。

注：根据 2017 年联合国世界卫生组织发布的年龄分段，44 岁及以下为青年人，45 岁到 59 岁为中年人，60 岁到 74 岁为年轻老年人，75 岁到 89 岁为老年人，90 岁及以上为长寿老年人。

2、时间模块提供三种时间表示方式，分别是时间戳 timestamp、格式化时间字符串 format string 以及时间元组 struct\_time。其中，时间戳和格式化时间字符串可通过特定函数与时间元组之间进行相互转换，转换条件如图 8-4。

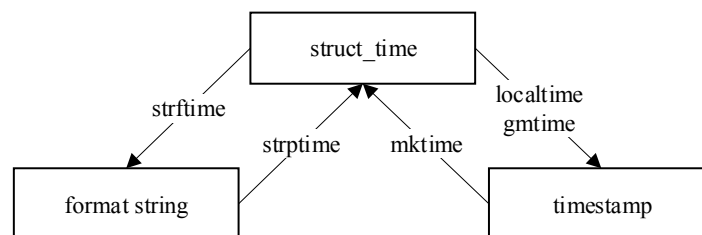
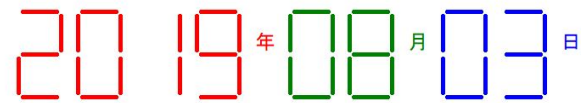


图8-1 时间表示方式转换图

请参考上图编写代码获取系统当前时间戳并转换为时间元组和格式化时间字符串输出，时间字符串格式要求为：xxxx 年 xx 月 xx 日 xx:xx:xx。

3、已知蒙特卡洛方法使用随机数（或更常见的伪随机数）来解决计算问题，可以通过撒点后点的分布得出面积比，进而求出圆周率的近似值。请使用蒙特卡洛方法计算圆周率的近似值，并输出程序执行时间。

4、请编写程序实现当前日期的七段晶体管绘制。效果如图下图。



20 19 年 08 月 03 日

5、获取当前日期时间，并输出当月的日历。

6、将浮点数 3.14 和 decimal 对象 Decimal(3.14)转换为 Fraction 实例，并求该实例的向上取整值和向下取整值、舍入值以及近似估计值。

7、使用 turtle 模块绘制一个红色五角星。

## 实验七 文件读写与数据格式化（3 学时）

### 一、实验目的

#### 1、掌握文件的读写方法以及打开和关闭等基本操作

Python 提供了内置函数 `open()` 来打开文件，使用 `with` 语句或文件对象的 `close()` 方法来关闭文件。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握文件的读写方法以及打开和关闭等基本操作。

#### 2、掌握 txt 文件的读写方法

txt 文件是一种最常见的文件格式，主要存储文本信息，可通过 Python 的文件对象使用相关函数，如 `f.read()`、`f.readline()`、`f.readlines()`、`f.write()` 和 `f.writelines()` 直接进行操作。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握 txt 文件的读写方法。

#### 3、掌握 excel 文件和读写方法

excel 的文件有两种文件后缀形式：xls 与 xlsx，分别通过 Python 的 `xlrd` 模块与 `xlwt` 模块进行读写。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握 excel 文件的读写方法。

#### 4、掌握 csv 文件和读写方法

csv 文件是一种电子表格和数据库中最常见的输入、输出文件格式，将表格数据存储为纯文本，便于进行存储、转换和处理数据。通过 Python 的 `csv` 模块进行读写。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握 csv 文件的读写方法。

### 二、实验要求

1、把给的 txt 文件中句首单词改成小写，超过 10 个单词的句子，18 个单词后面的部分删去，重新存回一个新的 txt 中。给出的 txt 文件中的内容如下：

I don't know what that dream is that you have. I don't care how disappointing it might be as you're working toward that dream. Some of you already know that it's hard. It's not easy. It's hard changing your life. In the process of chasing your dreams, you are going to incur a lot of disappointment, a lot of failure, a lot of pain.

2、创建学生数组，内容为学号、姓名、年龄、性别、成绩，将数组写入 test.xlsx 表格文件中。结果示例如下图所示：

学号	姓名	年龄	性别	成绩
1001	A	11	男	12
100	B	12	女	22
1003	C	13	女	32
1004	D	14	男	52

3、创建 csv 文件，其中数据排列方式为国家,男性,女性，例如 China,30,45，中国对应男性 30 人，女性 45 人。中的国家和所对应的性别统计数据。请提取原 csv 文件中的国家及性别对应的统计数据。

## 实验八 函数和代码复用：进阶（3 学时）

### 一. 实验目的

#### 1、掌握lambda函数的使用

在计算机程序设计中，匿名函数是未绑定到标识符的函数定义。匿名函数通常是传递给高阶函数的参数，或用于构造需要返回函数的高阶函数的结果。

所谓匿名函数，就是指所声明的函数没有函数名，`lambda`表达式是常见的匿名函数定义方式，其主体是一个表达式，而不是一个代码块，其函数体比`def`中定义的语句体要简单很多，故适用于定义小型函数。使用`lambda`声明的函数可以返回一个值，在调用函数时可直接使用该返回值。

本实验的目的是通过编写和运行相关 `python` 程序，让学生掌握 `lambda` 匿名函数的方法。

#### 2、理解递归的概念

在函数内部，可以调用其他函数。如果一个函数在内部调用自身本身，这个函数就是递归函数，即递归函数是一种特殊的函数嵌套。

递归是一种程序设计方法，它的过程主要分为两个阶段：递推和回归。在递推阶段，递归函数在内部调用自己。每一次函数在调用自己之后，又重新开始执行此函数的代码，直到某一级递归程序结束为止；在回归阶段，递归函数从后往前逐级返回的。

递归函数从函数调用的最后一级（也就是递归程序最先结束的那一层）开始逐层返回，一直到返回到函数调用的第一层（也就是产生第一次调用的函数体内）。递归函数逐级返回的顺序与其逐级调用顺序相反。

本实验的目的是通过演示和运行相关 `python` 程序，让学生理解递归的概念。

#### 3、使用递归解决问题

采用递归策略解决问题，通常可以把一个大型复杂的问题层层转化为一个与原问题相似的规模较小的问题来求解，只需少量的程序就可描述出解题过程所需要的多次重复计算，大大地减少了程序的代码量。

递归的能力在于用有限的语句来定义对象的无限集合，因此用递归思想写出的程序往往十分简洁易懂。需要注意的是，在使用递归策略时，函数中必须有一个明确的递归结束的条件，称为递归出口，否则递归程序将无法结束。一般是通过判断语句来作为递归出口，结束递归程序。

本实验的目的是通过编写和运行相关`python`程序，让学生深入理解递归思想，掌握递归策略，能够使用递归函数解决实际问题。

### 二、实验要求

1. 实现 `isPrime()`函数，参数为整数。如果整数是素数，返回 `True`，否则返回 `False`。编写程序并保存为 `isPrimeFun.py`
1. 请思考：如果用户输入的不是整数，是浮点数或者字符串等，程序就会出错。那么，如何进行异常处理，保证程序不出错呢？
2. 调用 `isPrime()`函数并在屏幕输出 3-100 以内的素数，编写程序并保存为 `showPrime.py`。

提交程序文件 `isPrimeFun.py` 和 `showPrime.py`。

3. 分别用递归函数和非递归函数的形式实现：输入两个 1 至 10000 之间的正整数，输出这两个数之间的所有 Fibonacci 数列。提示：Fibonacci 数列为 1,1,2,3,5,8,13,21……
4. 用递归函数将输入的字符串以相反顺序输出。

## 实验九 期中综合实验（3 学时）

### 一、实验目的

1、综合运用 Python 语言数据类型、控制结构、函数及文件等开展实际训练。

前八次实验涵盖了 python 语言数据类型、程序控制结构、函数、模块以及文件读写等内容，本实验的目的是通过编写和运行相关 python 程序，让学生巩固并加强理解已学过的内容。

2、提交综合实验报告（纸质）

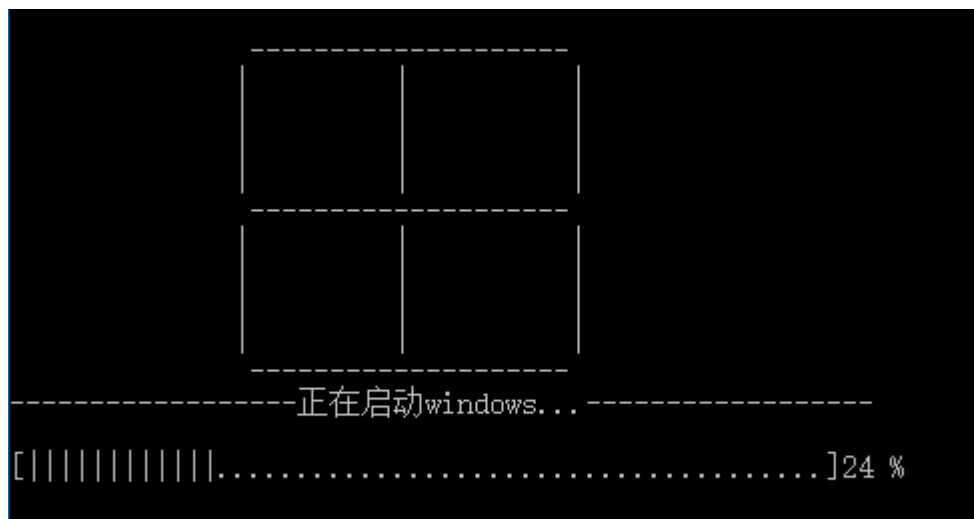
### 二、实验要求

1、编写程序模拟超市交易系统，能实现下列操作。

- (1)输入自己所有的钱。
- (2)展示商品的序号，名称及其价格。
- (3)输入要买商品的序号。
- (4)输入要买商品的数量。
- (5)购物车中显示购买的水果名称及其对应的数量和剩余钱。
- (6)如果序号输入有误就提示用户重新输入。
- (7)如果钱不够了提示用户钱不够，并且退出程序。

2、七段晶体管绘制。请编写程序实现自己的出生日期的七段晶体管绘制。效果图参考实验六第四题。

3、制作一个模拟 windows 启动界面的文本进度条，执行效果如下图所示。



4、猜数字游戏。

（1）在程序中预设一个 0-9 之间的整数，让用户通过键盘输入所猜的数，如果大于预设的数，显示“你猜的数字大于正确答案”；小于预设的数，显示“你猜的数字小于正确答案”，如此循环，直至猜中该数，显示“你猜了 N 次，猜对了，真厉害”，其中 N 是用户输入数字的次数。

（2）异常处理，增加程序健壮性。请用异常处理改造猜数字游戏，使其输入的不是整数(如字母、浮点数等)时，不再出错终止，而是给出“输入内容必须为整数！”的提示，并让用户



重新输入。

5、判断某一年是否为闰年，可以根据“四年闰百年不闰，四百年又闰”来判断。

问题：编写一个 `leap` 函数，要求输入一个年份，判断其是否为闰年，并输出判断结果。

输入格式示例：输入一个年份：2020

输出格式示例：2020 年是闰年

6.汉诺塔问题。

汉诺塔（又称河内塔）问题是源于印度一个古老传说的益智玩具：开天辟地的神博拉码创造世界时做了三根金刚石柱子，在一根柱子上套着 64 片黄金圆盘，最大的圆盘在最底下，其余圆盘从下往上依次按大小顺序排列。神博拉码命令婆罗门把所有圆盘从底下开始按大小顺序重新摆放在另一根柱子上。并且规定，在小圆盘上不能放大圆盘，一次只能移动一个圆盘，可以利用中间的一根棒作为帮助。

设三个柱子分别编号为 A，B，C，圆盘个数为 N，若需将 A 柱上的所有 N 个圆盘移到 C 柱上，请思考：当 n 取不同值时圆盘移动的步骤？

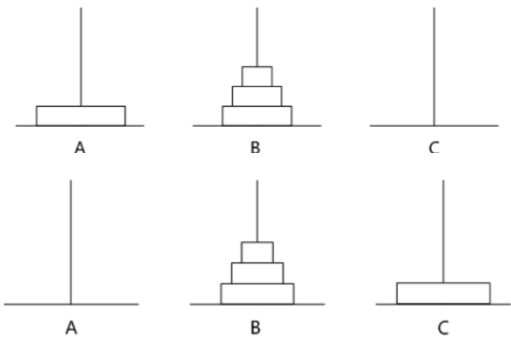
问题：编写一个函数，给定输入 N，A，B，C，输出圆盘移动的步骤。

输入格式示例：共有几个圆盘：4

输出格式示例：移动步骤为：

从 A 到 B

.....



## 实验十 正则表达式（3 学时）

### 一、实验目的

#### 1、掌握字符串类型的函数操作

Python 提供了一系列函数对字符串进行操作，以实现字符串的大小写转换，获取字符串长度，删除字符串中空白，分割字符串，替换字符，转换字符串类型以及字符串的切片与索引等。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握字符串类型的函数操作。

#### 2、掌握正则表达式的使用

正则表达式（Regular Expression, RE）用于处理文件和数据，是一种高级的文本模式匹配方式，为搜索和替代等功能的实现提供了基础。正则表达式是一些由字符和特殊符号组成的字符串，它们描述了这些字符和符号的某种重复方式，能够按照某种预先设定的模式来匹配一个具有相似特征的字符串的集合。Python 通过标准库的 re 模块来进行正则表达式的解释和功能的实现。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握正则表达式的使用。

### 二、实验要求

#### 1、现有语料如下，输出这段英文中所有长度为 4 个字母的单词。

Emma Woodhouse, handsome, clever, and rich, with a comfortable home and happy disposition, seemed to unite some of the best blessings of existence; and had lived nearly twenty-one years in the world with very little to distress or vex her.

#### 2、利用正则表达式，将句子“i am a college student, I am not a businessman.”中拼写错误的“i”替换为“I”；

#### 3、利用正则表达式，将句子“i am a college student, I am not a busInessman.”，其中有单词中间的字母“i”误写为“I”，请编写程序进行纠正。

#### 4、构造正则表达式进行用户名匹配，用户名要求为任意数组和字母的组合（大小写均可）但是不超过 8 个字符。

#### 5、构造正则表达式进行用户密码匹配，用户登录密码必须以字母开头（大小写不限）至少 8 个字符，并且至少包含 1 个大写字母，一个小写字母和是一个数字，可包含特殊符号“.”，“!”，“-”，“\_”；支付密码必须是 5 位的数字密码。

#### 6、构造正则表达式进行固话地址匹配。固话地址由区号加座机号码构成，其中区号为 3 至 4 位，座机号码为 7-8 位。

#### 7、构建正则表达式进行 url 匹配。URL 可以分为两个部分，以常见的 http 协议的 URL 为例，第一部分是协议部分即“http://”；第二部分是域名，域名可以视为是一个以“www.”为开头的，中间为任意数字和字母的组合，最后以“.com”、“.cn”、“.net”等结尾的一个字符串。

#### 8、构建正则表达式进行电子邮箱地址匹配。电子邮箱地址通常由三部分组成，第一部分是用户名，一般由任意数字和字母组成，并且允许使用“-”和“\_”；第二部分是分隔符“@”；第三部分是邮箱服务器的域名如“xxx.com”。

## 实验十一 文件读写与数据格式化：进阶（3 学时）

### 一、实验目的

#### 1、掌握 json 格式与 Python 对象的转换方法

JavaScript 对象标记（JavaScript Object Notation，JSON）是一种常用的轻量级的标准数据交换格式，采用完全独立于编程语言的文本格式来存储和表示数据。JSON 是 JavaScript 对象的字符串表示法，使用文本表示一个 JavaScript 对象的信息，本质是一个字符串。通过 Python 的 json 模块能够实现 Python 对象和 JSON 格式之间的互相转换。另外，json 模块还提供了编码器类 JSONDecoder 和解码器类 JSONEncoder，支持自定义的编码和解码操作。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握 json 格式与 Python 对象的转换方法以及编解码操作。

#### 2、掌握 sql 文件的读写方法

结构化查询语言(Structured Query Language)简称 SQL，是一种特殊目的的编程语言，是一种数据库查询和程序设计语言，用于存取数据以及查询、更新和管理关系数据库系统。SQL 文件是存储在数据库中的文件格式，通常表现为二维表，可通过 Python 的 PyMySQL 模块进行读写。通过该模块可以对 sql 文件进行创建、插入、查询、修改和删除等操作。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握 sql 文件的读写方式。

### 二、实验要求

1、将数组[{'a': 1, 'b': 2, 'c': 3, 'd': 4, 'e': 5}]编码为 json 格式的数据并输出。

2、将字典{'a': 'Runoob', 'b': 7}编码为 json 格式的数据，并进行格式化输出。

注：输出格式为：按字典键的大小排序，设置缩进为 4 字节。

3、将 json 对象{"a":1,"b":2,"c":3,"d":4,"e":5}解码为 Python 格式并输出。

4、使用 dumps 将字典转化为字符串格式并存入 txt 文件当中，如：student = {'Gina': '123456', 'Hellen': '7891', 'Tom': '111111', 'Jerry': '111'}。

5、在 SQL 数据库中创建一个新的二维表，记录某个班级学生的考试成绩，字段包括：学号、姓名、语文、数学、英语、理综以及总分，设置合适的数据类型，并实现下列操作：

- (1) 使用 SQL 的 insert 语句添加至少 10 位学生的成绩记录；
- (2) 使用 fetchall()方法查询数学成绩不及格的学生记录；
- (3) 使用 SQL 的 update 语句修改某一位学生的语文成绩；
- (4) 使用 SQL 的 delete 语句将数学成绩不及格的学生记录删除。

6、在 SQL 数据库中创建学生学籍记录，记载学生的学号、姓名、出生日期、籍贯、院系、专业，并实现下列操作：

- (1) 使用 SQL 的 insert 语句向其中添加多名学生的信息；
- (2) 打印前两行学生的信息。

## 实验十二 Jieba 库使用与词频统计（3 学时）

### 一、实验目的

#### 1、安装第三方库 Jieba 分词

Jieba 分词是目前使用最多的中文分词工具之一，支持精确模式、全模式分词、搜索引擎模式分词和 paddle 模式四种分词模式，同时还支持提取关键词和词形标注等功能。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握 Jieba 库的安装。

#### 2、熟练运用 Jieba 库开展文本数据的分析

jieba 通过调用 jieba.cut 函数实现分词，该函数有四个参数：

- 需要分词的字符串；
- cut\_all 参数用来控制是否采用全模式；
- HMM 参数用来控制是否使用 HMM 模型；
- use\_paddle 参数用来控制是否使用 paddle 模式下的分词模式。

jieba 分词有两种关键词提取模式：基于 TF-IDF 算法的关键词提取和基于 TextRank 算法的关键词提取。基于 TF-IDF 关键词提取模式，通过调用函数 jieba.analyse.extract\_tags(sentence, topK=20, withWeight=False, allowPOS=())来实现；基于 TextRank 算法的关键词提取方法，通过调用函数 jieba.analyse.textrank(sentence, topK=20, withWeight=False, allowPOS=('ns', 'n', 'vn', 'v'))来实现（此方法默认进行词性过滤）。这两个函数都有四个参数，并且意义相同：

- sentence 为待提取的文本；
- topK 为返回 TF-IDF 权重最大的关键词的个数，默认值为 20；
- withWeight 为是否一并返回关键词权重值，默认值为 False 即不返回权重值；
- allowPOS 为仅包括指定词性的词，默认值为空，即不筛选。

jieba 通过 jieba.posseg 函数进行词性标注，采用和 ictclas 兼容的标记法，具体见下表。

ICTCLAS 词性标注集

词性标签	名称	子类
n	名词	nr 人名；ns 地名；nt 机构团体名；nz 其它专名； nl 名词性惯用语；ng 名词性语素
t	时间词	tg 时间词性语素
v	动词	vd 副动词；vn 名动词；vshi 动词“是”；vyou 动词“有”； vf 趋向动词；vx 形式动词；vi 不及物动词（内动词）； vl 动词性惯用语；vg 动词性语素
a	形容词	ad 副形词；an 名形词；ag 形容词性语素；al 形容词性惯用。
b	区别词	bl 区别词性惯用语
r	代词	rr 人称代词；rz 指示代词；ry 疑问代词；rg 代词性语素
m	数词	mq 数量词
q	量词	qv 动量词；qt 时量词
p	介词	pba 介词“把”；pbei 介词“被”
c	连词	cc 并列连词

u	助词	uzhe 着; ule 了, 喽; uguo 过; ude1 的, 底; ude2 地; ude3 得; usuo 所; udeng 等, 等等, 云云; uyy 一样, 一般, 似的, 般; udh 的话; uls 来讲, 来说, 而言, 说来; uzhi 之; ulian 连
x	字符串	xe Email 字符串; xs 微博会话分隔符; xm 表情符合; xu 网址 URL
w	标点符号	wkz 左括号; wky 右括号; wyz 左引号; wyy 右引号; wj 句号; ww 问号; wt 叹号; wd 逗号; wf 分号; wn 顿号; wm 冒号; ws 省略号; wp 破折号; wb 百分号千分号; wh 单位符号
e	叹词	/
d	副词	/
s	所处词	/
f	方位词	/
y	语气词	/
o	拟声词	/
h	前缀	/
k	后缀	/
z	状态词	/

本实验的目的是通过编写和运行相关 python 程序, 让学生掌握如何利用 Jieba 库进行基本的文本分析。

### 3、掌握词频统计的方法

词频是指在一份给定的文本中, 某一个给定的词汇在该文本中出现的次数。词频统计是自然语言处理 and 数据分析中的一种基础手段, 它以简单直观的数字形式帮助读者了解文本中的重要信息, 以便获得信息。

本实验的目的是通过编写和运行相关 python 程序, 让学生掌握词频统计的方法。

## 二、实验要求

- 1、分别利用 Jieba 分词提供的四种分词模式对句子“自然语言处理是研究人与计算机之间用自然语言进行有效通信的各种理论和方法。”进行分词, 并对比分词结果。
- 2、利用 Jieba 分词对题 1 中的句子进行词性标注。
- 3、对下面一段话进行词频统计, 然后分别利用 Jieba 分词提供的两种关键词提取方式进行关键词提取:

武汉大学溯源于 1893 年清末湖广总督张之洞奏请清政府创办的自强学堂, 历经传承演变, 1928 年定名为国立武汉大学, 是近代中国第一批国立大学。1949 年新中国成立, 学校更名为武汉大学。现行武汉大学校徽为 1993 年庆祝百年校庆时设计的图案。校徽图式为圆形, 上方为武汉大学英文校名, 呈弧形, 表达学校国际化办学理念和成为国际一流大学的奋斗目标与价值追求; 中居学校老图书馆造形, 表达学校独有地标特征和文化标志; 中间下书阿拉伯数字“1893”, 表明学校建校年代; 下方为中文汉字毛体校名。1993 年, 在广泛征求各方面意见的基础上, 经校务委员会审议, 武汉大学新校训定为: 自强弘毅求是拓新。“自强”语出《周易》“天行健、君子以自强不息”。意为自尊自重, 不断自力图强, 奋发向上。自强是中华民族的传统美德, 成就事业当以此为训。我校最早前身为“自强学堂”, 其名也

取此意。“弘毅”出自《论语》“士不可以不弘毅，任重而道远”一语。意谓抱负远大，坚强刚毅。我校 30 年代校训“明诚弘毅”就含此一词。用“自强”、“弘毅”，既概括了上述含义，又体现了我校的历史纵深与校风延续。“求是”即为博学求知，努力探索规律，追求真理。语出《汉书》“修学好古，实事求是”。“拓新”，意为开拓、创新，不断进取。概言之，我校新校训的整体含义是：继承和发扬中华民族自强不息的伟大精神，树立为国家的繁荣昌盛刻苦学习、积极奉献的伟大志向，以坚毅刚强的品格和科学严谨的治学态度，努力探求事物发展的客观规律，开创新局面，取得新成绩，办好社会主义的武汉大学，不断为国家作出新贡献。

## 实验十三 文本分析综合实验（3 学时）

### 一、实验目的

#### 1、掌握文本预处理的基本流程

一般而言，想要执行不同的操作并分析文本，首先需要将文本数据处理和解析为更干净和更容易解读的格式。通常，文本语料和原始文本的数据格式是非规范的，文本预处理就是使用各种技术将原始文本转换为定义良好的语言成分序列。具体分为工具导入、数据读入、数据清洗、去停用词等操作。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握文本预处理的基本流程。

#### 2、了解词云的基本构建流程并掌握词云的构建方法

词云是一种文本分析的可视化方式，能够帮助读者准确快速地筛选出重要的文本信息，进行阅读前的筛选。词云是对文本中出现频率较高的关键词在视觉上的突出呈现，形成关键词的渲染形成类似云层一样的图片，从而过滤掉大量的无用信息，使读者能一眼就领略到文本主要表达的意思。在词云中，通常是不同的词组采用不同的颜色表示，不同词频的词组采用不同的字号大小表示。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握文本预处理的基本流程。

#### 3、提交文本分析综合实验报告（纸质）

### 二、实验要求

#### 1、以《红楼梦》为例，进行文本预处理操作。

##### （1）工具导入

在文本预处理阶段，主要使用的工具是正则表达式（re）和 jieba 工具。首先使用 pip 命令安装工具包，再用 import 语句导入到 Python 中。

```
import re
import jieba
```

##### （2）数据读入

数据存在的形式多种多样，有 csv、Excel、txt 等文件格式，Python 提供了内置函数 open() 可以打开这些格式的文件，基本语法格式为：

```
file object = open(file_name[, access_mode][, buffering])
```

其中，file\_name 为要访问的文件名称，access\_mode 为打开文件的模式，如只读、写入、读写等等，默认文件访问模式为只读(r)，buffering 为访问文件时寄存区的缓冲大小。

文件对象方法 read() 则从一个打开的文件中读取字符串。需要注意的是，Python 字符串可以是二进制数据，而不仅仅是文字，故特殊符号、数字等也会被读取。read() 方法的基本语法格式为：

```
fileObject.read([size])
```

其中 size 为被读取的字节个数。该方法是从文件的开头开始读入，若没有指定 size 参数，将会返回整个文件。

读取《红楼梦》的代码如下：

```
# 读入数据文件

content = open('hlm.txt').read()

content[:99]      #显示部分数据内容
```

### (3) 数据清理

读取数据之后，首先对数据进行清理。在文本分析中，数据清理是指处理文本中包含的大量无关和不必要的标识和字符，例如空格、特殊符号、标点符号等。文本数据的清理可以通过正则表达式操作，使用 sub 算法删除换行符、空白和特殊字符等。

对《红楼梦》进行数据清理，其中每个特殊字符用空字符串来替换：

```
#数据清理

content = re.sub(r'\n+', '', content) #删除换行符

content = re.sub(r' +', '', content) #删除空白

content = re.sub(r'\W+', ' ', content) #空白替换符号

content[:99]      #显示部分文本内容
```

数据清理后，就可以对干净的文本进行分词了。

```
#分词

seg_list = list(jieba.cut(content))

print("分词结果：\n", "/".join(seg_list[:99]))      #显示部分分词结果
```

### (4) 停用词导入

通常经过分词后文本中仍存在一些无意义的词汇，如“此”“也”等，这些词汇对文本分析任务并无任何益处，甚至还会干扰分析结果。这些词也称为停用词。停用词是指没有意义或只有极小意义的词汇，主要包括数字、数学字符、标点符号及使用频率特高的单汉字等，例如“你”“我”“啊”“哦”。通常在文本预处理过程中将它们从文本中删除，使文本保留具有最大意义及语境的词汇。停用词都是人工输入、非自动化生成的，生成后的停用词会形成一个停用词表。但是，不同的领域有不同的停用词，目前并没有普遍或已穷尽的停用词表，每个领域可能都有一系列独有的停用词。

使用 read() 语句读取部分所给的停用词：

```
#加载停用词表

stopwords = open('stopwords.txt').read() #长字符串

stopwords = stopwords.split('\n')      #字符串按'\n'分割，构建列表类型

print("停用词：\n", ", ".join(stopwords[:20]))      #显示部分停用词，第一个为空格
```

利用词表删除《红楼梦》中的停用词：

```
#去停用词

final_content = []
```



```

for seg in seg_list:

    if seg not in stopwords:

        final_content.append(seg)

print("分词结果: \n", "/" .join(final_content[:99]))    #显示部分处理结果

```

2、以 2018 年两会《政府工作报告（全文）》为数据源，构建词云。

### （1）准备工作

在进行本节的词云构建项目实践之前，需要进行软件工具包和文本数据的准备。

构建词云的数据来源必须是纯文本数据，如果读者所用的文本数据的是 PDF 或 word 等格式的文件，必须将需要分析的文本内容转换成纯文本的格式进行存储。本实验采用的数据源是 2018 年两会《政府工作报告（全文）》

（下载地址：[http://www.xinhuanet.com/politics/2018lh/2018-03/22/c\\_1122575588.htm](http://www.xinhuanet.com/politics/2018lh/2018-03/22/c_1122575588.htm)）。

随后需要在 Python 环境下安装 jieba、WordCloud、numpy、re 等第三方软件包并在数据预处理前导入；

```

import jieba
from WordCloud import WordCloud, ImageColorGenerator
import matplotlib.pyplot as plt
from imageio import imread
from collections import Counter
import numpy as np
import re

```

### （2）数据预处理

根据题 1 介绍的方法，对《政府工作报告（全文）》进行文本预处理。首先读取文本数据：

```

# 读入数据文件

content = open('gzb2018.txt').read()

content[:99]    #显示部分数据内容

```

然后对文本数据进行清理，这里清除的是原始文本中大量的换行符、空格和特殊符号：

```

#数据清理

content = re.sub(r'\n+', '', content)

content = re.sub(r' +', '', content)

content = re.sub(r'\W+', ' ', content)

content[:99]    #显示部分文本内容

```

接着加载停用词表进行分词，采用 jieba 分词中的精确模式，代码如下：

```

#加载停用词表

```

```

stopwords = open('stopwords.txt').read() #长字符串

stopwords = stopwords.split('\n')      #字符串按'\n'分割，构建列表类型

#去停用词

final_content = []

for seg in seg_list:

    if seg not in stopwords:

        final_content.append(seg)

print("分词结果：\n", "/".join(final_content[:99]))    #显示部分处理结果

```

在构建词云之前，还要对文本进行词频统计，以便对高频词汇进行可视化展示，词频统计代码如下：

```

#使用 counter 做词频统计，选取出现频率前 500 的词汇

counting_words = Counter(final_content)

print(str(counting_words))

common_words = counting_words.most_common(500)

```

### （3）词云生成

利用 WordCloud 制作词云时，一般经过三个步骤：一是使用 WordCloud.WorldCloud 函数来设置词云对象的有关参数（或者说是属性）；二是利用 WordCloud.generate(text) 函数或 WordCloud.generate\_from\_frequencies(words) 函数生成词云，前者是根据文本生成词云，后者是根据词频生成词云；三是利用 WordCloud.to\_file(file\_name) 函数将词云输出到文件进行保存。

对于 WordCloud 库来说，每个词云是一个 WordCloud 对象，通过对其 20 多个参数进行配置可以设置词云。其中，常用参数说明见下表

WordCloud 常用参数

参数	说明
font_path	字体路径，制作中文词云时必须指定字体文件，否则不能正常显示
width	画布的宽度，默认为 400
height	画布的高度，默认为 200
mask	指定遮罩图（即背景图片、词云的形状图）
contour_width	遮罩轮廓宽度，默认为 0。如果 mask 不为 None 且轮廓宽度大于 0，则绘制遮罩轮廓
contour_color	遮罩轮廓颜色，默认为 “black”
scale	放大画布的比例，默认为 1
max_font_size	最大字体大小，默认为 None，表示使用图像的高度
min_font_size	最小字体大小，默认为 4

font_step	字体的步长，默认为 1
max_words	词云中词组的最大个数，默认为 200
stopwords	设置需要屏蔽的词（即停用词）。如果为空，则使用内置的停用词
background_color	词云图像的背景色，默认为 “black”
relative_scaling	相对词频对字体大小的重要性，值为 0 时，仅考虑词组排名，值为 1 时，频繁出现的词组的大小为 2 倍。默认为 0.5
min_word_length	一个词组必须包含的最小单词数，默认为 0

由于 WordCloud 中默认的字体是英文的，不包含中文编码，因此在绘制中文词云时需要准备一个中文字体文件。配置 WordCloud 中的参数，其中背景色设为白色，词云形状为读入的中国边界图，字体为黑体。

```
# 读入图片，配置词云背景
background_pic = imread('China.jpg')
# 配置词云参数
wc = WordCloud(
    background_color = 'white',
    mask = background_pic,
    font_path = 'simhei.ttf',
    max_words=2000,      # 设置最大现实的字数
    max_font_size=150,  # 设置字体最大值
    margin=1,           # 设置词间间距
    random_state=30,    # 设置有多少种随机生成状态，即有多少种配色方案
    scale = 1           # 按照比例进行放大画布
)
```

接下来把选取的出现频率前 500 的词汇以字典的形式通过 generate\_from\_frequencies() 生成词云，该函数需要指定每个词汇和它对应的频率组成的字典，生成代码如下：

```
wc.generate_from_frequencies(dict(common_words)) # 从字典生成词云
wc.to_file("myWordCloud.png")
```

wc.to\_file() 函数将生成的词云存储到 png 文件里，若要展示词云，可以采用可视化工具 matplotlib，代码如下：

```
%matplotlib inline
wc_pic = imread('myWordCloud.png')
plt.figure(figsize=(15,11))
plt.imshow(wc_pic) # 显示词云
plt.axis('off') # 关闭坐标轴
plt.show()
```

## 实验十四 科学计算与可视化（3 学时）

### 一、实验目的

#### 1、熟练运用 numpy 等科学计算库进行矩阵分析与数值运算

Numpy 是 Python 中一个常用的第三方库，用于科学计算，支持高维数组与矩阵运算。同时，Numpy 可以为用户提供多维数组对象、各种派生对象（如掩码数组和矩阵），以及用于数组快速操作的各种 API，包括数学、逻辑、形状操作、排序、选择、输入输出、离散傅立叶变换、基本线性代数、基本统计运算和随机模拟等。另外，Numpy 不仅是一个常用的、可独立调用的科学计算库，也是许多其他第三方库（例如 Scipy 和 Pandas）的基础库。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握 numpy 等科学计算库进行矩阵分析与数值运算。

#### 2、熟练运用 matplotlib 库进行数据可视化与图形绘制

Matplotlib 是 Python 最基本的二维绘图库，也可进行简单的三维绘图，它能在各种交互式环境中将数据绘制各类图形，达到出版打印的质量级别。在使用 Matplotlib 绘制图形之前，一般会需要进行较为复杂的数据分析和数据处理过程，因此数据分析处理包 NumPy 常与 Matplotlib 共同出现与使用。Matplotlib 可以使用 NumPy 进行数组运算，并调用一系列相关的 Python 库来实现与硬件的交互。Python 中有许多数据分析和数据处理的库，都是通过调用 Matplotlib 的绘图语句来实现数据可视化的。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握运用 matplotlib 库进行数据可视化与图形绘制。

### 二、实验要求

1、使用 numpy 中的 arange 函数来创建三个包含 1~10 的整数的 numpy 数组，使三个数组的形状分别为 10\*1、2\*5、5\*2，并对生成的数组做 exp、exp2、sqrt、sin、log 函数运算；

2、创建两个 numpy 数组，其中 arr1 中存储学生姓名，arr2 中存储学生数学、语文、英语的成绩，使用索引输出第二个学生的英语成绩和第四个学生三门课程的平均分。

3、从二维数组创建一个学生 DataFrame，并为其加上索引和列标，输出年龄大于 18 的学生信息。

4、编写程序绘制下列数学表达式的图像：

（1）余弦三角函数  $y = \cos(2\pi x)$  的图像。

（2）函数  $f(x) = \sin^2(x - 2)e^{-x^2}$  的图像。

（3）多项式  $f(x) = 4x^5 - 10x^3 + 7x + x^{-2} + 10$  及其导函数的图像。

5、请下载 Iris 数据集，并使用 matplotlib 对其作可视化探索，具体包括：

（1）程序绘制纵轴表示 sepal length in cm、sepal width in cm、petal length in cm、petal width in cm 的折线图，并己不同颜色、不同线型区分；

（2）程序将数据集上的每个特征属性取值分成 6 个区间，并绘制其对应的柱状图，其中柱状图横轴表示区间编号，纵轴表示每个区间对应的样本数量（提示：区间划分时注意不应超过每个特征属性的最大值与最小值）；

(3)程序绘制以 sepal length in cm 为 x 轴、sepal width in cm 为 y 轴的散点图和以 petal length in cm 为 x 轴、petal width in cm 为 y 轴的散点图，并在图中用不同颜色、不同散点类型区分出鸢尾花类别（Iris Setosa—红色 ‘o’、Iris Versicolour—蓝色 ‘\*’、Iris Virginica—绿色 ‘+’）。

注：数据集下载地址：<https://archive.ics.uci.edu/ml/datasets/Iris>

## 实验十五 数据分析综合实验（一）（3 学时）

### 一、实验目的

#### 1、了解数据分析的基本流程

一般而言，数据分析包含四个常规流程，分别是数据清理、数据转换、数据分析和数据可视化。原始的海量数据通常存在大量不完整、不一致、有异常的数据，可能影响数据分析的效率，并且导致分析结果的偏差，所以进行数据分析工作前应对数据进行清理。具体操作包括缺失值处理、异常值处理以及重复值处理；数据转换主要是对数据进行规范化处理，将数据转换为适当的形式，以适用于数据分析任务的需要。如数据规范化和连续数据离散化等；在数据探索阶段主要通过绘制图表、计算某些特征量等手段进行数据的特征分析。分布分析能揭示数据的分布特征和分布类型；数据可视化能够加深对数据的认识。可视化方法包括：条形图和饼图、箱线图、气泡图、条形图、核密度估计图、网络图、雷达图、散点图、树状图等。

本实验的目的是通过演示和运行相关 python 程序，让学生了解数据分析的基本流程。

#### 2、掌握数据清理和数据转换的方法

数据清理通常包括缺失值处理、异常值处理和重复值处理，由于异常值处理涉及数据挖掘中的离群点检测问题，本实验仅进行缺失值处理和重复值处理。数据转换通常包括规范化和连续属性离散化，另外，可以把无关的属性从数据集中删除，便于后续分析。

本实验的目的是通过编写和运行相关 python 程序，让学生掌握数据清理和数据转换的方法。

### 二、实验要求

1、请下载 Adult 数据集（数据集下载地址：<https://archive.ics.uci.edu/ml/datasets/Adult>），完成数据清理和数据转换操作。

#### （1）数据说明

此次试验数据集名称为 Adult，来源为 UCI dataset。数据是从美国人口普查局数据库中提取出来的，网址为 <https://archive.ics.uci.edu/ml/datasets/Adult>。UCI dataset 对该数据集的描述如下图：

Data Set Characteristics:	Multivariate	Number of Instances:	48842	Area:	Social
Attribute Characteristics:	Categorical, Integer	Number of Attributes:	14	Date Donated	1996-05-01
Associated Tasks:	Classification	Missing Values?	Yes	Number of Web Hits:	1156747

Adult 数据集描述

该训练数据集包含 15 个字段（14 个描述字段及 1 个预测字段），32561 个元组。字段名称及说明如下：

1) age（年龄）：取连续值；

2) workclass（工作类别）：类别变量，取值范围为 { Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked }；

- 3) `fnlwgt` (最终权重法值): 取连续值;
- 4) `education` (学历): 类别变量, 取值范围为{ Bachelors , Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool};
- 5) `education-num` (受教育年份): 取连续值;
- 6) `marital-status` (婚姻状况): 类别变量, 取值范围为{ Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse};
- 7) `occupation` (职业): 类别变量, 取值范围为{Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces};
- 8) `relationship` (社会关系): 类别变量, 取值范围为{ Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried};
- 9) `race` (种族): 类别变量, 取值范围为{White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black};
- 10) `sex` (性别): 类别变量, 取值范围为{Female, Male};
- 11) `capital-gain` (资本收入): 取连续值;
- 12) `capital-loss` (资本损失): 取连续值;
- 13) `hours-per-week` (每周工作小时数): 取连续值;
- 14) `native-country` (祖国): 类别变量, 取值范围为{United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinidad&Tobago, Peru, Hong, Holand-Netherlands};
- 15) `income` (年收入): 类别变量, 取值范围为{>50K, <=50K}。

利用该数据集进行数据分析的目的是在通过数据预处理(包括缺失值、重复值处理和数据规范化等)后, 统计变量的集中趋势、离中趋势, 并且发现变量间的相关关系, 最终实现对 `income` 变量取值的预测。

## (2) 数据清理

数据清理通常包括缺失值处理、异常值处理和重复值处理, 由于异常值处理涉及数据挖掘中的离群点检测问题, 在本实验中不进行操作。

### 1) 缺失值处理

由于 `Adult` 训练数据集本身已经对缺失值进行了处理, 将所有缺失的数据表示为'?', 因此检查缺失值时应设置'?'为缺失值, 再进行后续处理。

① 利用 pandas 的 isnull()方法检查缺失值

代码如下：

```
import pandas as pd
raw_data = pd.read_csv('Adult.csv',na_values='?') #设定'?'为缺失值
raw_data.isnull().any() #检查含有缺失值的列
```

输出如下：

```
age                False
workclass           True
fnlwgt             False
education           False
education-num       False
marital-status      False
occupation          True
relationship        False
race               False
sex               False
capital-gain        False
capital-loss        False
hours-per-week      False
native-country      True
income             False
dtype: bool
```

由输出结果可知，workclass（工作类别）、occupation（职业）、native\_country（祖国）三列存在缺失值。为了决定采用何种方法处理缺失值，检查数据集中含有缺失值的记录数量，代码如下：

```
raw_data[raw_data.isnull().values==True]
```

输出见下图

	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	sex	capital-gain	capital-loss	hours-per-week	native-country	income	
	14	40	Private	121772	Assoc-voc	11	Married-civ-spouse	Craft-repair	Husband	Asian-Pac-Islander	Male	0	0	40	NaN	>50K
	27	54	NaN	180211	Some-college	10	Married-civ-spouse	NaN	Husband	Asian-Pac-Islander	Male	0	0	60	South	>50K
	27	54	NaN	180211	Some-college	10	Married-civ-spouse	NaN	Husband	Asian-Pac-Islander	Male	0	0	60	South	>50K
	38	31	Private	84154	Some-college	10	Married-civ-spouse	Sales	Husband	White	Male	0	0	38	NaN	>50K
	51	18	Private	226956	HS-grad	9	Never-married	Other-service	Own-child	White	Female	0	0	30	NaN	<=50K
	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
	32539	71	NaN	287372	Doctorate	16	Married-civ-spouse	NaN	Husband	White	Male	0	0	10	United-States	>50K
	32541	41	NaN	202822	HS-grad	9	Separated	NaN	Not-in-family	Black	Female	0	0	32	United-States	<=50K
	32541	41	NaN	202822	HS-grad	9	Separated	NaN	Not-in-family	Black	Female	0	0	32	United-States	<=50K
	32542	72	NaN	129912	HS-grad	9	Married-civ-spouse	NaN	Husband	White	Male	0	0	25	United-States	<=50K
	32542	72	NaN	129912	HS-grad	9	Married-civ-spouse	NaN	Husband	White	Male	0	0	25	United-States	<=50K
4262 rows × 15 columns																

4262 rows × 15 columns

缺失记录输出



由输出结果可知，数据集中共有 4262 条含有缺失值的记录，占有记录数量的 13%，若直接删除含有缺失值的记录会损失很多信息，且因为缺失列都为分类变量，在本例中采用众数进行缺失值填充是一种比较合适的方法。

## ② 利用 pandas 的 fillna()方法填充缺失值

代码如下：

```
fill_na = lambda col:col.fillna(col.mode()[0]) #定义 fill_na 函数，用众数填充缺失值
fill_data = raw_data.apply(fill_na, axis=0) #将填充后的数据赋给 fill_data
fill_data.isnull().any() #检查是否填充成功
```

输出如下：

```
age                False
workclass           False
fnlwgt             False
education           False
education-num       False
marital-status     False
occupation         False
relationship        False
race               False
sex                False
capital-gain        False
capital-loss        False
hours-per-week     False
native-country     False
income             False
dtype: bool
```

由输出结果可知，填充后的数据集 fill\_data 中已不存在缺失值。

## 2) 重复值处理

通过 pandas 的 duplicated()方法判断数据集中是否存在重复值。

代码如下：

```
isDuplicated=fill_data.duplicated() #判断重复数据记录
print(isDuplicated)
```

输出如下：

```
0      False
1      False
2      False
3      False
4      False
...
32556  False
32557  False
```

```
32558    False
32559    False
32560    False
Length: 32561, dtype: bool
```

由输出结果可知，数据集中不存在重复记录，因此不用进行重复值处理。

### （3）数据转换

数据转换通常包括规范化和连续属性离散化，另外，可以把无关的属性从数据集中删除，便于后续分析。

在本例中，`fnlwgt`（最终权重法值）属性可删除，`age` 可划分为等宽区间。删除 `fnlwgt` 属性列及 `age` 属性离散化代码如下：

```
data = fill_data.drop(['fnlwgt'],axis=1) #删除 fnlwgt 属性列
ages = data['age'].copy() #提取 age 列
print('min_age:{}'.format(ages.min())) #输出 age 属性的最小值
print('max_age:{}'.format(ages.max())) #输出 age 属性的最大值，据此得出区间端点值
```

输出如下：

```
min_age:17
max_age:90
```

由输出结果可知，`age` 属性最小值为 17，最大值为 90，因此将范围划分为(25, 35]、(35, 45]、(45,55]、(55,65]、(65, 75]、(75,85]、(85,95]。

划分等宽区间代码如下：

```
bins = [25,35,45,55,65,75,85,95]
_ages = pd.cut(ages, bins, right=True) #right=False 表示区间是左闭右开的
#对不同区间的数进行统计
_ages.value_counts()
```

输出如下：

```
(25, 35]    8514
(35, 45]    8009
(45, 55]    5538
(55, 65]    2931
(65, 75]     917
(75, 85]     193
(85, 95]      48
```

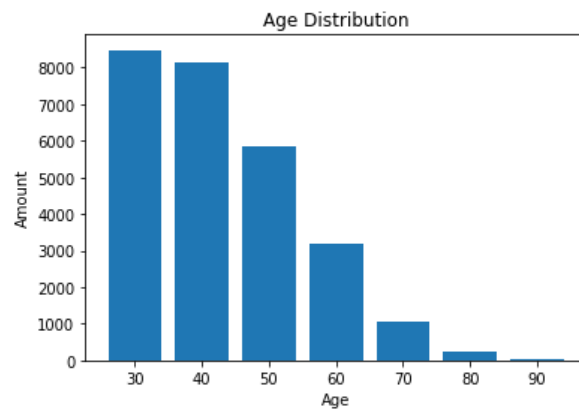
```
Name: age, dtype: int64
```

为了更直观地展示每一个区间内的记录数量，利用 `matplotlib.pyplot` 库绘制年龄分布的条形图。

代码如下：

```
import matplotlib.pyplot as plt #导入matplotlib.pyplot库, 别名为 plt
bins = [25,35,45,55,65,75,85,95] #设置分组端点
plt.hist(ages,bins,histtype='bar',rwidth=0.8) #绘制条形图
plt.xlabel('Age') #x轴命名为'Age'
plt.ylabel('Amount') #y轴命名为'Amount'
plt.title('Age Distribution') #设置图形标题为'Age Distribution'
plt.show() #展示图形
```

输出结果如下图所示。



年龄分布条形图

## 实验十六 数据分析综合实验（二）（3 学时）

### 一、实验目的

1、熟练的运用所学相关模块开展数据分析及可视化等综合项目。

数据分析的目的在于分析变量的取值分布以及变量之间的关系,包括分析定量变量取值的集中趋势和离中趋势、定性变量的取值分布以及变量间的相关关系,最终实现对 salary 变量值的预测。

本次实验的目的是通过进行分布分析的实践,让学生熟练的运用所学相关模块开展数据分析及可视化等综合项目。

2、提交数据分析综合实验报告（纸质）

### 二、实验要求

分布分析实践:

Pandas 库提供了很多函数用于统计变量特征,本例中通过 describe()方法给出定量数据的基本描述(包括基本统计量如均值、标准差等),对于定性数据,通过绘制条形图或饼图展示其分布情况。

代码如下:

```
import pandas as pd
import matplotlib.pyplot as plt  #导入 matplotlib.pyplot 库用于绘图
import collections  #导入 collections 库用于定性变量取值计数
#数据预处理
raw_data = pd.read_csv('Adult.csv',na_values=' ?')  #设定 ' ?'为缺失值
fill_na = lambda col:col.fillna(col.mode()[0])  #定义 fill_na 函数,用众数填充缺失值
fill_data = raw_data.apply(fill_na, axis = 0)  #将填充后的数据赋给 fill_data
data = fill_data.drop(['fnlwgt'],axis=1)  #去除 'fnlwgt' 属性列
#绘制饼图
from collections import Counter
items = data['relationship']  #提取需绘制饼图的数据列
count = Counter(items)  #返回该列取值的计数字典
print(count)  #输出计数字典
countSort = collections.OrderedDict()  #定义有序字典 countSort
countSort = dict(count.most_common())  #将 count 转换为有序字典 countSort
#设置饼图百分比
sizes = list(countSort.values())
for i in range(len(sizes)):
    sizes[i] = round(sizes[i]/32561 * 100,2)
labels = list(countSort.keys())  #设置饼图标签
plt.pie(sizes,labels=labels,autopct='%.2f',counterclock=False,startangle = 90)
```

```

#绘制饼图
#绘制条形图

from collections import Counter

items = data['education-num'] #提取需绘制直方图的数据列
count = Counter(items) #返回该列取值的计数字典
print(count) #输出计数字典

countSort = collections.OrderedDict()
countSort = dict(count.most_common())
print(countSort.values())

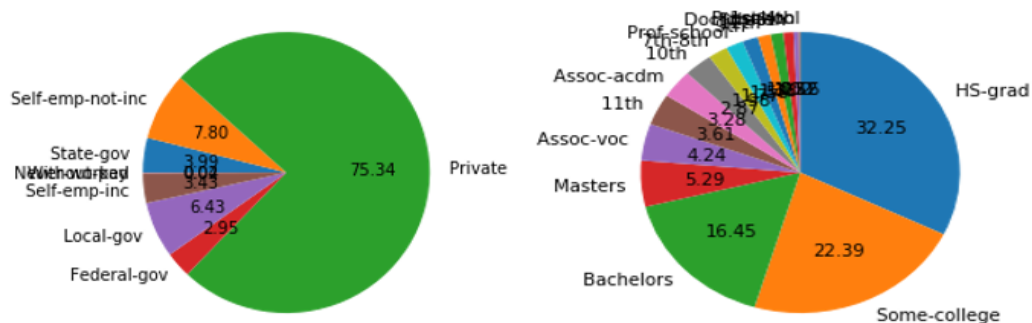
plt.bar(countSort.keys(),countSort.values()) #绘制条形图
plt.xlabel('education-num') #设置横轴标签
plt.ylabel('account') #设置纵轴标签
plt.show()

#统计分析
items = data['hours-per-week'] #提取需分析统计量的数据列
items.describe()

```

输出结果按变量进行分析，如下。

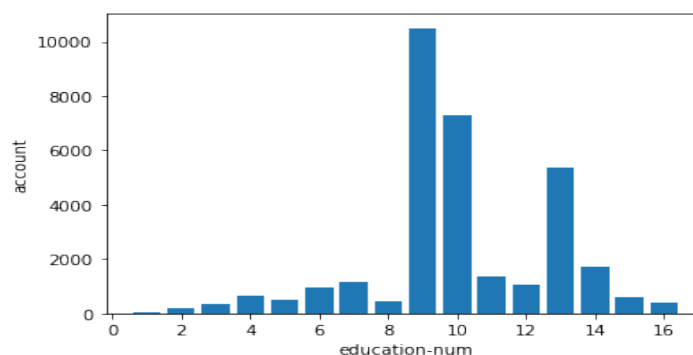
Workclass（工作类别）、Education（学历）分布情况分别为：



工作类别分布图

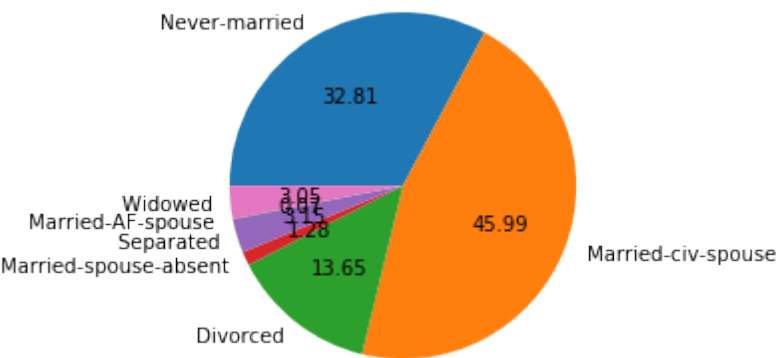
学历分布图

Education-num（受教育年限）分布情况：



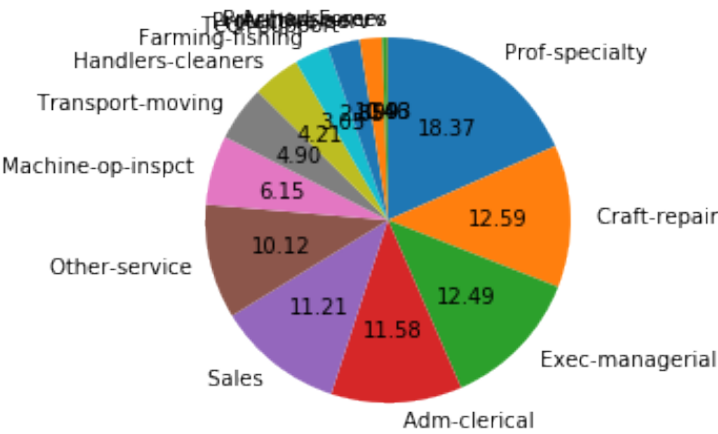
受教育年限分布图

Marital-status（婚姻状况）分布情况：



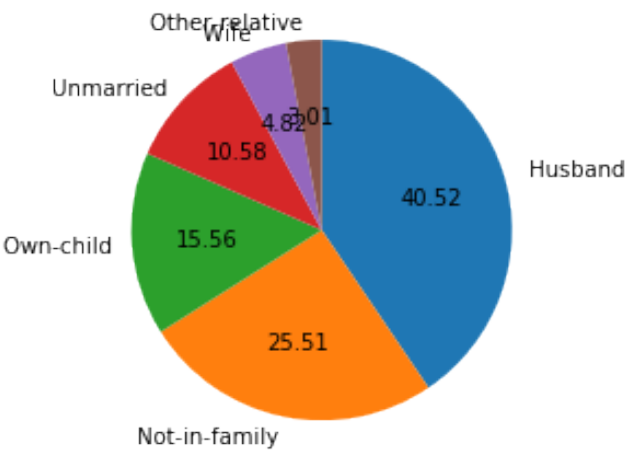
婚姻状况分布图

Occupation（职业）分布情况：



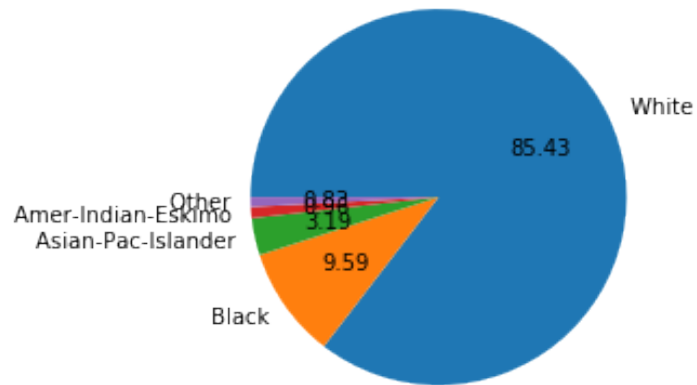
职业分布图

Relationship（社会关系）分布情况：



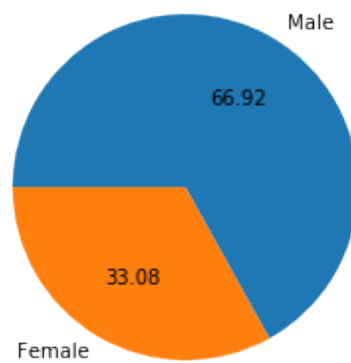
社会关系分布图

Race（人种）分布情况：



人种分布图

Sex（性别）分布情况：



性别分布图

Capital-gain（资本收入）统计量分析结果如下：

```
count    32561.000000
mean      1077.648844
std       7385.292085
min        0.000000
25%        0.000000
50%        0.000000
75%        0.000000
max      99999.000000
```

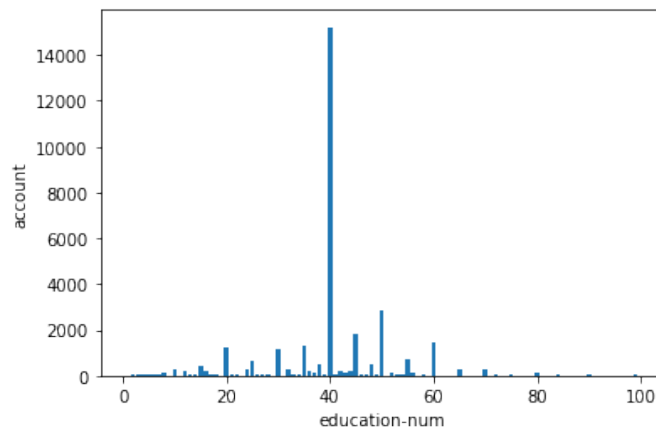
Name: capital-gain, dtype: float64

Capital-loss（资本损失）统计量分析结果如下：

```
count    32561.000000
mean       87.303830
std       402.960219
min        0.000000
```

```
25%      0.000000
50%      0.000000
75%      0.000000
max      4356.000000
Name: capital-loss, dtype: float64
```

Hours-per-week（每周工作小时数）分布情况：



每周工作小时数分布图

```
count      32561.000000
mean        40.437456
std         12.347429
min          1.000000
25%         40.000000
50%         40.000000
75%         45.000000
max         99.000000
Name: hours-per-week, dtype: float64
```

Native-country（祖国）取值计数如下：

```
Counter({' United-States': 29753, ' Mexico': 643, ' Philippines': 198, ' Germany':
137, ' Canada': 121, ' Puerto-Rico': 114, ' El-Salvador': 106, ' India': 100, ' Cuba':
95, ' England': 90, ' Jamaica': 81, ' South': 80, ' China': 75, ' Italy': 73, '
Dominican-Republic': 70, ' Vietnam': 67, ' Guatemala': 64, ' Japan': 62, ' Poland':
60, ' Columbia': 59, ' Taiwan': 51, ' Haiti': 44, ' Iran': 43, ' Portugal': 37, '
Nicaragua': 34, ' Peru': 31, ' France': 29, ' Greece': 29, ' Ecuador': 28, ' Ireland':
24, ' Hong': 20, ' Cambodia': 19, ' Trinidad&Tobago': 19, ' Thailand': 18, ' Laos':
18, ' Yugoslavia': 16, ' Outlying-US(Guam-USVI-etc)': 14, ' Honduras': 13, ' Hungary':
13, ' Scotland': 12, ' Holand-Netherlands': 1})
```



## 实践拓展

进行相关分析和预测分析的实践拓展。

### (1) 相关分析实践

为了分析定量描述变量（包括年龄、受教育年限、资本收入、资本损失、每周工作小时数）对预测变量（年收入）取值的影响，可以进行相关分析。在进行相关分析之前，需要对年收入的取值进行转换，即若 $\leq 50K$  为 0，若 $> 50K$  为 1。该步骤可通过 Excel 中的替换功能实现。

计算相关系数代码如下：

```
import pandas as pd
data = pd.read_csv('Adult2.csv')
#Adult2 中将 income 取值转换为 0 或 1
income = data['income']

#计算 income 与 age 的 pearson 相关系数
corr_age = income.corr(data['age'],method='pearson')
corr_education_num = income.corr(data['education-num'],method='pearson')
corr_gain = income.corr(data['capital-gain'],method='pearson')
corr_loss = income.corr(data['capital-loss'],method='pearson')
corr_hoursperweek = income.corr(data['hours-per-week'],method='pearson')

print('相关系数计算如下：')
print('corr_age:{:.2f}'.format(corr_age))
print('corr_education_num:{:.2f}'.format(corr_education_num))
print('corr_gain:{:.2f}'.format(corr_gain))
print('corr_loss:{:.2f}'.format(corr_loss))
print('corr_hoursperweek:{:.2f}'.format(corr_hoursperweek))
```

输出结果如下：

相关系数计算如下：

```
corr_age:0.23
corr_education_num:0.34
corr_gain:0.22
corr_loss:0.15
corr_hoursperweek:0.23
```

由输出结果可知，由于年龄、资本收入、资本损失、每周工作小时数与收入的 Pearson 相关系数绝对值小于 0.3，认为不存在线性相关关系；受教育年限与收入的 Pearson 系数绝对值介于(0.3,0.5]之间，存在低度正线性相关关系。

(2) 预测分析实践

经过数据探索与预处理，得到了可以直接建模的数据。根据挖掘目标和数据形式可以建立分类与预测、聚类分析、关联规则、时序模式和偏差检测等模型。

其中，分类和预测是预测分析的两种主要类型，分类主要是预测分类标号（离散属性），而预测主要是建立连续值函数模型，预测给定自变量对应的因变量的值。例如，本例中预测收入是否大于 50K 就是一个典型的二分类问题。

常用的分类与预测算法见下表。

常用的分类与预测算法	
算法名称	算法描述
回归分析	回归分析是确定预测属性（数值型）与其他变量间相互依赖的定量关系最常用的统计学方法，包括线性回归、非线性回归、Logistic 回归、岭回归、主成分回归、偏最小二乘回归等模型
决策树	决策树采用自顶向下的递归方式，在内部节点进行属性值的比较，并根据不同的属性值从该节点向下分支，最终得到的叶节点是学习划分的类
人工神经网络	人工神经网络是一种模仿大脑神经网络结构和功能而建立的信息处理系统，表示神经网络的输入与输出变量之间关系的模型
贝叶斯网络	贝叶斯网络又称信度网络，是 Bayes 方法的拓展，是目前不确定知识表达和推理领域最有效的理论模型之一
支持向量机	支持向量机是一种通过某种非线性映射，把低维的非线性可分转化为高维的线性可分，在高维空间进行线性分析的算法

本实验采用决策树算法进行分类预测，具体实现代码及数据文件下载地址如下：

（实验课题提供）