

**商务数据分析基础**

**指导书**

学 院 管理科学与工程学院

专 业 信息管理与信息系统

指导教师 王杰

二〇二四年九月二日

目 录

[任务一 1](#_Toc9951)

任务[二 2](#_Toc32652)

任务[三 4](#_Toc17437)

任务[四 6](#_Toc3817)

任务[五 8](#_Toc3817)

任务[六 9](#_Toc3817)

任务[七 1](#_Toc3817)1

任务[八 1](#_Toc3817)3

任务[九 1](#_Toc3817)5

报告封皮 [1](#_Toc3817)8

[报告内容及格式要求 1](#_Toc9951)9

成绩评定表20

任务一：Python及pycharm软件的安装与配置

一、实验内容：

1.了解Python语言发展过程及Python语言特点。

2.掌握如何搭建Python运行环境。

3.熟练掌握pycharm开发软件的安装过程及配置

二、实验步骤

1.了解Python语言发展过程及特点：通过查阅Python官方文档、专业书籍（如《Python核心编程》）以及知名技术博客（如InfoQ、开源中国等相关Python板块），详细记录Python从1989年由Guido van Rossum开始设计以来的重要版本发布、功能演进等关键发展节点。同时，总结归纳Python语言简洁易读、开发效率高、可扩展性强、拥有庞大标准库、支持多种编程范式等特点，并结合具体代码示例理解，如简单的变量定义、函数调用代码体现其语法简洁性。

2.搭建Python运行环境：访问Python官方网站（https://www.python.org/ ），在下载页面根据操作系统类型（如Windows、Linux、macOS）选择合适的Python安装包进行下载。下载完成后，在Windows系统中双击安装文件，安装过程中勾选“Add Python to PATH”选项，按照安装向导提示完成安装。安装结束后，打开命令提示符（Windows）或终端（Linux、macOS），输入“python --version”命令，若显示正确的Python版本号，则表明运行环境搭建成功。

3.安装及配置Pycharm开发软件：进入JetBrains官网（https://www.jetbrains.com/pycharm/ ）下载Pycharm安装包，有专业版和社区版可供选择。下载完成后，双击安装包，按照安装向导提示进行操作，如选择安装路径、创建桌面快捷方式等。安装完成后启动Pycharm，在新建项目界面，设置项目保存路径，在“Project Interpreter”中，点击齿轮图标选择“Add”，选择“Python Interpreter”，然后选择已安装的Python解释器路径，完成Pycharm与Python环境的关联配置。

三、实验目的：

本实验旨在使学生掌握Python运行环境的搭建及pycharm开发软件的安装及配置。

四、实验要求：

1.对于了解Python语言发展过程及特点，需通过多种渠道（如官方文档、专业书籍、权威技术网站）收集资料，准确梳理Python从起源至今的关键发展阶段，详细总结其语法简洁、可扩展性强、支持多种编程范式等特点，并能结合实际代码示例阐述。

2.在搭建Python运行环境方面，要严格按照官方指引，根据自身操作系统正确下载对应版本的Python安装包，安装过程中确保关键选项正确勾选。安装完成后，能通过命令行验证运行环境搭建是否成功，若出现问题，可独立排查并解决。

3.针对熟练掌握pycharm开发软件的安装过程及配置，需从官网获取安装包并按步骤完成安装。安装时注意各项参数设置，安装后能正确配置项目与Python解释器的关联，确保Pycharm能正常识别并使用Python环境进行开发，遇到配置问题时，能通过查阅文档或网络资源解决。

任务二 字符串类型上机实践

一、实验内容

本次实验主要围绕Python中字符串类型的基本操作展开，包括字符串的定义、常用方法的应用以及字符串的格式化输出等内容。具体实验内容如下：

1.字符串的定义与基本操作

（1）创建字符串

（2）访问字符串中的字符

（3）字符串切片

（4）字符串连接与重复

2.字符串方法的应用

（1）字符串大小写转换

（2）字符串查找与替换

（3）字符串分割与合并

（4）字符串去空格

3.字符串的格式化输出

（1）使用%进行格式化

（2）使用format()方法进行格式化

（3）使用f-string进行格式化

二、实验步骤

1.环境准备

确保已安装Python环境（推荐使用Python 3.6及以上版本）。打开Python IDE（如PyCharm、VSCode等）或使用Jupyter Notebook。

2.字符串的定义与基本操作

创建一个字符串变量并赋值，例如：s = "Hello, World!"，访问字符串中的字符，例如：print(s[0])，进行字符串切片操作，例如：print(s[7:12])，将两个字符串连接在一起，例如：s1 = "Hello" + ", " + "World!"，重复字符串，例如：s2 = "abc" \* 3。

3.字符串方法的应用

（1）将字符串转换为大写，例如：print(s.upper())

（2）查找字符串中某个子串的位置，例如：print(s.find("World"))

（3）替换字符串中的子串，例如：s3 = s.replace("World", "Python")

（4）分割字符串，例如：words = s.split(", ")

（5）合并字符串列表，例如：s4 = "-".join(words)

（6）去除字符串两端的空格，例如：s5 = " Hello, World! ".strip()

4.字符串的格式化输出

（1）使用%进行格式化，例如：name = "Alice"; age = 25; print("My name is %s and I am %d years old." % (name, age))

（2）使用format()方法进行格式化，例如：print("My name is {} and I am {} years old.".format(name, age))

（3）使用f-string进行格式化，例如：print(f"My name is {name} and I am {age} years old.")

三、实验目的

1.掌握Python中字符串的定义方法及其基本操作。

2.熟练应用字符串的各种方法，如大小写转换、查找与替换、分割与合并等。

3.学会使用不同的方法进行字符串的格式化输出。

4.提高编程实践能力，培养良好的编程习惯。

四、实验要求

1.实验报告：每位同学需提交一份实验报告，内容包括实验目的、实验步骤、实验结果及分析。

2.代码规范：代码需符合Python编程规范，注释清晰，变量命名合理。

3.独立完成：实验过程中需独立完成各项任务，不得抄袭他人代码。

任务三 if分支结构上机实践

一、实验内容

本次实验主要围绕Python中if分支结构的基本用法展开，包括单分支、双分支、多分支以及嵌套if语句等内容。具体实验内容如下：

1.单分支结构：使用if语句实现简单的条件判断。

2.双分支结构：使用if-else语句实现两种不同情况的处理。

3.多分支结构：使用if-elif-else语句实现多种不同情况的处理。

4.嵌套if语句：在if语句内部嵌套其他if语句，实现更复杂的逻辑控制。

二、实验步骤

1.环境准备

确保已安装Python环境（推荐使用Python 3.6及以上版本）。

打开Python IDE（如PyCharm、VSCode等）或使用Jupyter Notebook。

2.单分支结构

编写一个简单的程序，判断一个数是否为正数。

输入一个数，使用if语句判断该数是否大于0，如果是则输出“这是一个正数”。

3.双分支结构

编写一个程序，判断一个数是正数还是负数。

输入一个数，使用if-else语句判断该数是否大于0，如果是则输出“这是一个正数”，否则输出“这是一个负数”。

4.多分支结构

编写一个程序，判断一个数是正数、负数还是零。

输入一个数，使用if-elif-else语句判断该数是否大于0，如果小于0则输出“这是一个负数”，如果等于0则输出“这是零”。

5.嵌套if语句

编写一个程序，判断一个数是否为奇数或偶数，并进一步判断其是否为正数或负数。

输入一个整数，首先使用if语句判断该数是否为偶数，如果是偶数再判断其是否为正数；如果不是偶数，则判断其是否为正数。

三、实验目的

1.掌握if分支结构的定义方法及其基本用法：通过编写和运行简单的if语句，学生能够理解if分支结构的基本语法和执行流程。

2.熟练应用单分支、双分支、多分支以及嵌套if语句：通过不同的实验内容，学生能够掌握如何根据不同的条件选择合适的分支结构。

3.学会根据实际问题选择合适的分支结构：通过实验，学生能够理解在实际编程中如何根据问题的需求选择合适的分支结构，提高编程解决问题的能力。

4.提高编程实践能力：通过动手实践，学生能够提高编程技能，培养良好的编程习惯。

四、实验要求

1.实验报告：每位同学需提交一份实验报告，报告中应包含实验目的、实验内容、实验步骤、实验结果及分析等内容。

2.代码规范：编写代码时，应注意代码的可读性和规范性，包括但不限于变量命名、注释、缩进等。

3.独立完成：实验过程中，学生应独立完成各项任务，不得抄袭他人代码。

任务四 for循环结构实验

一、实验内容

本次实验主要围绕Python中for循环结构的基本用法展开，包括遍历列表、字符串、字典等可迭代对象，以及结合range()函数生成数字序列等内容。具体实验内容如下：

1.遍历列表：使用for循环遍历列表中的每个元素。

2.遍历字符串：使用for循环遍历字符串中的每个字符。

3.遍历字典：使用for循环遍历字典中的键值对。

4.使用range()函数：使用range()函数生成数字序列，并结合for循环进行遍历。

二、实验步骤

1.环境准备

确保已安装Python环境（推荐使用Python 3.6及以上版本）。

打开Python IDE（如PyCharm、VSCode等）或使用Jupyter Notebook。

2.遍历列表

创建一个包含多个元素的列表，如fruits = ["apple", "banana", "cherry"]。

使用for循环遍历该列表，输出每个元素。

3.遍历字符串

创建一个字符串，如word = "python"。

使用for循环遍历该字符串，输出每个字符。

4.遍历字典

创建一个字典，如person = {"name": "Alice", "age": 25, "city": "New York"}。

使用for循环遍历该字典，输出每个键值对。

5.使用range()函数

使用range()函数生成一个数字序列，如range(1, 10)。

使用for循环遍历该数字序列，输出每个数字。

三、实验目的

1.掌握for循环结构的定义方法及其基本用法：通过编写和运行简单的for循环，学生能够理解for循环结构的基本语法和执行流程。

2.熟练应用for循环遍历各种可迭代对象：通过不同的实验内容，学生能够掌握如何使用for循环遍历列表、字符串、字典等可迭代对象。

3.学会使用range()函数生成数字序列：通过实验，学生能够理解如何使用range()函数生成数字序列，并结合for循环进行遍历。

4.提高编程实践能力：通过动手实践，学生能够提高编程技能，培养良好的编程习惯。

四、实验要求

1.实验报告：每位同学需提交一份实验报告，报告中应包含实验目的、实验内容、实验步骤、实验结果及分析等内容。

2.代码规范：编写代码时，应注意代码的可读性和规范性，包括但不限于变量命名、注释、缩进等。

3.独立完成：实验过程中，学生应独立完成各项任务，不得抄袭他人代码。

任务五 while循环结构实践

一、实验内容

本次实验主要围绕Python中while循环结构的基本用法展开，包括基本的循环结构、条件判断、循环控制语句（如break和continue）等内容。具体实验内容如下：

1.基本循环结构：使用while循环实现简单的计数功能。

2.条件判断：结合条件判断语句，实现基于特定条件的循环。

3.循环控制语句：使用break和continue语句控制循环的执行流程。

4.嵌套循环：实现嵌套的while循环，完成更复杂的任务。

二、实验步骤

1.环境准备

确保已安装Python环境（推荐使用Python 3.6及以上版本）。

打开Python IDE（如PyCharm、VSCode等）或使用Jupyter Notebook。

2.基本循环结构

编写一个简单的while循环，实现从1到10的计数功能。

输出每次循环的计数值。

3.条件判断

编写一个while循环，实现从1到100的累加求和功能。

在循环中加入条件判断，当累加和超过500时，停止循环并输出当前的累加和。

4.循环控制语句

编写一个while循环，实现从1到50的遍历。

在循环中加入break语句，当遍历到数值25时，提前结束循环。

在另一个循环中加入continue语句，跳过数值为偶数的情况，只输出奇数。

5.嵌套循环

编写一个嵌套的while循环，实现打印九九乘法表的功能。

外层循环控制行数，内层循环控制列数，输出相应的乘积。

三、实验目的

1.掌握while循环结构的定义方法及其基本用法：通过编写和运行简单的while循环，学生能够理解while循环结构的基本语法和执行流程。

2.熟练应用while循环结合条件判断实现复杂逻辑：通过不同的实验内容，学生能够掌握如何使用while循环结合条件判断实现更复杂的逻辑。

3.学会使用循环控制语句：通过实验，学生能够理解如何使用break和continue语句控制循环的执行流程。

4.提高编程实践能力：通过动手实践，学生能够提高编程技能，培养良好的编程习惯。

四、实验要求

1.实验报告：每位同学需提交一份实验报告，报告中应包含实验目的、实验内容、实验步骤、实验结果及分析等内容。

2.代码规范：编写代码时，应注意代码的可读性和规范性，包括但不限于变量命名、注释、缩进等。

3.独立完成：实验过程中，学生应独立完成各项任务，不得抄袭他人代码。

任务六 break和continue关键字上机实践

一、实验内容

本次实验主要围绕Python中break和continue关键字的基本用法展开，包括基本的循环控制、条件判断、循环中断等内容。具体实验内容如下：

1.基本循环控制：使用break和continue关键字控制循环的执行流程。

2.条件判断：结合条件判断语句，实现基于特定条件的循环中断和跳过。

3.嵌套循环控制：在嵌套循环中使用break和continue关键字，完成更复杂的任务。

二、实验步骤

1.环境准备

确保已安装Python环境（推荐使用Python 3.6及以上版本）。

打开Python IDE（如PyCharm、VSCode等）或使用Jupyter Notebook。

2.基本循环控制

编写一个简单的while循环，实现从1到10的计数功能。

在循环中加入break语句，当计数值为5时，提前结束循环。

在另一个循环中加入continue语句，跳过数值为偶数的情况，只输出奇数。

3.条件判断

编写一个while循环，实现从1到100的累加求和功能。

在循环中加入条件判断，当累加和超过500时，使用break语句停止循环并输出当前的累加和。

在另一个循环中，使用continue语句跳过数值为3的倍数的情况，只累加其他数值。

4.嵌套循环控制

编写一个嵌套的while循环，实现打印九九乘法表的功能。

在外层循环中加入break语句，当行数为5时，提前结束外层循环。

在内层循环中加入continue语句，跳过数值为偶数的情况，只输出奇数的乘积。

三、实验目的

1.掌握break和continue关键字的定义方法及其基本用法：通过编写和运行简单的循环控制代码，学生能够理解break和continue关键字的基本语法和执行流程。

2.熟练应用break和continue关键字结合条件判断实现复杂逻辑：通过不同的实验内容，学生能够掌握如何使用这两个关键字结合条件判断实现更复杂的逻辑。

3.学会在嵌套循环中使用break和continue关键字：通过实验，学生能够理解如何在嵌套循环中使用这两个关键字控制循环的执行流程。

4.提高编程实践能力：通过动手实践，学生能够提高编程技能，培养良好的编程习惯。

四、实验要求

1.实验报告：每位同学需提交一份实验报告，报告中应包含实验目的、实验内容、实验步骤、实验结果及分析等内容。

2.代码规范：编写代码时，应注意代码的可读性和规范性，包括但不限于变量命名、注释、缩进等。

3.独立完成：实验过程中，学生应独立完成各项任务，不得抄袭他人代码。

任务七 组合数据类型上机实操

一、实验内容

本次实验主要围绕Python中的四种组合数据类型展开，包括列表、元组、字典和集合的基本操作和应用。具体实验内容如下：

1.列表操作：创建列表、添加元素、删除元素、修改元素、遍历列表等。

2.元组操作：创建元组、访问元素、遍历元组等。

3.字典操作：创建字典、添加键值对、删除键值对、修改键值对、遍历字典等。

4.集合操作：创建集合、添加元素、删除元素、集合运算等。

二、实验步骤

1.环境准备

确保已安装Python环境（推荐使用Python 3.6及以上版本）。

打开Python IDE（如PyCharm、VSCode等）或使用Jupyter Notebook。

2.列表操作

创建一个空列表，并使用append()方法添加元素。

使用remove()方法删除指定元素，使用pop()方法删除末尾元素。

修改列表中指定位置的元素。

使用for循环遍历列表，并打印每个元素。

3.元组操作

创建一个包含多个元素的元组。

访问元组中的指定元素，并打印其值。

使用for循环遍历元组，并打印每个元素。

4.字典操作

创建一个空字典，并使用[]操作符添加键值对。

使用del关键字删除指定键值对，使用pop()方法删除指定键值对。

修改字典中指定键的值。

使用for循环遍历字典的键和值，并打印每个键值对。

5.集合操作

创建一个空集合，并使用add()方法添加元素。

使用remove()方法删除指定元素，使用discard()方法删除指定元素。

进行集合的交集、并集、差集运算，并打印结果。

三、实验目的

1.掌握列表、元组、字典和集合的定义方法及其基本操作：通过编写和运行简单的代码，学生能够理解这四种组合数据类型的基本语法和操作方法。

2.熟练应用组合数据类型解决实际问题：通过不同的实验内容，学生能够掌握如何使用这些数据类型解决实际问题。

3.学会在不同场景下选择合适的数据类型：通过实验，学生能够理解不同数据类型的特点和适用场景，从而在实际编程中做出合理的选择。

4.提高编程实践能力：通过动手实践，学生能够提高编程技能，培养良好的编程习惯。

四、实验要求

1.实验报告：每位同学需提交一份实验报告，报告中应包含实验目的、实验内容、实验步骤、实验结果及分析等内容。

2.代码规范：编写代码时，应注意代码的可读性和规范性，包括但不限于变量命名、注释、缩进等。

3.独立完成：实验过程中，学生应独立完成各项任务，不得抄袭他人代码。

任务八 函数用法上机实践

一、实验内容

本次实验主要围绕Python中的函数用法展开，包括函数的定义、参数传递、变量作用域、嵌套调用和递归调用等内容。具体实验内容如下：

1.函数定义：定义一个简单的函数，并调用该函数。

2.参数传递：定义带有参数的函数，并传递不同类型的参数。

3.变量作用域：理解局部变量和全局变量的作用域。

4.嵌套调用：在一个函数内部调用另一个函数。

5.递归调用：定义一个递归函数，并解决实际问题。

二、实验步骤

1. 环境准备

确保已安装Python环境（推荐使用Python 3.6及以上版本）。

打开Python IDE（如PyCharm、VSCode等）或使用Jupyter Notebook。

2. 函数定义

定义简单函数：定义一个简单的函数，并在函数内部打印一条消息。

定义一个名为greet的函数，该函数不接受任何参数，仅在控制台打印“Hello, World!”。

调用greet函数以验证其功能。

3. 参数传递

定义带参数的函数：定义一个带有参数的函数，并在函数内部使用这些参数。

定义一个名为greet的函数，该函数接受一个参数name，并在控制台打印“Hello, {name}!”。

调用greet函数并传递不同的名字作为参数，以验证其功能。

4. 变量作用域

理解局部变量：在函数内部定义一个局部变量，并观察其作用域。

定义一个名为local\_scope的函数，在该函数内部定义一个局部变量x并赋值为10，然后在函数内部打印x的值。

调用local\_scope函数，并尝试在函数外部访问局部变量x（预期会报错）。

理解全局变量：在函数外部定义一个全局变量，并在函数内部访问和修改它。

在函数外部定义一个全局变量y并赋值为20。

定义一个名为global\_scope的函数，在该函数内部使用global关键字声明要修改全局变量y，将其值改为30，并在函数内部打印y的值。

调用global\_scope函数，并在函数外部再次打印y的值，以验证全局变量已被修改。

5. 嵌套调用

在一个函数内部调用另一个函数：定义两个函数，并在一个函数内部调用另一个函数。

定义一个名为inner\_function的函数，在该函数内部打印“这是内部函数”。

定义一个名为outer\_function的函数，在该函数内部首先调用inner\_function，然后打印“这是外部函数”。

调用outer\_function函数以验证嵌套调用的功能。

6. 递归调用

定义递归函数：定义一个递归函数来解决实际问题，例如计算阶乘。

定义一个名为factorial的递归函数，该函数接受一个参数n，如果n等于0，则返回1；否则返回n乘以factorial(n-1)的结果。

调用factorial函数并传递一个正整数作为参数，以验证递归函数的功能。

三、实验目的

通过本次实验，学生将能够：

1.掌握Python中函数的定义和调用方法。

2.理解函数参数的传递方式及其应用。

3.区分局部变量和全局变量的作用域。

4.掌握函数嵌套调用和递归调用的实现方法。

5.提高代码的模块化和复用性，增强编程能力。

四、实验要求

1.独立完成：学生需独立完成所有实验内容，不得抄袭他人代码。

2.代码规范：确保代码格式规范，变量命名清晰，注释详细。

3.实验报告：撰写实验报告，记录实验过程、遇到的问题及解决方案，并总结实验心得。

任务九 类和对象编程上机实践

一、实验内容：

1.了解面向对象编程的思想。

2.掌握类的三大特征及编程实现。

3.熟练掌握运用面向对象思想进行程序设计的方法。

二、实验步骤

1.了解面向对象编程的思想：查阅相关教材、在线教程（如慕课网、网易云课堂上的相关课程）以及技术文档，学习面向对象编程思想的概念，包括对象、类、封装、继承、多态等核心概念。通过阅读经典代码示例，如简单的学生类示例，分析代码如何将数据和操作封装在一起形成对象，以及对象之间如何通过消息传递进行交互，理解面向对象编程如何模拟现实世界中的事物和关系，记录学习过程中的理解和疑问。

2.掌握类的三大特征及编程实现：首先，在编程语言（如Python、Java等）中创建一个简单类，在类中定义属性和方法，将数据和操作封装在一起，实现封装特征，观察如何通过对象访问和修改属性及调用方法。接着，创建父类和子类，子类继承父类的属性和方法，体现继承特征，编写代码展示子类如何扩展父类功能。最后，利用多态性，在父类中定义抽象方法，子类重写该方法，通过父类引用调用不同子类对象的重写方法，实现同一操作对不同对象产生不同行为，记录实现过程和遇到的问题及解决方法。

3.熟练掌握运用面向对象思想进行程序设计的方法：确定一个实际问题场景，如设计一个简单的学校管理系统。分析问题，确定系统中的对象（如学生、教师、课程等）和类。为每个类定义合适的属性和方法，利用封装确保数据的安全性。通过继承实现类之间的层次关系，如学生类和教师类可能继承自人员类。运用多态处理不同对象的相同操作，如不同课程的教学方法。编写完整的程序代码实现系统功能，进行调试和测试，不断优化代码，总结运用面向对象思想解决实际问题的经验和技巧。

三、实验目的：

本实验旨在使学生掌握类的三大特征，并且会运用面向对象的方法进行程序设计。

四、实验要求：

1.对于了解面向对象编程的思想，需认真研读相关教材、专业文档及优质网络教程，清晰掌握面向对象编程中对象、类、封装、继承、多态等核心概念，能准确阐述其含义。通过分析多个实际代码案例，深入理解面向对象编程如何模拟现实世界，体会其相较于其他编程范式的优势，形成对该编程思想的全面认知，并撰写学习心得。

2.掌握类的三大特征及编程实现方面，要在熟悉的编程语言中，分别编写代码实现封装、继承和多态。封装时，合理设计类的属性和方法访问权限，确保数据安全与操作规范；继承实现上，构建清晰的类层次结构，明确父类与子类关系，展示子类对父类功能的扩展；多态实现需通过方法重写和父类引用调用不同子类对象方法，体现同一操作的不同表现。代码实现过程中，详细记录遇到的问题及解决方法，保证代码逻辑正确、运行无误。

3.熟练掌握运用面向对象思想进行程序设计的方法上，要自主选取一个具有一定复杂度的实际问题，运用面向对象思想进行系统分析与设计。准确识别问题中的对象与类，合理设计类的属性和方法，有效运用封装、继承、多态特性构建程序结构。编写完整、可运行的程序代码，进行全面的调试与测试，优化程序性能，最终提交详细的设计文档和高质量代码。4. 文档撰写：编写实验报告，记录实验过程、代码实现、测试结果以及广义表和数组的比较分析。



**商务数据分析基础实验报告**

**姓名：**

**学号：**

**班级：**

**报告内容包括以下三部分：**

一、实验概况：实验的过程、实验内容等。

二、实验的主要收获：结合实际实验内容，自己所获取的感性知识及实践经验等。

三、实验的真实体会及建议。

**报告格式要求：**

1. 一级标题

字号：二号

加粗：是

行间距：26磅

对齐方式：左对齐

2. 二级标题

字号：小二号

加粗：是

行间距：26磅

对齐方式：左对齐

3. 正文

字号：小四

行间距：26磅

段落首行缩进：2字符

4. 页边距

上、下、左、右：均设置为2.54厘米