

Python 程序设计实验课

实验一：Python 程序认知

实验目的

1. 熟悉 Python 开发环境的配置和使用
2. 初步了解 Python 基本语法和程序结构
3. 理解交互式编程与文件式编程
4. 识别和解决常见的编程问题

实验环境

- Python 3.x
 - 代码编辑器 (IDLE、Spyder、VS Code、PyCharm 或 Jupyter Notebook 均可)
-

第一部分：微实例 1A-1E 复现（课本 31 页-35 页）

实验要求：

1. 运行下述程序，观察输出结果
2. 尝试修改输入内容。
3. 观察程序问题，试着找出问题原因

微实例1：圆面积的计算



根据半径r计算圆面积

```
>>> r = 25  
>>> area = 3.1415 * r * r  
>>> print(area)  
1963.4375000000002  
>>> print("{:.2f}".format(area))  
1963.44
```

交互式

微实例1：圆面积的计算



根据半径r计算圆面积

```
r = 25  
area = 3.1415 * r * r  
print(area)  
print("{:.2f}".format(area))
```

输出结果如下：

```
1963.4375000000002  
1963.44
```

保存为CalCircleArea.py文件并运行

文件式

微实例2：简单的人名对话



对用户输入的人名进行回应

```
>>> name = input("输入姓名:")  
输入姓名:郭靖  
>>> print("{}同学, 学好 python, 前途无量!".format(name))  
郭靖同学, 学好 python, 前途无量!  
>>> print("{}大侠, 学好 Python, 大展拳脚!".format(name[0]))  
郭大侠, 学好 Python, 大展拳脚!  
>>> print("{}哥哥, 学好 Python, 人见人爱!".format(name[1:]))  
靖哥哥, 学好 Python, 人见人爱!
```

交互式

微实例2：简单的人名对话



对用户输入的人名进行回应

```
name = input("输入姓名:")  
print("{}同学, 学好 python, 前途无量!".format(name))  
print("{}大侠, 学好 Python, 大展拳脚!".format(name[0]))  
print("{}哥哥, 学好 Python, 人见人爱!".format(name[1:]))
```

保存为EchoName.py文件并运行

输出结果如下：
输入姓名:郭靖
郭靖同学, 学好 python, 前途无量!
郭大侠, 学好 Python, 大展拳脚!
靖哥哥, 学好 Python, 人见人爱!

文件式

微实例3：斐波那契数列的计算



输出不大于1000的斐波那契序列元素

```
>>> a,b = 1,1  
>>> while a < 1000:  
    print(a, end=',')  
    a, b = b, a + b  
  
1,1,2,3,5,8,13,21,34,55,89,144,233,377,610,987,
```

交互式

微实例3：斐波那契数列的计算



输出不大于1000的斐波那契序列元素

```
a,b = 1,1  
while a < 1000:  
    print(a, end=',')  
    a, b = b, a + b
```

输出结果如下：

```
1,1,2,3,5,8,13,21,34  
,55,89,144,233,377,6  
10,987,
```

保存为CalFibonacci.py文件并运行

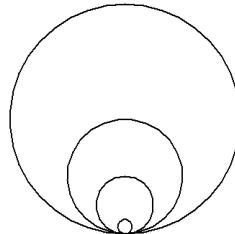
文件式

微实例4：同切圆的绘制



绘制多个同切圆

```
>>> import turtle  
>>> turtle.pensize(2)  
>>> turtle.circle(10)  
>>> turtle.circle(40)  
>>> turtle.circle(80)  
>>> turtle.circle(160)
```



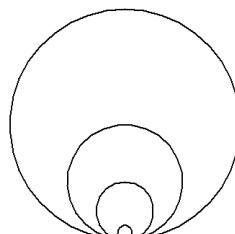
交互式

微实例4：同切圆的绘制



绘制多个同切圆

```
import turtle  
turtle.pensize(2)  
turtle.circle(10)  
turtle.circle(40)  
turtle.circle(80)  
turtle.circle(160)
```



保存为TangentCirclesDraw.py文件并运行

文件式

微实例5：日期和时间的输出



输出当前计算机的系统日期和时间

```
>>> from datetime import datetime  
>>> now = datetime.now()  
>>> print(now)  
2024-12-06 17:28:21.975283  
>>> print(now.strftime("%x"))  
12/06/24  
>>> print(now.strftime("%X"))  
17:28:21
```

交互式

微实例5：日期和时间的输出



输出当前计算机的系统日期和时间

```
from datetime import datetime  
now = datetime.now()  
print(now)  
print(now.strftime("%x"))  
print(now.strftime("%X"))
```

输出结果如下：

```
2024-12-06 17:28:21.975283  
12/06/24  
17:28:21
```

保存为PrintLocalTime.py文件并运行

文件式

第二部分：程序练习题

实验要求：

1. 运行课本 42-44 页程序练习题 1.1-1.8 程序，观察输出结果
2. 尝试修改输入内容。

3. 观察程序问题，试着找出问题原因

第三部分：问题记录与反思

常见问题记录

请根据你的实际编程体验，回答以下问题（参考课本 35 页-38 页）：

1. **语法错误**：你遇到了哪些语法错误？如何解决的？
 - 示例：缩进错误、括号不匹配、冒号缺失等
2. **运行时错误**：程序运行时报错的有哪些？
 - 示例：变量未定义、除零错误、类型错误等
3. **逻辑错误**：程序能运行但结果不对的情况？
 - 示例：条件判断错误、循环次数错误、公式写错等
4. **其他问题**：除了课本提到的问题，你还遇到了什么困难？
 - 环境配置问题
 - 中文编码问题
 - 输入输出格式问题
 - 其他...

反思与总结

1. 本次实验你最大的收获是什么？
 2. 哪个部分对你来说最有挑战性？
 3. 你如何解决遇到的问题？
 4. 对 Python 语言有什么新的认识？
-

实验报告要求（使用学习通里的实验报告模板）

报告格式

1. 封面：填写姓名、学号、指导老师、日期
2. 填写第二页个人和实验信息栏
3. 实验目的：简要说明
4. 实验内容：
 - 所有代码（可截图或粘贴代码）
 - 运行结果截图
5. 问题分析：
 - 详细记录遇到的问题
 - 说明解决方法和思路
6. 实验总结（参考上面反思与总结写）：
 - 收获与体会
 - 对后续学习的建议

提交方式

- 电子版：上传学习通作业目录下的实验报告 1 文件夹里（如下图），命名格式：实验报告 1_姓名_学号，**10月16日截止**

Python程序设计课程门户

首页 活动 统计 资料 通知 作业 考



- 纸质版：**10月17日实验课交**

注意事项：

1. 所有代码必须亲自编写和调试
2. 遇到问题先尝试独立解决，再寻求帮助
3. 保持代码规范和良好的注释习惯
4. 按时完成并提交实验报告

祝你实验顺利！