

实验二 Python变量、简单数据类型

班级： 21计科1班

学号： B20210302106

姓名： 彭浩

Github地址： <https://github.com/234519402/pythonpeng>

CodeWars地址： <https://www.codewars.com/users/234519402>

实验目的

1. 使用VSCode编写和运行Python程序
2. 学习Python变量和简单数据类型

实验环境

1. Git
2. Python 3.10
3. VSCode
4. VSCode插件

实验内容和步骤

第一部分

实验环境的安装

1. 安装Python，从Python官网下载Python 3.10安装包，下载后直接点击可以安装：[Python官网地址](#)
 2. 为了在VSCode集成环境下编写和运行Python程序，安装下列VScode插件
 - Python
 - Python Environment Manager
 - Python Indent
 - Python Extended
 - Python Docstring Generator
 - Jupyter
 - indent-rainbow
 - Jinja
-

第二部分

Python变量、简单数据类型和列表简介

完成教材《Python编程从入门到实践》下列章节的练习：

- 第2章 变量和简单数据类型

第三部分

在[Codewars网站](https://www.codewars.com/)注册账号，完成下列Kata挑战：

第1题：求离整数n最近的平方数 (Find Nearest square number)

难度：8kyu

你的任务是找到一个正整数n的最近的平方数 例如，如果n=111，那么nearest_sq(n) (nearestSq(n)) 等于121，因为111比100（10的平方）更接近121（11的平方）。如果n已经是完全平方（例如n=144，n=81，等等），你需要直接返回n。代码提交地址 <https://www.codewars.com/kata/5a805d8cafa10f8b930005ba>

第2题：弹跳的球 (Bouncing Balls)

难度：6kyu

一个孩子在一栋高楼的第N层玩球。这层楼离地面的高度h是已知的。他把球从窗口扔出去。球弹了起来，例如：弹到其高度的三分之二（弹力为0.66）。他的母亲从离地面w米的窗户向外看，母亲会看到球在她的窗前经过多少次（包括球下落和反弹的时候）？

一个有效的实验必须满足三个条件：

- 参数 "h"（米）必须大于0
- 参数 "bounce" 必须大于0且小于1
- 参数 "window" 必须小于h。

如果以上三个条件都满足，返回一个正整数，否则返回-1。 **注意：只有当反弹球的高度严格大于窗口参数时，才能看到球。** 代码提交地址 <https://www.codewars.com/kata/5544c7a5cb454edb3c000047/train/python>

第3题：元音统计(Vowel Count)

难度：7kyu

返回给定字符串中元音的数量（计数）。对于这个Kata，我们将考虑a、e、i、o、u作为元音（但不包括y）。输入的字符串将只由小写字母和/或空格组成。

代码提交地址： <https://www.codewars.com/kata/54ff3102c1bad923760001f3>

第4题：偶数或者奇数 (Even or Odd)

难度：8kyu

创建一个函数接收一个整数作为参数，当整数为偶数时返回"Even"当整数为奇数时返回"Odd"。代码提交地址： <https://www.codewars.com/kata/53da3dbb4a5168369a0000fe>

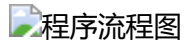
第四部分

使用Mermaid绘制程序流程图

安装Mermaid的VSCode插件：

- Markdown Preview Mermaid Support
- Mermaid Markdown Syntax Highlighting

使用Markdown语法绘制你的程序绘制程序流程图（至少一个），Markdown代码如下：



显示效果如下：

```
graph LR
    A[Start] --> B{Is it?}
    B -->|Yes| C[OK]
    C --> D[Rethink]
    D --> B
    B -.->|No| E[End]
```

查看Mermaid流程图语法-->[点击这里](#)

使用Markdown编辑器（例如VScode）编写本次实验的实验报告，包括[实验过程与结果](#)、[实验考查](#)和[实验总结](#)，并将其导出为 **PDF格式** 来提交。

实验过程与结果

请将实验过程与结果放在这里，包括：

- [第二部分 Python变量、简单数据类型和列表简介](#)
- [第三部分 Codewars Kata挑战](#)

第1题：求离整数n最近的平方数

```
def nearest_sq(n):
    import math
    m = round(math.sqrt(n))
    return m * m
```

第2题：弹跳的球

```
def bouncing_ball(h, bounce, window):
    if h <= 0 or bounce <= 0 or bounce >= 1 or window >= h:
        return -1

    bounce_count = 1
    current_height = h * bounce
```

```
while current_height > window:
    bounce_count += 2
    current_height *= bounce

return bounce_count
```

第3题：元音统计

```
def get_count(sentence):
    yuanyin = "aeiou"
    sentence = sentence.lower()
    count = 0
    for char in sentence:
        if char in yuanyin:
            count += 1
    return count
```

第4题：偶数或者奇数

```
def even_or_odd(number):
    if number % 2 == 0:
        return "Even"
    else:
        return "Odd"
```

- [第四部分 使用Mermaid绘制程序流程图](#)

```
flowchart LR
    A[Start] --> B{ number % 2 }
    B -->|0| C[Even]
    B -->|1| E[Odd]
```

注意代码需要使用markdown的代码块格式化，例如Git命令行语句应该使用下面的格式：

 Git命令

显示效果如下：

```
git init
git add .
git status
git commit -m "first commit"
```

如果是Python代码，应该使用下面代码块格式，例如：



显示效果如下：

```
def add_binary(a,b):  
    return bin(a+b)[2:]
```

代码运行结果的文本可以直接粘贴在这里。

注意：不要使用截图，Markdown文档转换为Pdf格式后，截图可能会无法显示。

实验考查

请使用自己的语言并使用尽量简短代码示例回答下面的问题，这些问题将在实验检查时用于提问和答辩以及实际的操作。

1. Python中的简单数据类型有那些？我们可以对这些数据类型做哪些操作？

数据类型有：整数（int），浮点数（float），字符串（str），列表（list）

可以进行：算术操作,比较操作,位运算,逻辑操作,比较操作 字符串连接,字符串复制,字符串长度,字符串索引和切片 增删元素,访问元素,列表切片,列表合并

2. 为什么说Python中的变量都是标签？

在传统的编程语言中，变量通常被认为是存储数据值的容器，当将一个值分配给变量时，通常会在计算机的内存中为该值分配一块存储空间，然后将变量绑定到该存储空间。这意味着变量实际上是数据的存储位置。然而，在Python中，变量实际上是指向内存中对象的标签。当将一个值分配给变量时，Python不是在内存中创建一个新的存储空间来存储该值，而是创建一个对象，然后将变量绑定到该对象。如果创建另一个变量并将其设置为与第一个变量相等，那么这两个变量将引用相同的对象，而不是复制对象。

3. 有哪些方法可以提高Python代码的可读性？

有意义的变量名：为变量、函数、类等使用描述性的名称，使其易于理解。避免使用单个字符或不相关的名称。

注释：为代码添加注释，解释代码的目的、功能和关键步骤。但不要过度注释，应注释那些真正需要解释的部分。

适当的缩进：使用一致的缩进风格（通常是4个空格）来表示代码块。Python的缩进是强制的，所以确保它始终保持一致。

代码布局：使用空行来分隔不同的代码块，使代码更具结构。例如，在函数之间、在类定义之间、在逻辑代码块之间使用空行。

规范的命名约定：遵循Python的命名约定，如使用小写字母和下划线来命名变量和函数（snake_case），使用首字母大写的驼峰命名法来命名类（CamelCase）。

代码审查：让同事或其他人审查您的代码，以获取反馈和建议，有助于发现潜在的改进点。

实验总结

总结一下这次实验你学习和使用到的知识，例如：编程工具的使用、数据结构、程序语言的语法、算法、编程技巧、编程思想。

本次实验学习了Python中的数据类型，并了解了如何对这些数据类型进行操作，进行了简单的python入门学习，且学习了使用Mermaid绘制程序流程图。