

实验二 Python变量、简单数据类型

班级： 21计科02

学号： 20210302221

姓名： 王日晖

Github地址： <https://github.com/wrrh>

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

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：

程序流程图

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](#)
- [第四部分 使用Mermaid绘制程序流程图](#)

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

显示效果如下：

```
import math

def nearest_sq(n):
    sqrt_n = int(math.sqrt(n))

    if sqrt_n**2 == n:
        return n
    else:
        lower_sq = sqrt_n**2
        upper_sq = (sqrt_n + 1)**2

        if n - lower_sq < upper_sq - n:
            return lower_sq
        else:
```

```
return upper_sq
```

第2题：弹跳的球（弹跳球）

显示效果如下：

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

    count = 1

    while h * bounce > window:
        count += 2
        h= h* bounce

    return count
```

第3题：元音统计（元音计数）

显示效果如下：

```
def get_count(sentence):
    s = ['a','e','i','o','u']
    count = 0
    for ch in sentence:
        if ch=='a' or ch=='e' or ch == 'i' or ch== 'o' or ch== 'u':
            count +=1
    return count
```

第4题：偶数或者奇数（偶数或奇数）

显示效果如下：

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

第4题：偶数或者奇数（偶数或奇数）程序流程图 显示效果如下：

```
flowchart LR
    A[Start] --> B{number % 2}
    B -->|Yes| C[return 'Odd']
    B -->|No| D[return 'Even']
    C --> E[End]
    D --> E[End]
```

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？

Python中的简单数据类型包括整数（int）、浮点数（float）、布尔值（bool）、字符串（str）。我们可以对这些数据类型进行基本的操作，比如加减乘除运算、比较运算、逻辑运算等。

2. 为什么说Python中的变量都是标签？

在Python中，变量是用来存储数据的名字，但实际上，变量并不直接存储数据，而是存储了数据的内存地址，也就是数据所在的位置。因此，可以说Python中的变量都是标签，它们指向了数据所在的内存地址。这种设计可以节省内存空间，并且使得变量可以指向不同的数据。

3. 有哪些方法可以提高Python代码的可读性？

提高Python代码的可读性的方法有：
使用有意义的变量名和函数名，可以让代码更易于理解和维护。
添加适当的注释，可以解释代码的功能和实现细节。
缩进代码块，使得代码的结构更清晰，便于阅读。
使用空行分隔不同的功能模块，让代码更具层次感。
遵循PEP 8编码规范，统一代码的风格，提高可读性。
使用合适的数据结构和算法，能够使代码更简洁和高效。
将代码分解成小的、可重用的函数或方法，使得代码更易于理解和调试。

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。

通过这次实验，我学习了Markdown语法的基本用法。同时，我也巩固了Python编程语言的知识，包括简单数据类型的操作和变量的特点。此外，我还学习了提高代码可读性的一些方法，这对于编写清晰、易懂的代码是非常重要的。

在今后的学习和工作中，我将继续运用Markdown语法来编写文档和笔记，以提高自己的文档整理和表达能力。同时，我也会继续深入学习Python编程语言，不断提升自己的编程技巧和解决问题的能力。

在编程实践中，我会更加注重代码的可读性和可维护性，尽量遵循良好的编码习惯和规范，使代码更易于理解和修改。我也会持续学习和掌握更多的数据结构、算法和编程技巧，以应对各种实际问题 and 挑战。

总之，通过这次实验，我不仅学到了Markdown语法的使用和Python编程的知识，还培养了自己的思考解决问题的能力。我会将这些知识和技能应用到实际中，不断进步和成长。