

实验二 Python变量、简单数据类型

班级：21计科1班

学号：20210302123

姓名：何香仪

Github地址：https://github.com/yourusername/python_course

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

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](https://www.codewars.com/)网站注册账号，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

1. Your task is to find the nearest square number, nearest_sq(n) or nearestSq(n), of a positive integer n. For example, if $n = 111$, then nearest_sq(n) (nearestSq(n)) equals 121, since 111 is closer to 121, the square of 11, than 100, the square of 10. If the n is already the perfect square (e.g. $n = 144$, $n = 81$, etc.), you need to just return n

```
import math
def nearest_sq(n):
    n=n**0.5
    n=round(n)
    n=n**2
    return n
```

2. A child is playing with a ball on the nth floor of a tall building. The height of this floor above ground level, h, is known. He drops the ball out of the window. The ball bounces (for example), to two-thirds of its height (a bounce of 0.66). His mother looks out of a window 1.5 meters from the ground. How many times will the mother see the ball pass in front of her window (including when it's falling and bouncing?)

```
import math
def bouncing_ball(h, bounce, window):
```

```

res = -1
if(bounce > 0 and bounce < 1):
    while(h > 0 and window < h):
        res +=2
        h *=bounce
    return res

```

3.Return the number (count) of vowels in the given string.We will consider a, e, i, o, u as vowels for this Kata (but not y).The input string will only consist of lower case letters and/or spaces

```

def get_count(input_str):
    num_vowels = 0
    h=['a','e','i','o','u']
    for i in h:
        num_vowels+=input_str.count(i)
    return num_vowels

```

4.Create a function that takes an integer as an argument and returns "Even" for even numbers or "Odd" for odd numbers

```

def even_or_odd(number):
    quo, rem = divmod(number,2)
    if rem == 0:
        pd="Even"
        return pd
    else:
        pd="Odd"
        return pd

```

```

graph TD
A(Start) --> B{Even?}
B --Yes --> C[return Even]
C --> D(End)
B --No --> E[return Odd]
E --> D

```

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？

整型：可以进行加减乘除、取余、幂运算等。还可以使用一些内置函数，如`abs`（返回绝对值）、`divmod`（返回商和余数）等。

浮点型：除了整型的那些运算外，还可以进行四舍五入、向上取整、向下取整等操作。也有一些内置函数，如`math`模块中的函数。

复数：可以进行加减乘除、取绝对值、求共轭复数等操作。

布尔型：可以进行逻辑运算，如与、或、非等。字符串：可以进行拼接、切片、查找替换等操作。还有很多内置方法，如`split`（分割）、`join`（连接）、`replace`（替换）等。

空值：`None`在Python中主要用于定义空值或者对象没有返回值时的默认值。你可以使用`is`和`is not`操作符来检查一个值是否为`None`。

每种数据类型都有自己的特性和方法，使得我们能对它们进行各种操作。在Python中，你可以使用`type()`函数来查看一个值的类型

2.为什么说Python中的变量都是标签？

在Python中，变量被视为标签，这是因为变量实际上是指向存储在内存中的对象的引用。换句话说，当你创建一个变量并为其赋值时，Python会在内存中创建一个对象来保存该值，然后将你的变量名（即标签）与这个对象关联起来。这意味着在Python中，变量本身并不直接存储值。相反，它们是指向存储实际数据的对象的指针或标签。这也是为什么Python中的变量可以动态地更改类型，因为变量只是指向对象的标签，对象本身的类型可以更改

3.有哪些方法可以提高Python代码的可读性？

使用有意义的变量名：变量名应该简洁而描述性，让读者能够理解变量的用途。

使用空格和空行来分隔代码块：这可以帮助读者更清晰地理解代码的结构。

使用注释：在复杂的代码块或不易理解的代码段之前，添加注释来解释它们的功能。

避免使用复杂的表达式：复杂的表达式可能会使代码难以阅读和理解。尽量将复杂的表达式分解成简单的步骤。

适当使用空格和缩进：在Python中，缩进是非常重要的，因为它决定了代码的结构。适当的使用空格和缩进可以使代码更易读。

使用文档字符串：为函数、类和模块添加文档字符串，以解释它们的功能和用法。

实验总结

通过本次实验，我学会了如何使用`mermaid`语法画流程图，且能够更加灵活的使用`vscode`，与此同时，我对`python`语法更加的熟悉。