

# 实验二 Python变量、简单数据类型

班级： 21计科1班

学号： 20210302306

姓名： 陈卓

Github地址： <https://github.com/1chenzhuo1/pythonshiyan>

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

---

## 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](#)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 $\text{nearest\_sq}(n)$  ( $\text{nearestSq}(n)$ ) 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—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 " $\text{bounce}$ " 必须大于0且小于1
- 参数 " $\text{window}$ " 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

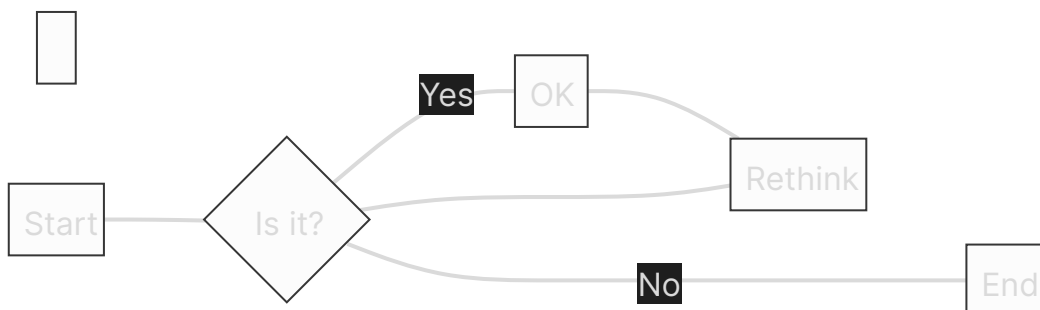
安装Mermaid的VSCode插件：

- 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：

"/Experiments/img/2023-08-05-22-00-00.png" 未创建，点击以创建。

显示效果如下：



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](#)

### 练习2.1:

```
message = "Hello World!"  
  
print(message)
```

输出：

Hello World!

### 练习2.2:

```
message = "Hello World!"  
  
print(message)  
  
message = "Hello Python World!"  
  
print(message)
```

输出：

Hello World!

Hello Python World!

### 练习2.3:

```
name = "Eric"  
  
message = f"Hello {name}, would you like to learn some Python today?"
```

```
print(message)
```

输出:

Hello Eric,would you like to learn some Python today?

### 练习2.4:

```
name = "Chen zhao"  
  
print(name.upper())  
  
print(name.lower())  
  
print(name.title())
```

输出:

CHEN ZHAO

chen zhao

Chen Zhao

### 练习2.5:

```
name = "Napoleon"  
  
message = f"{name} once said,\"True intelligence is determined ambition.\""  
  
print(message)
```

输出:

Napoleon once said,"True intelligence is determined ambition."

### 练习2.6:

```
famous_person = "Napoleon"

message = f"{famous_person} once said,\"True intelligence is determined
ambition.\"\"

print(message)
```

输出:

Napoleon once said,"True intelligence is determined ambition."

### 练习2.7:

```
name = '\t"chenzhuo"\t\n'

print(name.rstrip())

print(name.lstrip())

print(name.strip())
```

输出:

" chenzhuo"

"chenzhuo "

"chenzhuo"

### 练习2.8:

```
filename = 'python_notes.txt'

print(filename.removesuffix('.txt'))
```

输出:

python\_notes

## 练习2.9:

```
print(1+7)

print(14-6)

print(2*4)

print(24/3)
```

输出:

8

8

8

8

## 练习2.10:

```
n = 14

message = f"My favorite number is {n}."

print(message)
```

输出:

My favorite number is 14.

## 练习2.11:

```
#输出我最喜欢的数字

n = 14

message = f"My favorite number is {n}."
```



```
print(message)
```

输出：

My favorite number is 14.

**练习2.12：**在python终端会话中执行import this，并粗略地浏览一下其他指导原则。

- [第三部分 Codewars Kata挑战](#)

## 第一题 求离整数n最近的平方数

```
def nearest_sq(n):  
  
    # pass  
  
    r = int(n ** 0.5)  
  
    if r * r == n:  
  
        return n  
  
    x = (r+1)*(r+1)  
  
    if (x-n)<(n-r*r):  
  
        return x  
  
    else:  
  
        return (r*r)
```

输入：111

输出：121

## 第二题 弹跳的球

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

```

# your code

if h <= 0 or bounce >= 1 or window >= h or bounce <= 0:

    return -1

count = 1

while h * bounce > window:

    h *= bounce

    count += 2

return count

```

输入：10, 0.6, 10

输出：11

### 第三题 元音统计

```

def get_count(sentence):

    n = "aeiou"

    count = 0

    for char in sentence:

        if char in n:

            count += 1

    return count

```

输入：Sample tests

输出：3

### 第四题 偶数或者奇数

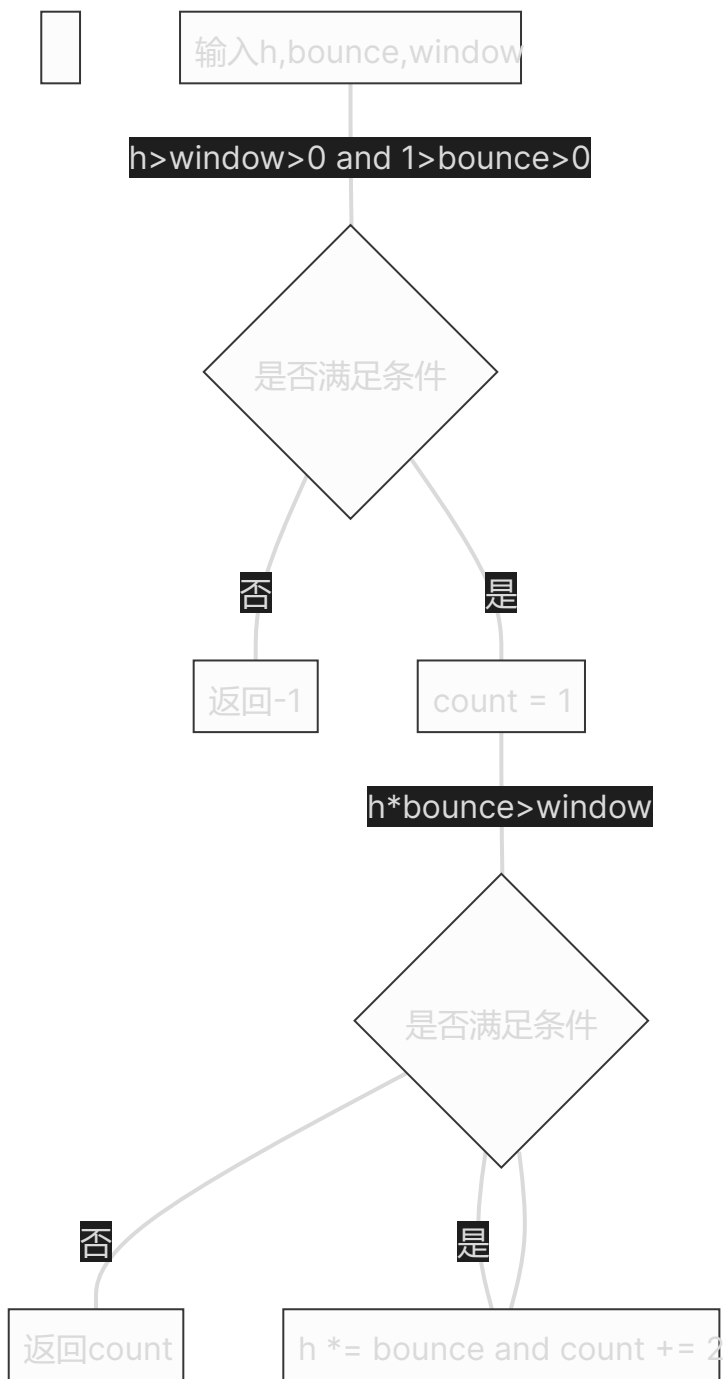
```
def even_or_odd(number):  
  
    a = "Even"  
  
    b = "Odd"  
  
    if number % 2 == 0:  
  
        return a  
  
    else:  
  
        return b
```

输入：2

输出：Even

- [第四部分 使用Mermaid绘制程序流程图](#)

## 第二题 弹跳的球流程图



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：

`"/Experiments/img/2023-07-26-22-48.png"` 未创建，点击以创建。

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：

"/Experiments/img/2023-07-26-22-52-20.png" 未创建，点击以创建。

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）、布尔值（bool）、空值（None）。可以执行算术运算、比较运算、逻辑运算、字符串操作、列表操作、元组操作、函数操作等。**

2. 为什么说Python中的变量都是标签？

**在python中每个变量都对应一个内存地址，这个地址可以用于访问和修改存储在其中的对象，因此，可以说Python中的变量都是标签。**

3. 有哪些方法可以提高Python代码的可读性？

**\*\*1.使用令人容易理解的函数名和变量名**

**2.使用注释**

**3.模块化**

4.避免全局变量和全局函数\*\*

## 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。

**在本次实验中，我学习了Python的简单数据类型和变量的使用。让我更加了解Python语言的语法和特性。python还有严格的格式要求。本次实验用到了VScode、Git、Markdown、Python等工具，让我更加熟练了这些软件的使用。**