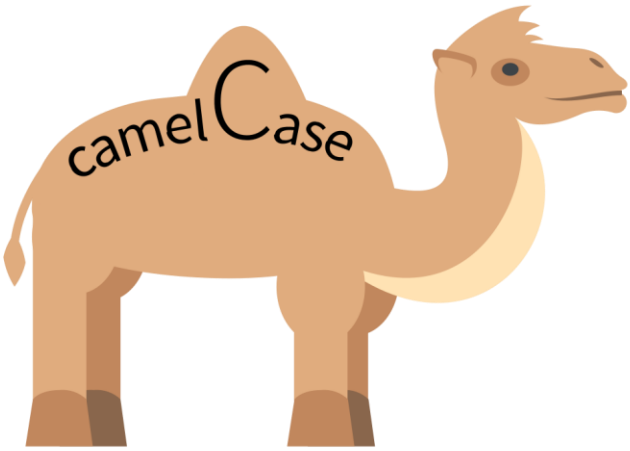
Bài 1. camelCase



Trong một số ngôn ngữ, người ta thường sử dụng cách viết camelCase (còn được gọi là “chữ hoa hỗn hợp”) cho tên biến khi các tên đó bao gồm nhiều từ, theo đó chữ cái đầu tiên của từ đầu tiên là chữ thường nhưng chữ cái đầu tiên của mỗi từ tiếp theo là chữ hoa. Ví dụ: trong khi một biến cho tên người dùng có thể được gọi name, một biến cho tên của người dùng có thể được gọi firstName và một biến cho tên ưa thích của người dùng (ví dụ: biệt danh) có thể được gọi preferredFirstName.

Ngược lại, Python khuyến nghị viết snake_case , theo đó các từ được phân tách bằng dấu gạch dưới (_), với tất cả các chữ cái viết thường. Ví dụ: các biến tương tự đó sẽ được gọi tương ứng name là , first_name và preferred_first_name, trong Python.

Trong tệp có tên camel.py, hãy triển khai một chương trình nhắc người dùng nhập vào tên của một biến ở dạng camelCase và xuất ra tên tương ứng ở dạng snake_case. Giả sử rằng đầu vào của người dùng thực sự sẽ ở dạng camelCase.

Gợi ý

Hãy nhớ lại rằng a str đi kèm với khá nhiều phương thức, theo docs.python.org/3/library/stdtypes.html#string-methods .

Giống như a list, a str là "có thể lặp lại", có nghĩa là bạn có thể lặp lại từng ký tự của nó trong một vòng lặp. Ví dụ: nếu s là a str, bạn có thể in từng ký tự của nó, mỗi lần một ký tự, với mã như:

```
for c in s:
    print(c, end="")
```

Ví dụ về kết quả chạy chương trình:

Chạy chương trình của bạn với python camel.py. Nhập name và nhấn Enter. Chương trình của bạn sẽ xuất ra: name

Chạy chương trình của bạn với python camel.py. Nhập firstName và nhấn Enter. Chương trình của bạn sẽ xuất ra: first_name

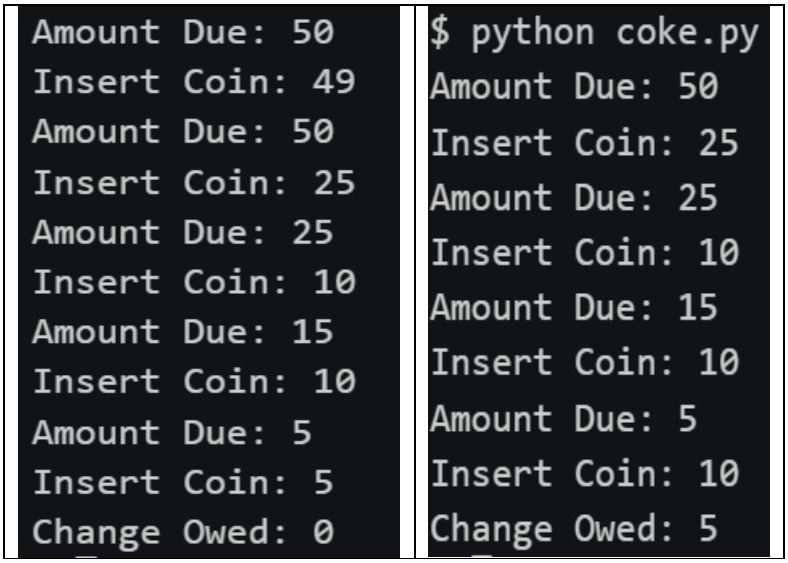
Chạy chương trình của bạn với python camel.py. Nhập preferredFirstName và nhấn Enter. Chương trình của bạn sẽ xuất ra: preferred_first_name

Bài 2. Coke Machine

Giả sử một chiếc máy bán chai Coca-Cola (Coke) với giá 50 xu và chỉ chấp nhận tiền xu có các mệnh giá sau: 25 xu, 10 xu và 5 xu.

Trong một tệp có tên coke.py, hãy triển khai một chương trình nhắc người dùng chèn từng đồng xu vào mỗi lần, thông báo cho người dùng về số tiền còn nợ. Sau khi người dùng đã nhập ít nhất 50 xu, hãy xuất ra số xu mà người dùng còn nợ. Giả sử rằng người dùng sẽ chỉ nhập số nguyên và bỏ qua bất kỳ số nguyên nào không phải là mệnh giá được chấp nhận.

Ví dụ về kết quả chạy chương trình:



Bài 3. twitter

Khi nhắn tin hoặc tweet, việc rút ngắn các từ để tiết kiệm thời gian hoặc không gian là điều bình thường, bằng cách bỏ qua các nguyên âm, giống như twitter ban đầu được gọi là *twttr* . Trong một tệp có tên twttr.py, hãy triển khai một chương trình nhắc người dùng nhập một str văn bản và sau đó xuất ra cùng một văn bản nhưng bỏ qua tất cả các nguyên âm (A, E, I, O và U), cho dù được nhập bằng chữ hoa hay chữ thường.

Gợi ý

Hãy nhớ lại rằng a str đi kèm với khá nhiều phương thức, theo docs.python.org/3/library/stdtypes.html#string-methods .

Giống như a list, a str là "có thể lặp lại", có nghĩa là bạn có thể lặp lại từng ký tự của nó trong một vòng lặp. Ví dụ: nếu s là a str, bạn có thể in từng ký tự của nó, mỗi lần một ký tự, với mã như:

```
for c in s:
    print(c, end="")
```

Ví dụ về kết quả chạy chương trình:

```
$ python twttr.py
Input: Twitter
Output: Twttr
$ python twttr.py
Input: What's your name?
Output: Wht's yr nm?
$ python twttr.py
Input: CS50
Output: CS50
```

Bài 4. Biển số

Ở Massachusetts, nơi có Đại học Harvard, bạn có thể yêu cầu biển số xe phù hợp cho ô tô của mình, với sự lựa chọn của bạn về các chữ cái và số thay vì các ký tự ngẫu nhiên. Tuy nhiên, trong số các yêu cầu là:

- “Tất cả các tấm biển số phải bắt đầu bằng ít nhất hai chữ cái.”
- “…biển số có thể chứa tối đa 6 ký tự (chữ cái hoặc số) và tối thiểu 2 ký tự.”
- “Không thể dùng số ở giữa một biển số; chúng phải nằm ở cuối. Ví dụ: AAA222 sẽ là một… tấm biển số có thể chấp nhận được; AAA22A sẽ không được chấp nhận. Số đầu tiên được sử dụng không được là '0'.”
- “Không được phép có dấu chấm, dấu cách hoặc dấu chấm câu.”

Trong plates.py, hãy triển khai một chương trình nhắc người dùng nhập vào một biển số rồi xuất ra Valid nếu đáp ứng tất cả các yêu cầu hoặc Invalid nếu không. Giả sử rằng mọi chữ cái trong dữ liệu nhập của người dùng sẽ là chữ hoa. Cấu trúc chương trình của bạn theo bên dưới, trong đó is_valid trả về True nếu s đáp ứng tất cả các yêu cầu và False nếu không. Giả sử đó s sẽ là một str. Bạn có thể triển khai các chức năng bổ sung để hàm is_valid gọi (ví dụ: một chức năng cho mỗi yêu cầu).

```
def main():
    plate = input("Plate: ")
    if is_valid(plate):
        print("Valid")
    else:
        print("Invalid")


def is_valid(s):
    ...


main()
```

Gợi ý

- Hãy nhớ lại rằng a str đi kèm với khá nhiều phương thức, theo docs.python.org/3/library/stdtypes.html#string-methods .
- Giống như a list, a str là một “chuỗi” (các ký tự), có nghĩa là nó có thể được “ cắt ” thành các chuỗi ngắn hơn với cú pháp như s[i:j]. Ví dụ: nếu s là "CS50", thì s[0:2] sẽ là "CS".

Ví dụ về kết quả chạy chương trình:

```
Plate: HELLO, WORLD
Invalid
$ python plates.py
Plate: GOODBYE
Invalid
$ python plates.py
Plate: CS50
Valid
$ python plates.py
Plate: CS05
Invalid
```

Bài 5. Nutrition

Cơ quan Quản lý Thực phẩm & Dược phẩm Hoa Kỳ (FDA) cung cấp các áp phích có thể tải xuống/in được "hiển thị thông tin dinh dưỡng cho 20 loại trái cây tươi được tiêu thụ thường xuyên nhất… tại Hoa Kỳ. Các cửa hàng bán lẻ được hoan nghênh tải xuống các áp phích, in, trưng bày và/hoặc phân phối chúng cho người tiêu dùng ở gần các loại thực phẩm có liên quan trong cửa hàng."

Trong tệp có tên nutrition.py, hãy triển khai một chương trình nhắc người tiêu dùng nhập một loại trái cây (không phân biệt chữ hoa chữ thường) và sau đó xuất ra số lượng calo trong một phần của loại trái cây đó, theo áp phích của FDA dành cho trái cây, cũng có sẵn dưới dạng văn bản . Bỏ viết hoa sang một bên, giả sử rằng người dùng sẽ nhập trái cây chính xác như được viết trên áp phích (ví dụ: strawberries, không phải strawberry). Bỏ qua bất kỳ đầu vào nào không phải là trái cây.

gợi ý

- Thay vì sử dụng câu điều kiện với 20 biểu thức Boolean, một biểu thức cho mỗi loại trái cây, tốt hơn nên sử dụng a dict để liên kết một loại trái cây với lượng calo của nó!
- Nếu k là a str và d là a dict, bạn có thể kiểm tra xem k có phải là một khóa trong d bằng mã như:

```
if k in d:
    ...
```

- Hãy cẩn thận để tạo ra lượng calo từ trái cây chứ không phải lượng calo từ chất béo!

Ví dụ về kết quả chạy chương trình:

```
$ python nutrition.py
Item: apple
Calories: 130
$ python nutrition.py
Item: banana
Calories: 110
$ python nutrition.py
Item: chocolate
$
```