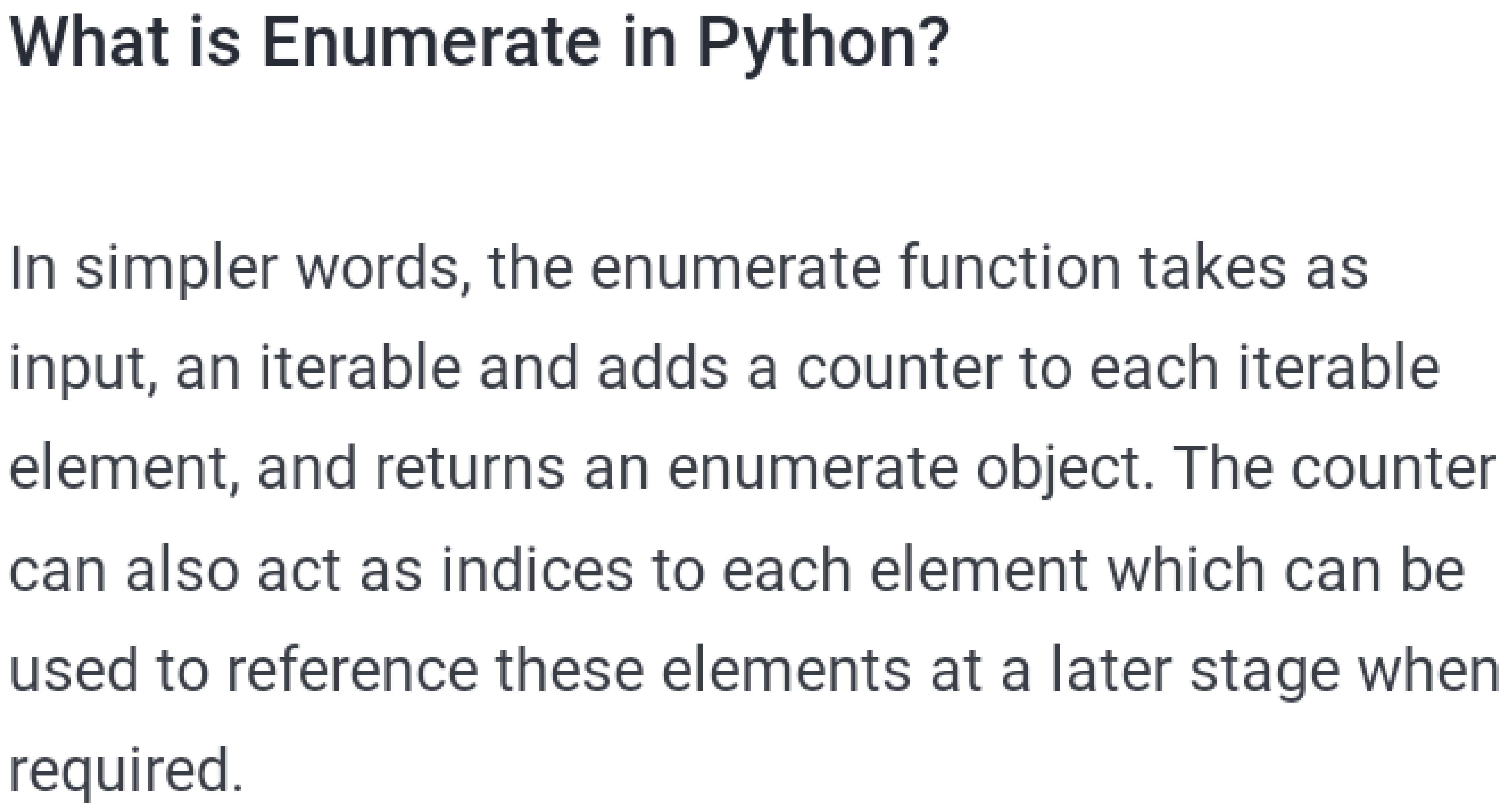
Python

Python Enumerate() Method Loop and List



What is Enumerate in Python?

In simpler words, the enumerate function takes as input, an iterable and adds a counter to each iterable element, and returns an enumerate object. The counter can also act as indices to each element which can be used to reference these elements at a later stage when required.

Python enumerate()

In this tutorial, we will learn about the Python enumerate() function with the help of examples.

The `enumerate()` function adds a counter to an iterable and returns it (the enumerate object).

Example

```
languages = ['Python', 'Java', 'JavaScript']

enumerate_prime = enumerate(languages)

# convert enumerate object to list
print(list(enumerate_prime))

# Output: [(0, 'Python'), (1, 'Java'), (2, 'JavaScript')]
```

Run Code »

Syntax of enumerate()

The syntax of `enumerate()` is:

```
enumerate(iterable, start=0)
```

enumerate() Arguments

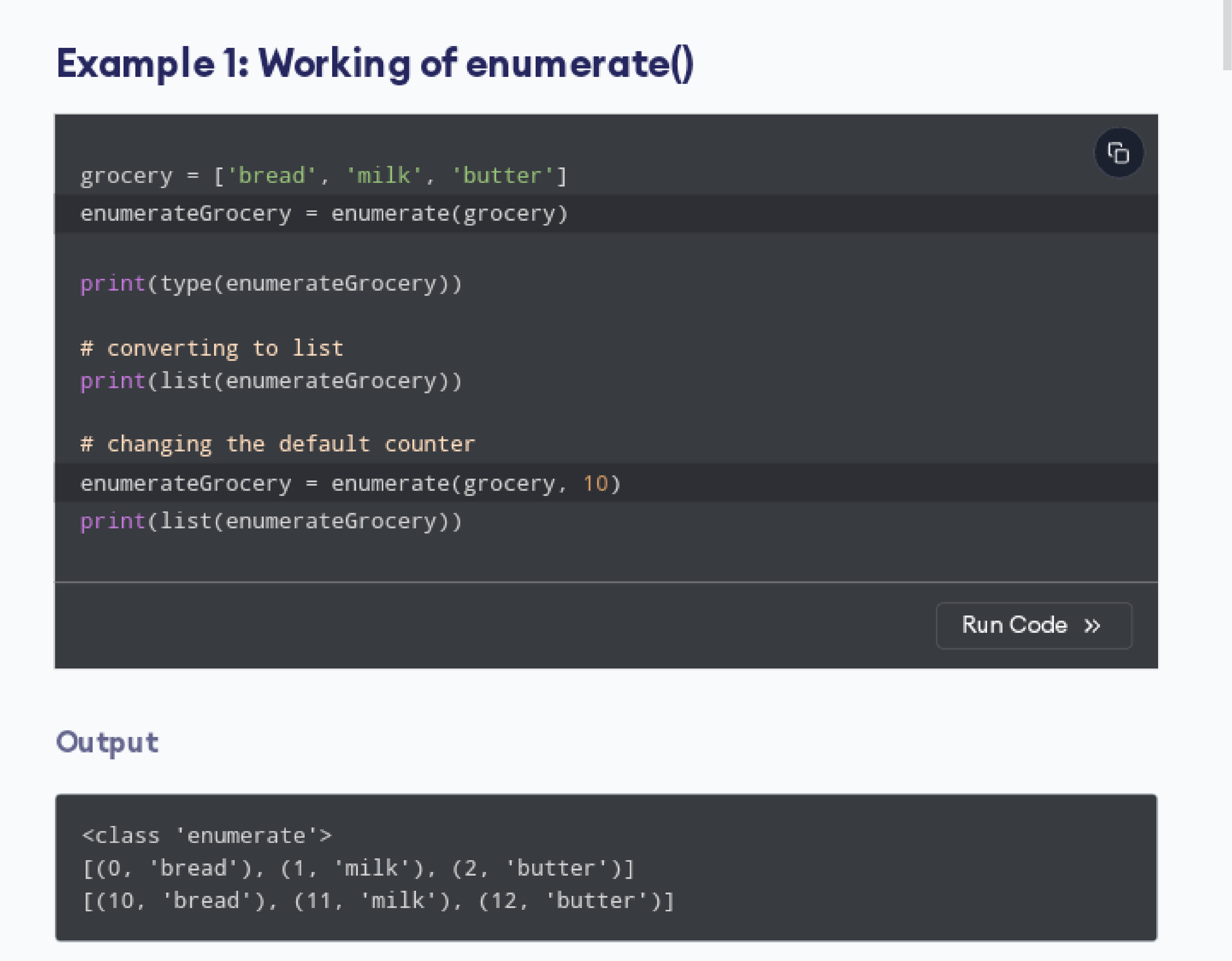
The `enumerate()` function takes two arguments:

- **iterable** - a sequence, an iterator, or objects that support iteration
- **start** (optional) - `enumerate()` starts counting from this number. If `start` is omitted, `0` is taken as `start`.

enumerate() Return Value

The `enumerate()` function adds counter to an iterable and returns it. The returned object is an enumerate object.

You can convert enumerate objects to list and tuple using list() and tuple() functions respectively.

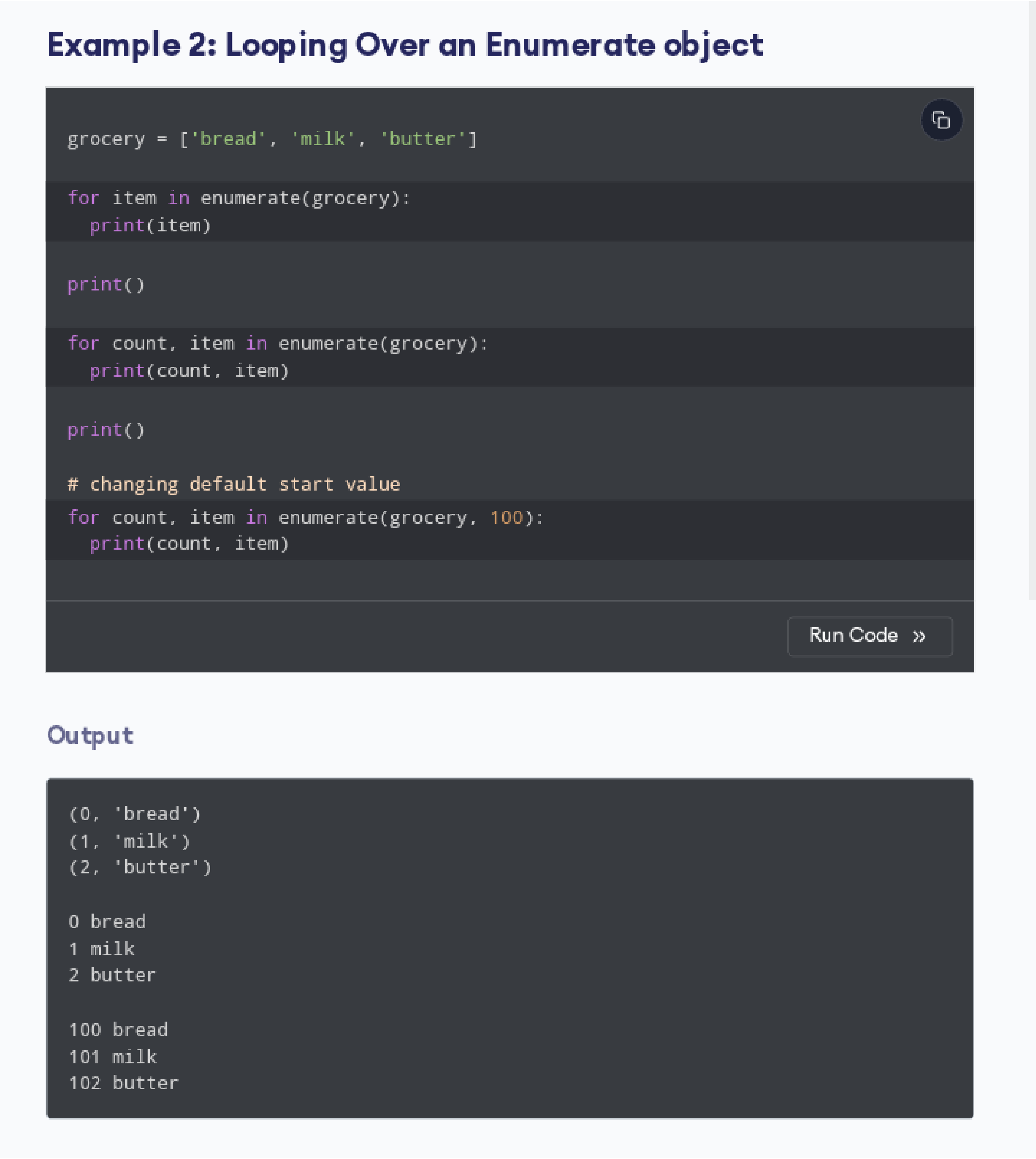


Example 1: Working of enumerate()

```
grocery = ['bread', 'milk', 'butter']
enumerateGrocery = enumerate(grocery)

print(type(enumerateGrocery))

# converting to list
print(list(enumerateGrocery))

# changing the default counter
enumerateGrocery = enumerate(grocery, 10)
print(list(enumerateGrocery))
```

Output

```
<class 'enumerate'>
[(0, 'bread'), (1, 'milk'), (2, 'butter')]
[(10, 'bread'), (11, 'milk'), (12, 'butter')]
```



Example 2: Looping Over an Enumerate object

```python
grocery = ['bread', 'milk', 'butter']

for item in enumerate(grocery):
  print(item)

print()

for count, item in enumerate(grocery):
  print(count, item)

print()

# changing default start value
for count, item in enumerate(grocery, 100):
  print(count, item)
```

Output

```
(0, 'bread')
(1, 'milk')
(2, 'butter')

0 bread
1 milk
2 butter

100 bread
101 milk
102 butter
```