Functions in Python

October 21, 2024

[17]:
```python
# Exercise 1

def custom_function(arg1, arg2=10, arg3=None):
    if arg3 is None:
        print(arg1 + arg2)
    else:
        print(arg1 * arg2 * arg3)

custom_function(5)
custom_function(5, 3)
custom_function(5, 3, 2)
```

```
15
8
30
```

[19]:
```python
# Exercise 2

def filter_long_strings(strings):
    return [string for string in strings if len(string) >= 5]

strings = ["apple", "pear", "banana", "kiwi"]
print(filter_long_strings(strings))
```

```
['apple', 'banana']
```

[21]:
```python
# Exercise 3

expression = "3 * 5 + 2"
result = eval(expression)
print(result)
```

```
17
```

[23]:
```python
# Exercise 4

def is_prime(num):
    if num < 2:
```

```
        return False
    for i in range(2, int(num**0.5) + 1):
        if num % i == 0:
            return False
    return True

numbers = [2, 3, 4, 5, 6, 7, 8, 9, 10]
primes = list(filter(is_prime, numbers))
print(primes)
```

```
[2, 3, 5, 7]
```

[25]:
```
# Exercise 5

strings = ["apple", "banana", "cherry"]
uppercase_strings = list(map(str.upper, strings))
print(uppercase_strings)
```

```
['APPLE', 'BANANA', 'CHERRY']
```

[27]:
```
# Exercise 6

strings = ["apple", "banana", "cherry"]
lengths = list(map(len, strings))
print(lengths)
```

```
[5, 6, 6]
```

[29]:
```
# Exercise 7

from functools import reduce

numbers = [1, 2, 3, 4, 5]
sum_of_elements = reduce(lambda x, y: x + y, numbers)
print(sum_of_elements)
```

```
15
```

[31]:
```
# Exercise 8

from functools import reduce

numbers = [1, 2, 3, 4, 5]
max_element = reduce(lambda x, y: x if x > y else y, numbers)
print(max_element)
```

```
5
```

[ ]: