

# "List Comprehensions" Exercise

---

Prerequisites:

- [Python Language Overview](#)
- [Lists](#)

## Learning Objectives

- Demystify the incredible powers of Python list comprehensions, a powerful data-processing technique
- Practice performing mapping and filtering operations on Python lists

## Setup

Create a new Python script called something like "total\_comprehension.py" somewhere on your computer, perhaps your Desktop.

Open that file with your text editor and place inside the following contents:

```
# total_comprehension.py

my_numbers = [1, 2, 3, 4, 5, 6, 7]
print("-----")
print("ORIGINAL LIST:", my_numbers)

print("-----")
print("TOTAL COMPREHENSION...")

# TODO: write python code here
```

From within your "base" environment, or any other Anaconda virtual environment, run the script:

```
python total_comprehension.py
```

## Instructions

Write Python code in the "total\_comprehension.py" file which will use the filtering and mapping capabilities of list comprehensions to transform the provided list (i.e. `my_numbers`) in each of the following ways:

- Use mapping capabilities to multiply each number by 100 (e.g. `[100, 200, 300, 400, 500, 600, 700]`)
- Use filtering capabilities to return only the numbers greater than three (e.g. `[4, 5, 6, 7]`)

- Use filtering capabilities to return only the numbers greater than eight (e.g. `[]`)
- Use mapping and filtering capabilities to return only the numbers greater than three, each multiplied by 100 (e.g. `[400, 500, 600, 700]`)

Example final output:

```
-----  
ORIGINAL LIST: [1, 2, 3, 4, 5, 6, 7]  
-----  
TOTAL COMPREHENSION...  
-----  
MAPPED LIST: [100, 200, 300, 400, 500, 600, 700]  
-----  
FILTERED LIST W/ MATCHES: [4, 5, 6, 7]  
-----  
FILTERED LIST W/O MATCHES: []  
-----  
MAPPED AND FILTERED LIST: [400, 500, 600, 700]
```

## Solution