

4-1. Pizzas: Think of at least three kinds of your favorite pizza. Store these pizza names in a list, and then use a for loop to print the name of each pizza.

- Modify your for loop to print a sentence using the name of the pizza, instead of printing just the name of the pizza. For each pizza, you should have one line of output containing a simple statement like I like pepperoni pizza.
- Add a line at the end of your program, outside the for loop, that states how much you like pizza. The output should consist of three or more lines about the kinds of pizza you like and then an additional sentence, such as I really love pizza!

```
In [ ]: pizzas = ['pepperoni', 'mushroom', 'cheese']
```

```
for pizza in pizzas:
    print(pizza)
    print(f'I like {pizza}')

print(f"I like pizza like
      1){pizzas[0]}
      2){pizzas[1]}
      3){pizzas[2]}
I really love Pizza
      ")
```

```
pepperoni
I like pepperoni
mushroom
I like mushroom
cheese
I like cheese
I like pizza like
      1)pepperoni
      2)mushroom
      3)cheese
I really love Pizza
```

4-2. Animals: Think of at least three different animals that have a common characteristic. Store the names of these animals in a list, and then use a for loop to print out the name of each animal.

- Modify your program to print a statement about each animal, such as A dog would make a great pet.
- Add a line at the end of your program, stating what these animals have in common. You could print a sentence, such as Any of these animals would make a great pet!

```
In [ ]: animals = ['Tiger', 'Wolf', 'panther']
```

```
for animal in animals:
    print(animal)
    print(f'A {animal} has a tale.')

print('They all have a tail.')
```

Tiger
A Tiger has a tale.
Wolf
A Wolf has a tale.
panther
A panther has a tale.
They all have a tail.

4-3. Counting to Twenty: Use a for loop to print the numbers from 1 to 20, inclusive.

```
In [ ]: for num in range(1,21):  
        print(num)
```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

4-4. One Million: Make a list of the numbers from one to one million, and then use a for loop to print the numbers. (If the output is taking too long, stop it by pressing CTRL-C or by closing the output window.)

```
In [ ]: num = list(range(1,1000001))  
        for value in num:  
            print(value)
```

4-5. Summing a Million: Make a list of the numbers from one to one million, and then use min() and max() to make sure your list actually starts at one and ends at one million. Also, use the sum() function to see how quickly Python can add a million numbers.

```
In [ ]: num = list(range(1,1000001))  
  
        print(min(num))  
  
        print(max(num))  
  
        sum(num)
```

```
Out[ ]: 1  
1000000  
500000500000
```

4-6. Odd Numbers: Use the third argument of the range() function to make a list of the odd numbers from 1 to 20. Use a for loop to print each number.

```
In [ ]: num = list(range(1,21,2))

for number in num:
    print(number)
```

```
1
3
5
7
9
11
13
15
17
19
```

4-7. Threes: Make a list of the multiples of 3, from 3 to 30. Use a for loop to print the numbers in your list.

```
In [ ]: numbers = list(range(3,31,3))

for number in numbers:
    print(number)
```

```
3
6
9
12
15
18
21
24
27
30
```

4-8. Cubes: A number raised to the third power is called a cube. For example, the cube of 2 is written as 2^3 in Python. Make a list of the first 10 cubes (that is, the cube of each integer from 1 through 10), and use a for loop to print out the value of each cube.

4-9. Cube Comprehension: Use a list comprehension to generate a list of the first 10 cubes.

```
In [ ]: #4.8
nums = []

for value in range(1,11):
    value = value**2
    nums.append(value)
    print(value)

print('\n')
#4.9
nums = [value**2 for value in range(1,11)]
for number in nums:
    print(number)
```

```

1
4
9
16
25
36
49
64
81
100

```

```

1
4
9
16
25
36
49
64
81
100

```

4-10. Slices: Using one of the programs you wrote in this chapter, add several lines to the end of the program that do the following:

- Print the message The first three items in the list are:. Then use a slice to print the first three items from that program's list.
- Print the message Three items from the middle of the list are:. Then use a slice to print three items from the middle of the list.
- Print the message The last three items in the list are:. Then use a slice to print the last three items in the list

```

In [ ]: numbers =[]

for value in range(1,11):
    numbers.append(value**2)
for number in numbers:
    print(number)

print(f'The first three items in the list are:{numbers[:3]} ')

print(f'Three items from the middle of the list are:{numbers[3:6]} ')

print(f'The last three items in the list are:{numbers[-3:]} ')

```

```

1
4
9
16
25
36
49
64
81
100
The first three items in the list are:[1, 4, 9]
Three items from the middle of the list are:[16, 25, 36]
The last three items in the list are:[64, 81, 100]

```

4-11. My Pizzas, Your Pizzas: Start with your program from Exercise 4-1 (page 56). Make a copy of the list of pizzas, and call it `friend_pizzas`. Then, do the following:

- Add a new pizza to the original list.
- Add a different pizza to the list `friend_pizzas`.
- Prove that you have two separate lists. Print the message `My favorite pizzas are;` and then use a `for` loop to print the first list. Print the message `My friend's favorite pizzas are;` and then use a `for` loop to print the second list.

Make sure each new pizza is stored in the appropriate list

```
In [ ]: pizzas = ['pepperoni', 'mushroom', 'cheese']

for pizza in pizzas:
    print(pizza)
    print(f'I like {pizza}')

print(f"""I like pizza like
      1){pizzas[0]}
      2){pizzas[1]}
      3){pizzas[2]}
I really love Pizza
""")

friend_pizzas = pizzas[:]

pizzas.append('green chillies')
friend_pizzas.append('mutton')

for pizza in pizzas:
    print(f'Favourite toppings:{pizza}')

print('\n')
for pizza in friend_pizzas:
    print(f'My friend's favourite pizzas are:{pizza}')
```

```

pepperoni
I like pepperoni
mushroom
I like mushroom
cheese
I like cheese
I like pizza like
    1)pepperoni
    2)mushroom
    3)cheese
I really love Pizza

```

```

Favourite toppings:pepperoni
Favourite toppings:mushroom
Favourite toppings:cheese
Favourite toppings:green chilies

```

```

My friend's favourite pizzas are:pepperoni
My friend's favourite pizzas are:mushroom
My friend's favourite pizzas are:cheese
My friend's favourite pizzas are:mutton

```

4-12. More Loops: All versions of foods.py in this section have avoided using for loops when printing, to save space. Choose a version of foods.py, and write two for loops to print each list of foods.

```

In [ ]: my_foods = ['mutton', 'mutton biriyani', 'chichen butter-fry']
friend_foods = my_foods[:]
for food in my_foods:
    print("My favorite foods are:")
    print(food)
for food in friend_foods:
    print("My friend's favorite foods are:")
    print(food)

```

```

My favorite foods are:
mutton
My favorite foods are:
mutton biriyani
My favorite foods are:
chichen butter-fry
My friend's favorite foods are:
mutton
My friend's favorite foods are:
mutton biriyani
My friend's favorite foods are:
chichen butter-fry

```

4-13. Buffet: A buffet-style restaurant offers only five basic foods. Think of five simple foods, and store them in a tuple.

- Use a for loop to print each food the restaurant offers.
- Try to modify one of the items, and make sure that Python rejects the change.
- The restaurant changes its menu, replacing two of the items with different foods. Add a line that rewrites the tuple, and then use a for loop to print each of the items on the revised menu.

```
In [ ]: buffet_items = ('cake','mutton','chilles','eggs','tea')

buffet_items[1] = 'juice'
```

```
-----
TypeError                                Traceback (most recent call last)
Cell In[24], line 3
      1 buffet_items = ('cake','mutton','chilles','eggs','tea')
----> 3 buffet_items[1] = 'juice'

TypeError: 'tuple' object does not support item assignment
```

```
In [ ]: buffet_items = ('cake','mutton','chilles','eggs','tea')

buffet_items = ('toast','mutton','chicken','omlet','jam')

for item in buffet_items:
    print(item)
```

```
toast
mutton
chicken
omlet
jam
```