

▼ Exercises

Answer the questions or complete the tasks outlined in bold below, use the specific method described if applicable.

**** What is 7 to the power of 4? ****

```
7**4
```

```
2401
```

**** Split this string: ****

```
s = "Hi there Sam!"
```

****into a list.****

```
s= "Hi there Sam!"  
print(list(s.split(" ")))
```

```
['Hi', 'there', 'Sam!']
```

**** Given the variables: ****

```
planet = "Earth"  
diameter = 12742
```

**** Use .format() to print the following string: ****

```
The diameter of Earth is 12742 kilometers.
```

```
planet = "Earth"  
diameter = 12742  
print("The diameter of {} is {} kilometers.".format(planet,diameter))
```

```
The diameter of Earth is 12742 kilometers.
```

**** Given this nested list, use indexing to grab the word "hello" ****

```
lst = [1,2,[3,4],[5,[100,200,['hello']],23,11],1,7]
```

```
print(lst[3][1][2][0])
```

```
hello
```

**** Given this nest dictionary grab the word "hello". Be prepared, this will be annoying/tricky ****

```
d = {'k1':[1,2,3,{'tricky':['oh','man','inception',{'target':[1,2,3,'hello']}]}]}
```

```
print(d['k1'][3]['tricky'][3]['target'][3])
```

```
hello
```

**** What is the main difference between a tuple and a list? ****

List is mutable while tuple is immutable and Lists consume more memory when compare to tuple

```
lst = [1,2,3,5,7]
lst[4]= 100
print(lst)
```

```
[1, 2, 3, 5, 100]
```

```
tuple1 = (1,2,3,5,7)
tuple1[1]=100 #As tuple is immutable
print(tuple1)
```

**** Create a function that grabs the email website domain from a string in the form: ****

```
user@domain.com
```

So for example, passing "[user@domain.com](#)" would return: domain.com

```
def id(mail):
    result = mail[mail.rfind('@')+1:]
    return result
```

```
print(id("user@domain.com"))
```

```
domain.com
```

**** Create a basic function that returns True if the word 'dog' is contained in the input string. Don't worry about edge cases like a punctuation being attached to the word dog, but do account for capitalization. ****

```
def function(s):
    if "dog" in s:
        return True
    return False
```

```
print(function("The dog is in the kettle"))
```

True

**** Create a function that counts the number of times the word "dog" occurs in a string. Again ignore edge cases. ****

```
def cnt(s):  
    s.lower()  
    return s.count("dog")
```

```
print(cnt("The dog is the most favorite pet for humans and the dog is in more home"))
```

2

▼ Problem

**You are driving a little too fast, and a police officer stops you. Write a function to return one of 3 possible results: "No ticket", "Small ticket", or "Big Ticket". If your speed is 60 or less, the result is "No Ticket". If speed is between 61 and 80 inclusive, the result is "Small Ticket". If speed is 81 or more, the result is "Big Ticket". Unless it is your birthday (encoded as a boolean value in the parameters of the function) -- on your birthday, your speed can be 5 higher in all cases. **

```
def caught_speeding(speed, is_birthday):  
  
    if is_birthday:  
        speeding = speed - 5  
    else:  
        speeding = speed  
  
    if speeding > 80:  
        return 'Big Ticket'  
    elif speeding > 60:  
        return 'Small Ticket'  
    else:  
        return 'No Ticket'
```

```
print(caught_speeding(99,1))
```

Big Ticket

```
print(caught_speeding(70,0))
```

Small Ticket

Create an employee list with basic salary values(at least 5 values for 5 employees) and using a for loop retrieve each employee salary and calculate total salary expenditure.

```
l=[10000,20000,30000,40000,90000]  
result=0  
for i in l:
```

```
result+=i  
print(result)
```

1999999

Create two dictionaries in Python:

First one to contain fields as Empid, Empname, Basicpay

Second dictionary to contain fields as DeptName, DeptId.

Combine both dictionaries.

```
dic = { "Empid" : 207,"Empname" : "thaarani","Basicpay" : 900000}  
dic1 = {"DeptName" : "Software Developer","DeptId" : 1001}  
dic.update(dic1)  
print(dic)
```

```
{'Empid': 207, 'Empname': 'thaarani', 'Basicpay': 900000, 'DeptName': 'Software Developer', 'DeptId': 1001}
```

[Colab paid products](#) - [Cancel contracts here](#)

✓ 0s completed at 11:25 AM

