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?

In [ ]:
```
base = 7
exponent = 4
result = 1
for exponent in range(exponent, 0, -1):
    result *= base
print(str(result))
```

Out[ ]:
```
2401
```

Split this string:

Split this string:

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

into a list.

In [ ]:
```
txt = input()
x = txt.split()
print(x)
```

Out[ ]:
```
['Hi', 'there', 'dad!']
```

Given the variables:

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

Use .format() to print the following string:

In [ ]:
```
planet = "Earth"
diameter = 12742
print( 'The diameter of {} is {} kilometers.' .format(planet,diameter));
```

Out[ ]:
```
The diameter of Earth is 12742 kilometers.
```

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

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

In [ ]:
```
lst = [1,2,[3,4],[5,[100,200,['hello']],23,11],1,7]
a=lst[3][1][2];
print(a)
```

Out[ ]:
```
'hello'
```

Given this nest dictionary grab the word “hello”. Be prepared, this annoying/tricky

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

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

Out[ ]: 'hello'

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

In [ ]:
```
The list is dynamic whereas tuple has static characteristics
The  list is mutable where as tuple is immutable
The list is denoted by square brackets and tuple is denoted by paranthesis.
```

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

In [ ]:
```
def domainGet(email):
    print("Your domain is: " + email.split('@')[-1])
email = input("Please enter your email: >")domainGet(email)
```

In [ ]:
```
domainGet('user@domain.com')
```

Out[ ]: ‘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.

In [ ]:
```
def findDog(st):
    if 'dog' in st.lower():
        print("True")
    else:
        print("False")
st = "Is there a dog here?"findDog(st)
```

Out[ ]: True

Create a function that counts the number of times the word "dog" occurs in astring.Again ignore edge case

In [ ]:
```
value = 'This dog runs faster than the other dog dude!';
def countdogs(value):
    count = 0
    for word in value.lower().split():
        if word == 'dog' or word == 'dogs':
           count = count + 1
           print(count)
countdogs(value)
```

Out[ ]: 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.

In [ ]:
```
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'
```

In []:
```
caught_speeding(81,True)
```

Out[ ]: 'Big Ticket'

In []:
```
caught_speeding(81,False)
```

Out[ ]: 'Small Ticket'

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

In [ ]:

```
keys={"empid":"101","empname":"XXXX","Basicpay":"100000"}
values={"DeptName":"Design","DeptId":"1012"}
print({**keys, **values})
```

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.