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 [3]:

print(7**\*\***4)

2401

\*\*Split this string:

s = "Hi there Sam!"

into a list

In [17]:

s **=** "Hi there Sam!" d**=**s**.**split(' ') print(d)

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

In [38]:

*# # method 1 -- user input*

*# s=input("enter the input :").split() #input-- Hi there dad! # print(s)*

*#method 2 -- manual input* s**=** "Hi there dad!" x**=**s**.**split()

print(x)

['Hi', 'there', 'dad!']

\*\*Given the variables:

planet = "Earth" diameter = 12742

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

The diameter of Earth is 12742 kilometers.

In [19]:

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"

In [39]:

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

In [40]:

x**=**lst[3][1][2]

print(**\***x)

hello

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

In [41]:

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

In [42]:

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

hello

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

# Tuple:

-- Enclosed with round brackets.

-- immutable

# List:

-- Enclosed with square brackets.

-- mutable

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

[user@domain.com](mailto:user@domain.com)

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

In [43]:

d**=**["user@domain.com"](mailto:user@domain.com)

**def** fun(d): a**=**""

res**= False for** i **in** d:

**if**(res): a**+=**i

**if**(i**==**"@"): res **= True**

**return** a

print(fun(d))

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 [46]:

**def** word\_checker(sentence,word): s**=**sentence**.**split(" ")

**for** i **in** s:

**if**(i**==**word): **return True**

**return False**

sentence**=**"My Dog name is Dora" Target\_word **=** "Dog" **if**(word\_checker(sentence,Target\_word)):

print("True")

**else**:

print("False")

True

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

In [48]:

**def** word\_checker(sentence, Target\_word): a **=** sentence**.**split(" ")

c **=** 0

**for** i **in** range(0, len(a)):

**if** (Target\_word**==** a[i]): c **=** c**+** 1

**return** c

sentence**=**"Dog is a pet and i love Dog" Target\_world**=**"Dog" print(word\_checker(sentence,Target\_word))

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 [53]:

**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(200,**True**))

Big Ticket

In [54]:

**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(71,**False**))

Small Ticket

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

In [55]:

employees **=** ["Dharshan","Pavi", "Abisha", "Dhayalini", "Savitha"] salary**=**{}

**for** emp **in** employees:

amount**=**int(input(f'Enter salary for {emp}: ')) salary[emp]**=**amount

print('Total salary ', sum(salary**.**values()))

Enter salary for Dharshan: 10000 Enter salary for Pavi: 10000 Enter salary for Abisha: 1200 Enter salary for Dhayalini: 3030 Enter salary for Savitha: 3333 Total salary 27563

\*\*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.

In [60]:

dict1**=**{'Empid' : 401, 'Empname':"Dharshan","Basicpay": 10000} dict2**=** {'DeptName': "Excecutive", 'Deptid': 12345}

**def** checker(dict1,dict2): result**=**dict1**.**copy() result**.**update(dict2) **return** result

result**=**checker(dict1,dict2) print(z)

{'Empid': 401, 'Empname': 'Dharshan', 'Basicpay': 10000, 'DeptName': 'Excecutive', 'Deptid': 12345}

In [ ]: