

Roll No : 191349

Name : Shruti S

Project Name : Estimate The Crop Yield Using Data Analytics

Exercises

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

1. What is 7 to the power of 4?

```
In [1]: 7**4
```

```
Out[1]: 2401
```

Split this string:

s = "Hi there Sam!" into a list

```
In [3]: s = "Hi there Sam!"
print(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.

```
In [4]: planet = "Earth"
diameter = 12742
print(f'The diameter of {planet} is {diameter} kilometers'.format(planet, diameter))
```

The diameter of Earth is 12742 kilometers

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

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

['hello']
```

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

```
In [6]: 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?

1. Tuples are immutable in nature whereas list is mutable
2. Tuples are enclosed within () whereas list is enclosed within []
3. Tuples take up less memory space when compared to lists

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 [7]: def domain_finder(email):
        return email.split("@")[-1]
        print(domain_finder("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.

```
In [8]: def find_dog(s):
        st = s.lower()
        if 'dog' in st:
            return True
        else:
            return False
        s = "Do you have a dog?"
        if find_dog(s):
            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 [9]: def dog_count(s):
        st = s.lower()
        count = 0
        for i in st.split():
            if i == 'dog' or i == 'dogs':
                count += 1
        return count
        value = 'This dog runs faster than the other dog dude!';
        print(dog_count(value))
```

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 [10]: def speed_check(speed, is_bday):
        if is_bday:
            speeding = speed - 5
        else:
            speeding = speed
        if speeding <= 60:
            print("No ticket")
        elif speeding >= 61 and speeding <= 80:
            print("Small ticket")
        else:
            print("Big ticket")
        speed_check(45, False)
```

No 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.

```
In [13]: emp_salary = {}
        emp_names = ["Edward", "Raj", "David", "Harry", "Oman"]
        for i in emp_names:
            emp_salary[i] = int(input())
        s = 0
        for k in emp_salary.values():
            s += k
        print("The total salary expenditure is: " + str(s))
```

2593

2453

2054

2053

5322

The total salary expenditure is: 14475

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 [14]: dict1 = {'Empid' : 1, 'Empname' : 'Harshu', 'Basicpay': 1000}
dict2 = {'DeptName': 'IT', 'DeptId': '2' }
res = {**dict1, **dict2}
print(res)

{'Empid': 1, 'Empname': 'Harshu', 'Basicpay': 1000, 'DeptName': 'IT', 'DeptId': '2'}
```