Assignment – 3

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?**

[1]
```
7**4
```
✓ 0.8s

```
2401
```

[2]
```
print("hello");
```
✓ 0.1s

```
hello
```

[3]
```
print("Karthikeyan")
```
✓ 0.7s

```
Karthikeyan
```

** Split this string:**

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

**into a list. **

⊗ 4 ⚠ 1

```python
s=['Hi','There','Sam!']
print(s)
```

[5] ✓ 0.1s

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

```python
k=['Hi','There','Dad!']
print(k)
```

[6] ✓ 0.1s

```
['Hi', 'There', 'Dad!']
```

** Given the variables:**

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

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

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

```python
p="Earth"
d=12742
print('The diameter of {} is {} kilometers.' .format(p,d))
```

[7] ✓ 0.1s

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

⊗ 4 ⚠ 1

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

[8] ✓ 0.1s

```
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]
```

[9] ✓ 0.1s

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

[14] ✓ 0.1s

```
['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']}]}]}
```

[15] ✓ 0.1s

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

⊗ 4 ⚠ 1

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

[18] ✓ 0.1s

```
hello
```

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

```
ıble. Therefore, it is possible to change a list but not a tuple. The contents of a tuple cannot cha
```

[ ]

** 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 domainMe(email):
    return email.split('@')[-1]
```

[19] ✓ 0.1s

```
domainMe('user@domain.com')
```

[21] ✓ 0.1s

```
'domain.com'
```

⊗ 4 ⚠ 1

** Create a basic function that returns True if the word 'dog' is contained in the input string. Don't worry about edge cas
capitalization. **

```
def searchDog(s):
  if 'dog' in s.lower():
    print("true")
  else:
    print("false")
    st = "Dog ia a pet animal!"
    searchDog(s)
```

[24] ✓ 0.1s

```
searchDog('Dog ia a pet animal!')
```

[25] ✓ 0.2s

```
true
```

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

```
string = 'The kept barking all night, The dogs run together! ';
def countdogs(string):
      count=0
      for word in string.lower().split():
            if word == 'dog' or word == 'dogs':
                  count = count+1
                  print(count)
countdogs(string)
```

[27] ✓ 0.9s

```
1
```

⊗ 4 ⚠ 1

```
seq=['soup','dog','saled','cat','great']
```

[28] ✓ 0.1s

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 tick
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 T
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'
```

[26] ✓ 0.1s

```
caught_speeding(90,False)
```

[29] ✓ 0.1s

```
'Big Ticket'
```

```
caught_speeding(78,True)
```

[30] ✓ 0.1s

```
'Small Ticket'
```

Create an employee list with basic salary values(at least 5 values for 5 employees) and using a f

```
emp_names=["abc","def","ghi","jkl","mno","pqr"]
emp_salaries={}
for employee in emp_names:
        while True:
                try:
                        emp_salaries[employee]=int(input{employee}'s' salary )
                        break;
                except valueError:
                        print("invalid input")
print("employee_salaries")
total=sum(emp_salaries.value())
print("total")
```

⊗ 4 ⚠ 1

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.

```
def Merge(d1, d2):
        output = {**d1, **d2}
        return output
d1 = {'Empid': 1, 'Empname': 'kishore', 'Basicpay': 5000}
d2 = {'DeptName':'BME', 'DeptId': 5}
d3 = Merge(d1, d2)
print(d3)
```

[40] ✓ 0.1s

```
{'Empid': 1, 'Empname': 'kishore', 'Basicpay': 5000, 'DeptName': 'BME', 'DeptId': 5}
```