What is 7 to the power of 4?

```
print(7**4)
```

```
2401
```

Split this string: s = "Hi there Sam!"

```
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.

```
planet="Earth"
diameter=12742

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

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

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])
```

```
['hello']
```

```
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?

```
The list is mutable, whereas the tuple in immutable
```

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 domain(email):
 return email.split('@')[-1]
domain("user@domain.com")
```

```
{"type":"string"}
```

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 check(word):
 for i in word.split():
   if i == "dog":
      return True
      return False
check("I love my dog")
```

```
True
```

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

```
def wordCount(word):
 count = 0
 for i in word.split():
   if i == 'dog':
     count += 1
     print("Dog repeated "+str(count)+" times")

wordCount("dog dog dog cat lion Dog DOG")
```

```
Dog repeated 1 times
Dog repeated 2 times
Dog repeated 3 times
```

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'


caught_speeding(87,False)
```

```
{"type":"string"}
```

```
caught_speeding(65,False)
```

```
{"type":"string"}
```

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.

```
e = [["emp1",[1325,4975,8660,6037,8308]],
     ["emp2",[5678,5847,3375,3415,6640]],
     ["emp3",[1806,5144,8340,1253,6667]],
     ["emp4",[8381,7678,8599,3629,9345]],
     ["emp5",[7066,1307,3023,6295,2046]]]

total = 0
for i in range(len(e)):
  for j in e[i][1]:
    total += j
    print(str(e[i][0])+" earns "+str(total))
```

```
emp1 earns 1325
emp1 earns 6300
emp1 earns 14960
emp1 earns 20997
emp1 earns 29305
emp2 earns 34983
emp2 earns 40830
emp2 earns 44205
emp2 earns 47620
emp2 earns 54260
emp3 earns 56066
emp3 earns 61210
emp3 earns 69550
emp3 earns 70803
emp3 earns 77470
emp4 earns 85851
emp4 earns 93529
emp4 earns 102128
emp4 earns 105757
emp4 earns 115102
emp5 earns 122168
emp5 earns 123475
emp5 earns 126498
emp5 earns 132793
emp5 earns 134839
```

```
print(e)
```

```
[['emp1', [1325, 4975, 8660, 6037, 8308]], ['emp2', [5678, 5847, 3375,
3415, 6640]], ['emp3', [1806, 5144, 8340, 1253, 6667]], ['emp4',
```

```
[8381, 7678, 8599, 3629, 9345]], ['emp5', [7066, 1307, 3023, 6295,
2046]]]
```

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.

```
dic1 = {"Empid": 123, "Empname": "dan", "Basicpay": 7500}
dic2 = {"DeptName": "CSE", "Deptid": 4500}
dic3 = {**dic1, **dic2}
print(dic3)
```

```
{'Empid': 123, 'Empname': 'dan', 'Basicpay': 7500, 'DeptName': 'CSE',
'Deptid': 4500}
```