1. What is 7 to the power of 4?

```
7**4
```

↳ 2401

2. Split this string: s = "Hi there Sam!" into a list and replace Sam with dad

```
s = "Hi there Sam!"
lst = s.split()
lst[2] = "dad!"
lst
#
```

↳ ['Hi', 'there', 'dad!']

3. 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 {p} is {d}".format(p=planet, d=diameter))
```

↳ The diameter of Earth is 12742

4. Given this nested list, use indexing to grab the word "hello" lst = [1,2,[3,4],[5,[100,200, ['hello']],23,11],1,7]

```
lst = [1,2,[3,4],[5,[100,200,['hello']],23,11],1,7]
lst[3][1][2][0]
```

↳ 'hello'

5. Given this nested 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']}]}]}

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

↳ 'hello'

6. What is the main difference between a tuple and a list? Intentionally generate an error by

```
a_list = [1,2,3,4,6] #list - is mutable
b_list = (1,2,3,4,6) #tuple - is immutable

#my error:
b_list[1] = 0
```

```
--------------------------------------------------------------------
TypeError                                   Traceback (most recent call last)
<ipython-input-23-f462cb52fa31> in <module>()
      3
      4 #my error:
----> 5 b_list[1] = 0

TypeError: 'tuple' object does not support item assignment
```

SEARCH STACK OVERFLOW

7. Create a function that grabs the email website domain from a string in the form:
   super_user@ee.pw.edu.pl So for example, passing "super_user@ee.pw.edu.pl" would
   return: ee.pw.edu.pl

```
form = "super_user@ee.pw.edu.pl"
form.split("@")[1] #you cannot have two @ in e-mail
```

```
'ee.pw.edu.pl'
```

8. Create a basic function that returns True if the word 'car' is contained in the input string.
   Don't worry about edge cases like a punctuation being attached to the word car, but do
   account for capitalization.

```
def BasicFunction(word):

  if "CAR" in word.upper().split() :
    return True
  else:
    return False

st = ["I have a car .",  "I have a Car.", " I have a bike .", "I have a CAr .", "I have ac

for i in st:
  print({BasicFunction(i)})
```

```
{True}
{False}
{False}
{True}
{False}
```

9. Create a function that counts the number of times the word "car" occurs in a string. Again ignore edge cases. Example: countCar('This car runs faster than the other car dude!') 2

```python
def CountCar(word):
  nr = 0
  words = word.upper().split()
  for j in words:
    if j == "CAR":
      nr += 1
  return nr
```

```python
CountCar("This car runs faster than the other car dude!")
```

    2

10. Use lambda expressions and the filter() function to filter out words from a list that don't start with the letter 's'. For example: seq = ['soup','dog','salad','cat','great'] should be filtered down to: ['soup','salad']

```python
seq = ['soup','dog','salad','cat','great']
lst = []
for i in seq:
  if i[0] == 's':
    lst.append(i)
lst
```

    ['soup', 'salad']

11. Final 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): pass

caught_speeding(81,True) 'Small Ticket'

caught_speeding(81,False) 'Big Ticket'

```python
def caught_speeding(speed, is_birthday):
  s = speed
  if is_birthday == True:
    s -= 5
```

```
    if s <= 60:
        return "No ticket"
    if s >= 81:
        return "Big ticket"
    else:
        return "Small ticket"

caught_speeding(81,True)
```

    ⌐→ 'Small ticket'

```
    if s <= 60:
        return "No ticket"
    if s >= 81:
        return "Big ticket"
    else:
        return "Small ticket"
```