Weekly Python Challenge 32

Questions:

Very easy:

In this challenge, a farmer is asking you to tell him how many legs can be counted among all his animals. The farmer breeds three species:

1. chickens = 2 legs
2. cows = 4 legs
3. pigs = 4 legs </br> The farmer has counted his animals and he gives you a subtotal for each species. You have to implement a function that returns the total number of legs of all the animals.

Notes

1. Don't forget to return the result.
2. The order of animals passed is animals(chickens, cows, pigs).
3. Remember that the farmer wants to know the total number of legs and not the total number of animals.

In [1]:
```
def animals(chickens, cows, pigs):
    return chickens*2 + cows*4 + pigs*4
```

In [2]:
```
animals(2, 3, 5)
```

Out[2]:
```
36
```

In [3]:
```
animals(1, 2, 3)
```

Out[3]:
```
22
```

In [4]:
```
animals(5, 2, 8)
```

Out[4]:
```
50
```

Easy:

Using list comprehensions, create a function that finds all even numbers from 1 to the given number.

In [5]:
```
def find_even_nums(x):
    y=x//2
    return [i*2 for i in range(1,y+1)]
```

In [6]:
```
find_even_nums(8)
```

Out[6]:
```
[2, 4, 6, 8]
```

In [7]:
```
find_even_nums(4)
```

Out[7]:
```
[2, 4]
```

In [8]:
```
find_even_nums(2)
```

Out[8]:
```
[2]
```

In [9]:
```
find_even_nums(9)
```

Out[9]:
```
[2, 4, 6, 8]
```

Hard:

Create a function that creates a box based on dimension n.

In [10]:
```
def make_box(x):
    o=[]
    if x==0:
        o=[]
    elif x==1:
        o=['#']
    elif x==2:
        o=['##','##']
    else:
        o.append('#'*x)
        for i in range(x-2):
            o.append('#'+' '*(x-2)+'#')
        o.append('#'*x)
    for i in o:
        print(i)
    return(o)
```

In [11]:
```
make_box(3)
```

Out[11]:
```
###
# #
###
['###', '# #', '###']
```

In [12]:
```
make_box(5)
```

Out[12]:
```
#####
#   #
#   #
#   #
#####
['#####', '#   #', '#   #', '#   #', '#####']
```

In [13]:
```
make_box(2)
```

Out[13]:
```
##
##
['##', '##']
```

In [14]:
```
make_box(1)
```

Out[14]:
```
#
['#']
```

In [15]:
```
make_box(0)
```

Out[15]:
```
[]
```

In [ ]:
```

```