## Lab Summary

Jhonatan_LAB5 > lab5_1.py > ...

```python
# Universal Variables
start_num, stop_num  = 1, 11
nums_range = range(start_num, stop_num)
nums_list = list(nums_range)

odd_nums = nums_list[::2]
cubed_nums = [num**3 for num in nums_range]

print(odd_nums)# A
for num in cubed_nums: print(num)# B
for num in cubed_nums: print(num, end='|')# C
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```
@jhonatanparada499 ➜/workspaces/ET574 (lab5) $ /home/codespace/.python/current/bin/python3 /workspaces/ET574/Jhonatan_LAB5/lab5_1.py
[1, 3, 5, 7, 9]
1
8
27
64
125
216
343
512
729
1000
1|8|27|64|125|216|343|512|729|1000|@jhonatanparada499 ➜/workspaces/ET574 (lab5) $
```

Jhonatan_LAB5 > lab5_2.py > ...

```python
even_nums = [num for num in range(0, 101, 2)]

print(even_nums[:5])
print(even_nums[-5:])
print(even_nums[even_nums.index(44) : even_nums.index(88) + 1])

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```
@jhonatanparada499 ➜/workspaces/ET574 (lab5) $ /home/codespace/.python/current/bin/python3 /workspaces/ET574/Jhonatan_LAB5/lab5_2.py
[0, 2, 4, 6, 8]
[92, 94, 96, 98, 100]
[44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88]
```

Jhonatan_LAB5 > lab5_3.py > ...

```python
mult_of_4 = [num*4 for num in range(0,10 + 1)]
sec_list = []

for num in mult_of_4: sec_list.append(num // 2)

thir_list = sec_list[:]

for num in thir_list:
    num_index = thir_list.index(num)
    thir_list[num_index] = num // 2

print(mult_of_4)
print(sec_list)
print(thir_list)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```
@jhonatanparada499 ➜/workspaces/ET574 (lab5) $ /home/codespace/.python/current/bin/python3 /workspaces/ET574/Jhonatan_LAB5/lab5_3.py
[0, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40]
[0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

Jhonatan_LAB5 > lab5_4.py > ...

```python
err_message = 'Invalid Input.'
try:
    input_range = int(input("Enter a range: "))
    try:
        input_num = int(input("Enter an integer number: "))
        nums_list = [num for num in range(1, input_range + 1)]

        print(f'Multiplication Table of {input_num}')
        for num in nums_list:
            print(f'{num}\t*\t{input_num}\t=\t{num * input_num}')

    except ValueError:
        print(err_message)
except ValueError:
    print(err_message)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```
@jhonatanparada499 ➜/workspaces/ET574 (lab5) $ /home/codespace/.python/current/bin/python3 /workspaces/ET574/Jhonatan_LAB5/lab5_4.py
Enter a range: 10
Enter an integer number: 6
Multiplication Table of 6
1       *       6       =       6
2       *       6       =       12
3       *       6       =       18
4       *       6       =       24
5       *       6       =       30
6       *       6       =       36
7       *       6       =       42
8       *       6       =       48
9       *       6       =       54
10      *       6       =       60
```

2) 1. During lab5_1.py I noticed that the range function and slicing are very similar and could be used interchangeably. For example, when we have a list of numbers, the third element in the slicing format ([a:b:this] can do the same as the step parameter in the range function (range(a, b, this)).

2. During  lab5_4.py I almost missed an important detail when printing “Multiplication Table of 6”. At first, I wrote the print statement exactly as the output example, but when I entered another number, it did not make sense that it still displayed “…Table of 6”, so to fix it, I had to state the input number variable instead as in “f’…Table of {input_number}’”.