

GENERATORS, LIST COMPREHENSION AND DECORATORS

1)

```
t = "HellOaLl"

letters = []
letters = [letter for letter in t if letter.isupper()]
print(letters)
```

output:

```
(venv) C:\Users\Vaishnavi\PycharmProjects\pythonProject2>python task.py
['H', 'O', 'L']
```

2)

```
students = ['Smit', 'Jaya', 'Rayyan']
subjects = ['CSE', 'Networking', 'Operating System']

res = {}
i = 0
for key in students:
    # if there are extra keys without corresponding values, then add ""
    if i == len(subjects):
        value = ""
    else:
        value = subjects[i]

    res[key] = value
    i += 1

print(res)
```

output:

```
(venv) C:\Users\Vaishnavi\PycharmProjects\pythonProject2>python task.py
{'Smit': 'CSE', 'Jaya': 'Networking', 'Rayyan': 'Operating System'}
```

4)

```
str = 'Consultadd Training'

def reverse_string(str):
    str_len = len(str)
    for i in range(str_len-1, -1, -1):
        yield str[i]
```

```

        for item in reverse_string(str):
            print(item, end='')

print(str[::-1])

```

output:

```

(venv) C:\Users\Vaishnavi\PycharmProjects\pythonProject2>python task.py
gniniarT ddatlusnoC

```

5)

```

def div(a, b):
    yield a / b
    yield "This line will not print if i had written return above instead of yield"

def reverse_div(func):
    def inner(a, b):
        a, b = b, a
        return func(a, b)
    return inner

r_div = reverse_div(div)

r = div(4, 8)
print(r.__next__())
print(r.__next__())

```

output:

```

(venv) C:\Users\Vaishnavi\PycharmProjects\pythonProject2>python task.py
0.5
This line will not print if i had written return above instead of yield
2.0
This line will not print if i had written return above instead of yield

```