```
####################################
# Abdurrahman Bulut
####################################
####################################
# Task 1 - Python Exercises
####################################
x = 8
y = 3.2
z = 8j + 18
a = "Hello World"
b = True
c = 23 < 22
l = [1, 2, 3, 4]
d = {"Name": "Jake",
     "Age": 27,
     "Adress": "Downtown"}
t = ("Machine Learning", "Data Science")
s = {"Python", "Machine Learning", "Data Science"}

print(type(x), type(y), type(z), type(a), type(b), type(c), type(l), type(d), type(t), type(s))
# <class 'int'> <class 'float'> <class 'complex'> <class 'str'> <class 'bool'> <class 'bool'>
<class 'list'> <class 'dict'> <class 'tuple'> <class 'set'>

####################################
# Task 2 - Convert to Upper case
####################################
text = "The goal is to turn data into information, and information into insight"

# Classic Way
task2Result = []
for word in text.upper().split():
    task2Result.append(word.replace(',', ''))
print(task2Result)

# Comprehension
task2Result = [word.replace(',', '') for word in text.upper().split()]
print(task2Result)

####################################
# Task 3 - List Operations
####################################
lst = ["D", "A", "T", "A", "S", "C", "I", "E", "N", "C", "E"]

# step1
print(len(lst))
# step2
print("index 0", lst[0], "  index 10", lst[10])
# step3
takeDATA = lst[0:4]
print(takeDATA)
# step4
lst.remove(lst[8])
print(lst)
# step5
lst.append("B")
print(lst)
# step6
lst.insert(8, 'N')
print(lst)

####################################
# Task 4 - Dict Operations
####################################
dict = {'Christian': ["America", 18],
        'Daisy': ["England", 12],
        'Antonio': ["Spain", 22],
        'Dante': ["Italy", 25]}

# step1
```

```
keys = dict.keys()
print(keys)
# step2
values = dict.values()
print(values)
# step3
dict['Daisy'][1] = 13
print(dict)
# step4
dict.update({'Ahmet': ["Turkey", 24]})
print(dict)
# step5
dict.pop('Antonio')
print(dict)

####################################
# Task 5 - List operation - func
####################################
l = [2, 13, 18, 93, 22]


def func(lst):
    result = [[], []]
    for item in lst:
        if item % 2 == 0:
            result[0].append(item)
        else:
            result[1].append(item)
    return result[0], result[1]


even_list, odd_list = func(l)
print(even_list, odd_list)

####################################
# Task 6- Enumerate
####################################
ogrenciler = ["Ali", "Veli", "AyÅŸe", "Talat", "Zeynep", "Ece"]

for index, ogrenci in enumerate(ogrenciler):
    if index < 3:
        print("MÃ¼hendislik FakÃ¼ltesi " + str(index + 1) + " . Ã¶ÄŸrenci:   " + ogrenci)
    else:
        print("TÄ±p FakÃ¼ltesi " + str(index - 3) + " . Ã¶ÄŸrenci:   " + ogrenci)

####################################
# Task 7- Zip
####################################
ders_kodu = ["CMP1005", "PSY1001", "HUK1005", "SEN2204"]
kredi = [3, 4, 2, 4]
kontenjan = [30, 75, 150, 25]

zippedList = list(zip(kredi, ders_kodu, kontenjan))

for item in zippedList:
    print("Kredisi " + str(item[0]) + " olan " + item[1] + " kodlu dersin kontenjanÄ± " +
str(item[2]) + " kiÅŸidir.")

####################################
# Task 8- Set
####################################

kume1 = set(["data", "python"])
kume2 = set(["data", "function", "qcut", "lambda", "python", "miuul"])


def foo(set1, set2):
    if set1.issuperset(set2):
        print(set1.intersection(set2))
```

```
    else:
        print(set2.difference(set1))


foo(kume1, kume2)
```