



PROJE

Haftalık Rapor – 03.04.2021

3 NİSAN 2021

KIRIKKALE ÜNİVERSİTESİ – BİLGİSAYAR MÜHENDİSLİĞİ İÖ

AHMET MUNGAN – 160255081

İÇİNDEKİLER

ÖZET.....	2
VERİ BİLİMİ İÇİN PYTHON.....	3
ÖZEL KÜTÜPHANELERİN KURULUMU VE KULLANIMI.....	6
ANAHTAR KELİMELER.....	8
REFERANS VE KAYNAKÇA	9

ÖZET

Veri bilimine yönelik Python temellerinde öğrenilmesi gerekenler öğrenilmeye devam edilmiştir. Nesneye yönelik programlama ile ilgili araştırmalar ve örneklemeler yapılmıştır. Python’da hazır method nedir, ne işe yarar gibi sorulara cevaplar bulunmuştur. Özel bazı kütüphanelerin kurulumu ve basit ilk kullanımları ile ilgili çalışmalar yapılmıştır.

VERİ BİLİMİ İÇİN PYTHON

Veri biliminde önemli olan bazı Python özel bilgileri araştırılmaya ve uygulanmaya devam edilmiştir. Bu çerçevede Jupyter Notebook üzerinde yazılan örnek kodlar olabildiğince özet şeklinde paylaşılacaktır.

JPY 1

```
#args ve kwargs
>>> def Addition(*args):
>>>     return sum(args)
>>> Addition (10,20,30)
60
>>> def Funcution(*args):
>>>     return args
>>> type(Funcution(5,"deneme"))
tuple
>>> def Funcution2(**kwargs):
>>>     return kwargs
>>> type(Funcution2(a = 5, b = 4, c =3))
dict
#map
>>> def Division(number):
>>>     return number / 2
>>> myList = [1,2,3,4,5,6,7,8,9,10]
>>> list(map(lambda number : number * 4,myList))
[4, 8, 12, 16, 20, 24, 28, 32, 36, 40]
>>> list(map(Division, myList))
[0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0]
>>> def ControlFunction(string):
>>>     return "a" in string
>>> stringList = ["ahmet", "mungan", "fahrettin", "horasan", "proj
e", "deneme"]
>>> resultList = list(map(ControlFunction, stringList))
>>> resultList.count(True)
4
>>> resultList.count(False)
2
>>> list(filter(ControlFunction, stringList))
['ahmet', 'mungan', 'fahrettin', 'horasan']
```

JPY 1’de veri bilimine yönelik bazı örnek kodlamalar verilmiştir. Mapping, lambda ve filter işlemlerinin üzerinden geçilmiştir. Bu bileşenlerin hangi tip döndürmelere sahip olduğu görülmüştür.

JPY 2

```
>>> class Fruit():
>>>     def __init__(self, name, calorie):
>>>         self.name = name
>>>         self.calorie = calorie
>>>     def __str__(self):
>>>         return f"{self.name} is {self.calorie} calories"
>>>     def __len__(self):
>>>         return self.calorie
>>> banana = Fruit("Banana", 150)
>>> banana.calorie
150
>>> print(banana)
Banana is 150 calories
>>> myList = [1,2,3,"a",4.5]
>>> print(myList)
[1, 2, 3, 'a', 4.5]
>>> len(myList)
5
>>> len(banana)
150
>>> apple = Fruit("Apple", 200)
>>> len(apple)
200
>>> print(apple)
Apple is 200 calories
```

Python denince akla ilk gelen nesneye yönelik programlama uygulamalarıdır. Best practice’lere uygun olarak anlaşılıp araştırılmaya çalışılmıştır. JPY 2’de nesneye yönelik bir uygulama örneği yapılmaya çalışılmıştır. Ayrıca Python için özel methodların da idraki ve araştırmaları devam etmektedir. JPY 2’de özel methodlardan yazdırma methodlarının yanında uzunluk olarak belirlenebilecek “len” methodu bulunmaktadır. Veri biliminde listelerin len gibi hazır fonksiyonlar gibi nesnelerin üzerine yazılabilecek¹ özel methodları olması avantajdır.

¹ Methodun ezilmesi kastedilmiştir.

```
>>> while True:
>>>     try:
>>>         integer = int(input("Give me number: "))
>>>     except:
>>>         print("Please enter a real number!")
>>>         continue
>>>     else:
>>>         print("Thanks!")
>>>         break
>>>     finally:
>>>         print("finally called")
```

Hata yönetimi ve tip kontrolü de JPY 3'teki gibi kolay bir şekilde gerçekleştirilebilmektedir.

ÖZEL KÜTÜPHANELERİN KURULUMU VE KULLANIMI

Kullanılan bazı kütüphanelerin dahil edilmesi için paket kurulumları yapılması gerekmektedir. İster Python interpreter'ı içerisinde “pip” veya başka generator'lar ile ya da ortamın sunduğu kolaylıklar ile bu paketlerin kurulup modül şeklinde kullanımı söz konusudur.

CP 1

```
C:\Users\Ahmet Mungan> conda install numpy
Collecting package metadata (current_repodata.json): done
Solving environment: done
# All requested packages already installed.

C:\Users\Ahmet Mungan> conda install pandas
Collecting package metadata (current_repodata.json): done
Solving environment: done
# All requested packages already installed.

C:\Users\Ahmet Mungan> conda install matplotlib
Collecting package metadata (current_repodata.json): done
Solving environment: done
# All requested packages already installed.
```

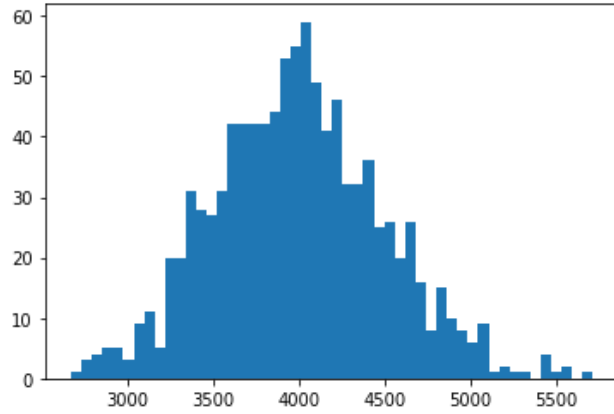
CP 1'deki komut satırı yazısında Anaconda için veri biliminde sıkça kullanılan temel üç kütüphane eklenmiştir.² Artık Anaconda ortamlarında numpy, pandas ve matplotlib kütüphaneleri kullanılabilir. RStudio'da yapılan ortalama hesaplama ve normal dağılım işlemlerini Python'da yapabilmek için birtakım yollar izlenmelidir. Tabiki bu yolların başında CP 1'de kurulmuş olan numpy kullanılacaktır. Tabi RStudio'nun en büyük avantajlarından görselleştirme ve grafik haline getirme işlemleri matplotlib ile gerçekleştirilecektir. Tamamen resmi dokümantasyonlardan yola çıkarak aslında RStudio'da gerçekleştirilen çoğu işlem gerçekleştirilebilir. Üstelik programlama dili kapsamında Python daha avantajlı ve esnek bir yapı sunacaktır.

JPY 4

```
>>> import numpy as np
>>> import matplotlib.pyplot as plt
```

² Eklenen kütüphanelerin tümü resmi kaynaklı kütüphanelerdir. Kütüphanelerin resmi kaynakları için bkz. [Referans ve Kaynakça](#)

```
>>> salaryList = np.random.normal(4000, 500, 1000)
>>> np.mean(salaryList)
3997.5077939681673
>>> matplotlib.hist(salaryList, 50)
>>> matplotlib.show()
```



JPY 4'te görüldüğü üzere numpy kullanılarak normal dağılıma uyacak şekilde bir maaş listesi hazırlanmıştır. Ayrıca yine numpy ve nimetleri kullanılarak bu listenin ortalaması hesaplanmıştır. Matplotlib ile de bu listenin çubuk grafikler ile ifadesi sağlanmıştır.

ANAHTAR KELİMELER

- ***JPY X*** : X. Jupyter Notebook çıktısı.
- ***CP X*** : X. Command Prompt çıktısı.

REFERANS VE KAYNAKÇA

- Python – Link için [tıklayınız](#).
- Python 3 Documents – Link için [tıklayınız](#).
- Anaconda – Link için [tıklayınız](#).
- Anaconda Individual Edition – Link için [tıklayınız](#).
- Anaconda Resources – Link için [tıklayınız](#).
- Numpy Documentation – Link için [tıklayınız](#).
- Pandas Documentation – Link için [tıklayınız](#).
- Matplotlib Documentation – Link için [tıklayınız](#).