**18.10 Ders Notları:**

Tecrübe yılı ve maaş: 2 ayrı grafik çizilmeli

Tecrübe yılı:1 boyut

Min. maaş:1 boyut

Max. maaş:1 boyut 2 ayrı grafik olmalı

Örnekler: Trivago’dan inceleyebilirsin

İlk hedef:1 tane data ile 1 döngü tamamlamak

**Ödev: Min 3 defa aynı şeyi tekrar yazın ve çalıştırın**

**\*\*Python: pypi.org**

pypi.org

“data sets” yazıp aratalım (datasets 3.0.1)

Pip install datasets

Örnek;

İsim=”Selcen”

Değişken

Değeri “string” türünde Selcen

Atama operatörü(iş gerçekleştirici olan)

“Ali” değerini bu değişkene atadık

İsim=”Ali”

\*\*Bir şeyin tipini öğrenmek için “type” fonsiyonunu çağır: type(ds)

Veri tipleri:

* string (içerisine metinsel ifadeler atanan)- ilkel olmayan veri tipi/ Non-primitif

İlkel veri tipleri/ Primitif

* Integer(tam sayılar)
* Float ondalıklı sayılar)
* Boolean (True/False)

**Veri tiplerini ezberle!**

**\*\*datasets.dataset\_dict (Datasetdict)**

\*\*Veri biliminde sütunlara “features/özellikler veyahut dimensions/boyut” denir.

\*\*0.data: ilk datamız oluyor, bunu 1 olarak sayacağız.

”=” : atama işaretidir

\*\*Değişkeni ilk defa tanımlayıp değer atıyoruz, daha önceden atanmış bir değişkenin değerini değiştirmiş oluyoruz) Örneğin bir okula yeni bir müdür atanması

İlkel veri tipleri:

# : comment, yorum (anlamı şu: makine için yazmadım demek)

* Tek satırlı yorumlar
* Çok satırlı yorumlar : 3 tırnak arka arkaya

Not: yorum olan kodlar çalışmaz!

\*\*Bir şeyin fonksiyon olup olmadığını nasıl anlarsın: kelimenin hemen devamında () açarak

type(x)

fonksiyon: işlem

Notlar:

**“Sider AI”** kurduğunda, çeviri ile uğraşmana gerek kalmaz. <https://sider.ai/tr/>

Sider AI, makine öğrenmesini kullanarak:

* Metin Analizi
* Bilgi Çıkarma
* Doğal Dil İşleme
* Görüntü İşleme

**“notebookLM.google**”: Google tarafından geliştirilen yapay zeka destekli bir not alma ve araştırma aracıdır.

NotebookLM'in temel işlevleri arasında şunlar yer alır:

* Hızlı Özetleme
* Not Alma
* Bilgi Arama
* Soru-Cevap
* Öğrenme ve Araştırma

**#Koleksiyonlar/Collections:** programlama dillerinde birden fazla veriyi tek bir yapı altında toplamak için kullanılan yapılardır

#Listeler/Lists

#Array/Dizi: aynı veri tipindeki birden fazla değeri tek bir isim altında toplamak için kullanılan bir veri yapısıdır

#Katar: makine öğrenmesinde "katman" (layer) veya "zincirleme" (chain) gibi kavramların Türkçe karşılığı olarak bazen informal bir şekilde "katar" kelimesi kullanılabilir.

\*\*Bütün stringler “” tırnak içeriside yazılır, örneğin yaş=”17”

\*\*Listelerin tanımının yolu köşeli parantez “ [ ] “ ile ifade edilir.

\*\*Bir listenin olmayışı boş olması ayrı bir durumdur. Örneğin;

meyveler=[ ] ->[ “elma”, “armut”, “muz”]

fonksiyon->len(uzunluk)

len->meyveler

3->bu listenin 3 tane elemanı var demek

Sınıf\_isim\_listeleri=[ [ ] ] : içinde 1 tane eleman olan liste, 0 eleman:boş liste

Listelerden oluşan listeler

Birden fazla veriyi bir arada tutmamızı sağlar

Koleksiyonlar