NOVA
it academy

HOŞGELDİNİZ! :)

Tipini Sevdiğimin Verileri Serisi – 3 : Tuple

NOVA it academy

BİR HATIRLATMA

- Sınıfa **ağzı açık kaplarda** içecekle LÜTFEN gelmeyiniz (Çay, kahve…)
  - Tetikli Termos, Kapaklı Su şişesi…
- Ayrılırken;
  - + Masa üstlerinde LÜTFEN ÇÖP BIRAKMAYINIZ,
  - + Bilgisayarlarınızı LÜTFEN KAPATINIZ,
  - + Sandalyelerinizi LÜTFEN DÜZENLİ bırakınız,
- Ders başlangıcı : Max. 5 Dk. Tolerans…

  19:05 Ders başlar!

NOVA
it academy

List : Neydi ki bu...

- Birden fazla değeri hafızada sıralı bir şekilde depolar
- [] kullanılarak yaratılır
- Elemanlar değiştirilebilir, eklenebilir, silinebilir
- Bütün veri tiplerini depolayabilir…!!!
- İndex [0] dan başlar
- Kendine ait fonksiyonları ile hayatımıza renk katar

Peki ya bu derste :

- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple
- tuple

NOVA it academy

tuple Nedir...

- Birden fazla değeri hafızada sıralı bir şekilde depolar
- Yaratılış itibari ile ilginç bir varlıktır
- Elemanlar DEĞİŞİRİLEMEZ, EKLENEMEZ, SİLİNEMEZ
- Bütün veri tiplerini depolayabilir…!!!
- İndex [0] dan başlar

NOVA it academy

Bir tuple şeyler...

```
demet = (“Gerbera”, “Lilyum”) #Boş tanımlama

#TEK ELEMANLI TUPLE YARATMA KURALI...

print(type(demet))

#Her bir eleman, ile ayrılır

print(demet)

OUTPUT : (“Gerbera”, “Lilyum”) # :) Noldu şimdi

len(demet)

demet[0] → Index ile kullanımı uygundur,

demet[start:end:step]
```

NOVA it academy

tuple elemanlarına erişim

```
Demet[0]

Demet[2:5:2]

demet[::-1]
```

Tanıdık geldi mi? :)

NOVA it academy

tuple elemanları kontrolü

```
demet = (“Gerbera”, “Lilyum”)

if “Gerbera” in demet:
  print(“Gerberayı çok severim”)
```

NOVA it academy

Hiç mi değişmez bu tuple

Patron bizken hiç şansı yok,
verinin içinden geçeriz

```
demet = (“Gerbera”, “Lilyum”)
temp = list(demet)
temp[1] = “Orkide”
demet = tuple(temp)
print(demet)
OUTPUT : (“Gerbera”, “Lilyum”,“Orkide”)
```

NOVA it academy

Ya eleman eklemek/çıkarmak istersem: tuple

```
"""
Yine listeye çeviriyoruz, liste nin
ekleme/çıkarma fonksiyonunu kullanıyor ve tekrar
tuple a çeviriyoruz. Hepsi bu kadan...
"""

demet = ("Gerbera", "Lilyum")

temp = list(demet)

. . . . . .

print(demet)
OUTPUT : ("Gerbera", "Lilyum","Orkide")
```

NOVA it academy

Packing - Unpacking tuple

```
"""
Aslında bunu daha önce gördük : Çoktan Çoka, bla bla
"""

demet = ("Gerbera", "Lilyum", "Orkide") #Packing

a, b, c = demet

x, *y = demet
```

NOVA it academy

tuple loop

```
demet = (“Gerbera”, “Lilyum”, “Orkide”) #Packing

for eleman in demet :
  print(eleman)

#------------------------------

for i in range(len(demet)):
  print(demet[i])

#------------------------------

i=0
while i < len(demet):
  print(demet[i])
  i+=1
```

tuple çoğaltma

```
demet = (“Gerbera”, “Lilyum”, “Orkide”)
demet2 = (“Gerbera”, “Lilyum”, “Orkide”, “Yasemin”)

buket = demet + demet2

#-----------------------------

buket = demet * 2 + demet2 * 2
```

NOVA it academy

tuple Fonksiyonları...

1 : count
2 : index

Ve bi kaç tane daha...

Bu sefer HİÇ üzmücem, söz...

NOVA it academy

tuple Fonksiyonları...

```
.count(obj)
```

Tuple içinde aradığınız elemanın kaç adet olduğunu verir.

```
demet = ("Gerbera", "Lilyum", "Orkide")

a = demet.count("Gerbera")
```

NOVA it academy

tuple Fonksiyonları...

.index(obj)

Tuple içinde aradığınız elemanın index numarasını verir.

```
demet = (“Gerbera”, “Lilyum”, “Orkide”)

idx = demet.index(“Gerbera”)

#Tuple içinde bulamazsa “Value not Found” #hatası verir.
```

TİPİNİ SEVDİĞİMİN VERİ SERİSİ - 3: Tuple

BURDA BİTER.

Peki ya bu derste :

◦ Method /

Function ◦ Method / ◦ Method / ◦ Method /

◦ Method / Function ◦ Method / Function Function

Function ◦ Method / ◦ Method / Function ◦ Method /

Function Function ◦ Method / Function ◦ Method / ◦ Method /

◦ Method / Function ◦ Method / Function Function

Function ◦ Method / ◦ Method / Function ◦ Method /

Function Function ◦ Method / Function ◦ Method / ◦ Method /

◦ Method / Function ◦ Method / Function Function

Function ◦ Method / ◦ Method / Function ◦ Method /

Function Function ◦ Method / Function ◦ Method / ◦ Method /

◦ Method / Function ◦ Method / Function Function

Function ◦ Method / ◦ Method / Function ◦ Method /

Function Function ◦ Method / Function ◦ Method / ◦ Method /

◦ Method / Function ◦ Method / Function Function

Function ◦ Method / ◦ Method / Function ◦ Method /

Function Function ◦ Method / Function ◦ Method / ◦ Method /

◦ Method / Function ◦ Method / Function Function

Function ◦ Method / ◦ Method / Function ◦ Method /

Function Function ◦ Method / Function ◦ Method / ◦ Method /

◦ Method / Function ◦ Method / Function Function

Function ◦ Method / ◦ Method / Function ◦ Method /

Function Function ◦ Method / Function ◦ Method / ◦ Method /

◦ Method / Function ◦ Method / Function Function

Function ◦ Method / ◦ Method / Function ◦ Method /

NOVA
it academy

Ne ola ki bu Fonksiyon

- Geliştirici tarafından tasarlanan,
- Kendine ait bir bloğu olan
- def anahtar kelimesi ile tanımlanan,
- Parametre/Arguman alan/almayan
- Sonuc donduren/dondurmeyen (Biz nasil istersek...)

NOVA it academy

Bir metodun hayatı...

```
#Oluşturma
def myFunction():
    print("Hello World")


#Çağırma
myFunction()
```

NOVA it academy

Ama niye...

```
# Benzer işlerin kodunu tekrar tekrar yazmamak için

# Okunabilir kod yazmak için

# Ortalıktaki karışıklığı azaltmak için

# Kodun modüler olabilmesi için
```

NOVA it academy

Bir adım ileri: Argümanlar

```
# Fonksiyonda kullanılmak üzere veri vermek için

# Adet sınırı yoktur, ne kadar lazımsa ama virgülle ayrılması gerekir,

# Fonksiyonlar, önce tanımlanır, sonra kullanılır!!!
```

NOVA it academy

Func(Args)

```
#Oluşturma
def myFunction(isim):
    print(isim , “World”)


#Çağırma
myFunction(isim)
```

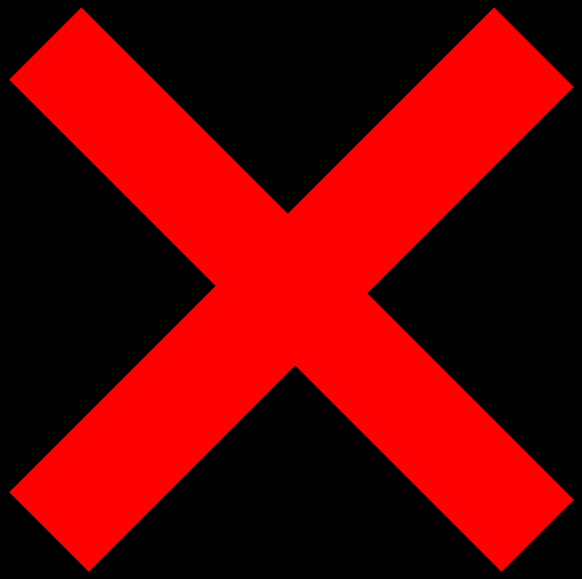
NOVA it academy

Func(Args)

```
#Oluşturma
def myFunction2(isim, soyisim):
    print(isim , soyisim)
-----------------------------------
#Çağırma
myFunction2(isim, soyisim)
-----------------------------------
#ÇALIŞMAZ !!!
myFunction2(isim)
```

NOVA it academy

Func(Args)



```
#Oluşturma
def myFunction(isim, soyisim):
    print(isim , soyisim)
-------------------------------------
def myFunction(isim):
    print(isim , "Hoşgeldiniz")
-------------------------------------
# KANDIRAMAZSINIZ, YEMEZ!!!
```

Bu sevda bitmez!

Ama bu ders biter!

NOVA it academy

BİR HATIRLATMA

- Ayrılırken;

  + Masa üstlerinde LÜTFEN ÇÖP BIRAKMAYINIZ,

  + Bilgisayarlarınızı LÜTFEN KAPATINIZ,

  + Sandalyelerinizi LÜTFEN DÜZENLİ bırakınız,

TEŞEKKÜRLER, GÖRÜŞMEK ÜZERE...