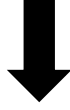


PYTHON DERSLERİ

Türkçe+ Mantık+ Matematik

Hayal Gücü+ Sanat+



Algoritma



Programlama

Programlama Sanatına Giriş

Giriş

Python Ders 00 - Programlamaya Giriş ve Python Kurulumu

Python Ders 01- Başlıyoruz (Python 3'ü Windows'a kurmak)

Aritmetik ve Matematik İşlemleri

Python Ders 02 – Aritmetik İşlemler (+,-,*,/, %, //)

Karakter, Kelime, Cümle Türkçe İşlemleri

Python Ders 03- Stringler

Python Ders 04 - Print Fonksiyonu

Python Ders 05- String İşlemleri

Mutfak Gereçleri -Kaplar, Bardaklar, Tencere, Tava

Python Ders 06- Değişkenler

Yumurta Kolileri

Python Ders 07– Listeler

Giriş ve Düzenleme İşlemleri

Python Ders 08- input() fonksiyonu ve format() metodu

Mantık İfadeleri

Python Ders 09- Koşullu Durumlar (if-elif-else)

Python Ders 10- if elif else (Örnek Program)

Döngüler (Tekrarlı İşlemler)

Python Ders 11- Döngüler-While Döngüsü

Python Ders 12- While Döngüsü (Örnek Program)

Python Ders 13- For Döngüsü

Python Ders 14-For Döngüsü Örnek Program

Python Ders 15- Break ve Continue Deyimleri

Fonksiyonlar (Mutfak Aletleri)

Python Ders 16- Fonksiyonlar'a Giriş

Python Ders 17 - Fonksiyonlarda Return

Python Ders 18 - Fonksiyonlarda Varsayılan Değerler

Python Ders 19 - Geometrik Şekil Hesaplama (Örnek Fonksiyon)

Python Ders 20 - Fonksiyonlarda Recursion(İç içe Çağırma)

Python Ders 21 - Global ve Yerel Değişkenler

Çevirmenler İş Başında

Python Ders 22 - Sözlükler(Dictionary)

Python Ders 23 - Modüller(Kütüphaneler)

Python Ders 24 - İnternette Resim İndirme(Modül Örneği)

Python Ders 25 - Bekle! Orda Hata Var (try ve except)

Python Ders 26 - try,except ve finally

Artık Her Şey Kayıt Altında

Python Ders 27 - Dosya Açmak ve Yazmak

Python Ders 28 - Dosyadan Veri Almak, Okuma İşlemleri

Python Ders 30 - Dosyalarda değişiklik yapmak

Veriler Hizalanıyor

Python Ders 31 - Sqlite Veritabanı Tablo Oluşturma

Python Ders 32 - Sqlite Veritabanı - Tabloya değer ekleme

Python Ders 33 - Sqlite Veritabanı - Tablodaki Verileri Çekme

Python Ders 34 - Sqlite Veritabanı - Tablo Güncelleme

Nesneleri Dünyası

Python Ders 35 - Sınıflar ve Nesneler (Nesne Tabanlı Programlama)

Python Ders 36 - NTP - Yapıcı Fonksiyon (init fonksiyonu)

Python Ders 37 - NTP - Inheritance (Kalıtım)

Python Ders 48 - NTP (Çoklu Kalıtım)

DERS 00- GİRİŞ VE KURULUM

Programlama bir sanattır. Bir programlama diline ait bilgiler program yazıcısının beceri ve düşüncesi ile birleşince program ortaya çıkacaktır.

Bu bakış açısı ile problem çözmeye başladığınızda

- ✓ Mantık işlemleri için; if/ elif/ else
- ✓ Değişkenler için; int/double/float/String/boolean
- ✓ Döngüler için; while/for
- ✓ Fonksiyonlar için; def bloğu

Tıpkı bir ressamın boyaları, fırçaları gibi; sizi yazdığınız programda hayal gücünüzün zenginliğine yaklaştıracaktır.

Bilgisayar: Sadece kendilerine verilen komutları (yönergeleri) yerine getiren makinelerdir.

Program: Bilgisayara bir hesaplamayı nasıl yapacağını söyleyen bir dizi yönergeye denir.

Hesaplama bazen bir matematik denklemini çözmeye, bazen metin içerisindeki bir kelimeyi arama olabilir.

Programlama Dili: Yönergelerin yazıldığı dillere denir. Makine Dilinden yüksek seviyeli dillere doğru gider.

Derlemeli Diller: Yazılan program kodu derleyici tarafından okunup makine diline çevrilir. Hızlıdır. Fakat o mimaride çalışır.

Yorumlamalı Diller: Program kodu bir metin dosyasında bulunur. Bu dosya yorumlayıcı tarafından okunup yönergeler yerine getirilir. İşletim sistemi ve mimari farkı gözetmeden her yer de çalışır.

Programlama Hataları: **HATASIZ PROGRAM OLMAZ!**

- 1- Yazım Hataları
- 2- Çalışma Zamanı Hataları
- 3- Anlamsal Hatalar

DERS 00- PYTHON KURULUMU

Ders 00: PYTHON KURULUMU	
Ders Notları	Kod Uygulamaları
Python IDLE	<ul style="list-style-type: none">- Python 3 ve üzeri (Güncel 3.10 sürümü)- Download Adresi : https://www.python.org/- Python Kurulum(Add Python Path (Cmd) çalışmasını sağlar)- Python Etkileşimli Kabuk(Shell), IDLE,
PYCHARM	PyhCharm https://www.jetbrains.com/pycharm/download/ Cominity Sürümü
VISUAL STUDIO	https://code.visualstudio.com/download
ONLINE	http://repl.it (Üyelik ile giriş yapın)
GOOGLE	https://colab.research.google.com

Aritmetik ve Matematik İşlemleri

Ders 01: ARİTMETİK İŞLEMLERİ	
Ders Notları	Kod Uygulamaları
ARİTMETİK İŞLEMLERİ	
(+) Toplama (-) Çıkarma (%) Kalanı verir	(*) Çarpma (/) Bölme (//) Bölüm(Tam Kısımlı verme) (**) Üssü(Kuvvet)
İşlem Önceliği 1. () Parantez 2. **,*/,+,- Değişken(İlk Giriş) x= 4, y=3 >>>x+y >>>7	>>> 3+4 7 >>> 60/7 10.0 >>> 22//7 3 >>> 22/7 3.142857142857143 >>> (8*2)/10-(5/3) -0.06666666666666665 >>> 18%4 2 >>> 2**3 8
Alıştırma 1: Bir kenar uzunluğu girilen karenin çevresi ve alanını hesaplayınız? Alıştırma 2 : Yarıçapı girilen dairenin çevresini ve alanını hesaplayınız? Alıştırma 3: Kullanıcının girdiği (3825) saniyeleri; saat, dakika ve saniye olarak gösteriniz.	
Süper soru: 16'nın karekökünü alınız.	

TÜRKÇE METİN İŞLEMLERİ

DERS 02: STRING (KARAKTERLER, DİZGE, YAZI, METİN...)

Ders Notları

- Tek Tırnak (shift + 2)
- Çift Tırnak
- Üç tane Çift Tırnak

Kod Uygulamaları

- Karakterler (') veya (") arasında yazılır.
- Hangi tırnak çifti ile başlanırsa onunla biter.
- Tek tırnak kullanırken ; 'Serkan\'ın bugün dersi var.'
(**Kaçış dizisi kullanılır.)
- (+) Toplama işareti ile Stringler birleştirilir.
'Python'+ 'Dersleri'
- (*) Çarpma işareti Stringlerden aynı karakterden yazar.
'Python'*3

```
>>> Python Dersleri
SyntaxError: invalid syntax
>>> "Python Dersleri"
'Python Dersleri'
>>> 'Python Dersleri'
'Python Dersleri'
>>> 'Serkan\'ın bugün dersi var.'
SyntaxError: invalid syntax
>>> "Serkan\'ın bugün dersi var."
"Serkan\'ın bugün dersi var."
>>> 'Serkan\'ın bugün dersi var.'
"Serkan\'ın bugün dersi var."
>>> ilk= "Python"
>>> son="Dersleri"
>>> ilk
'Python'
>>> son
'Dersleri'
>>> ilk+son
'PythonDersleri'
>>> ilk+"Öğrenmek"+son+"Çalışmakla olur"+"3"
'PythonÖğrenmekDersleriÇalışmakla olur3'
>>> ilk+ilk+ilk
'PythonPythonPython'
>>> ilk*3
'PythonPythonPython'
```

DERS 03- Ekranaya Yazdırma

Ders 03: PRINT FONKSİYONU	
Ders Notları	Kod Uygulamaları
print(): İçine aldığı değerleri ekrana yazar.	Print() Fonksiyonu
<pre>>>>Print("Serkan'ın bugün dersi var.")</pre> <p>- (,) stringleri birleştirir.(Boşluklu) print("Python","Dersleri")</p> <p>- (+)stringleri birleştirir. (Boşluksuz) print("Python"+"Dersleri")</p> <p>- str() : Sayısal verileri Stringe çevirir. - int() : String verileri tam sayıya çevirir. - float():String verileri float (ondalıklı) sayıya çevirir.</p> <p>Hatırlatma:</p> <ol style="list-style-type: none">1. Tek tırnak (' ')2. Çift tırnak (" ")3. Üç tırnak (""" """)	<pre>>>> print("Serkan'ın bugün dersi var.")</pre> <p>Serkan'ın bugün dersi var.</p> <pre>>>> print("Python","Dersleri")</pre> <p>Python Dersleri</p> <pre>>>> print("Python"+"Dersleri")</pre> <p>PythonDersleri</p> <pre>>>> print("Python",3)</pre> <p>Python 3</p> <pre>>>> print("Ders",3,3.3,"Dersleri")</pre> <p>Ders 3 3.3 Dersleri</p> <pre>>>> "Bilgisayar"+3</pre> <p>Traceback (most recent call last): File "<pyshell#5>", line 1, in <module> "Bilgisayar"+3 TypeError: must be str, not int</p> <pre>>>> print("Python"+3)</pre> <p>Traceback (most recent call last): File "<pyshell#6>", line 1, in <module> print("Python"+3) TypeError: must be str, not int</p> <pre>>>> print("Python"+str(3))</pre> <p>Python3</p> <pre>>>> print(int("3")+5)</pre> <p>8</p> <pre>>>> print(float('3.5')+5)</pre> <p>8.5</p>

DERS 04 - STRING İŞLEMLERİ -2 (DİZİLER[])

Ders 04: STRING İŞLEMLERİ -2 (DİZİLER[])	
Ders Notları	Kod Uygulamaları
Dizi İşlemleri	String []
<p>s="Python Dersleri" print(s) s[0]: dizinin ilk index'inin değerini verir: P *** Dizilerin ilk index'i 0 'dır. s[-1]: dizinin son index'inin değerini verir.</p> <p>Aralıkta işlem s[2:10] : 2'den 10' kadar olan değerleri yazar. s[:10] : baştan 10 kadar olan değerleri yazar. s[2:] : 2'den sona kadar olan değerleri yazar. s[:] : Baştan sona bütün değerleri yazar. len() : stringlerin uzunluğunu verir. **** Kaç tane karakter olduğunu. len(s) :15</p>	<pre>>>> s = "Python Dersleri" >>> print(s) Python Dersleri >>> s[0] 'P' >>> s[-1] 'i' >>> s[2:10] 'thon Der' >>> s[:10] 'Python Der' >>> s[1:10] 'ython Der' >>> s[:] 'Python Dersleri' >>> len(s) 15</pre>
ata ="Mustafa Kemal ATATÜRK"	

KAPLAR, TENCERE, TAVA- MUTFAK GEREÇLERİ

Ders 06: DEĞİŞKENLER	
Ders Notları	Kod Uygulamaları
Değişken: Programlamada verilerin konulacağı kaplara denir.	Veri Tipleri Type() Fonksiyonu
<p>int: Tamsayılar float: Ondalıklı Sayılar *** Ondalıklı sayı yazılırken ayraç olarak (.) kullanılır. string: Karakterler, metinler</p> <p>type() :İçine aldığı parametrenin(veri) tipini verir.</p> <p>type(12) class int type(2.34) class float type("Python") class str type(True) class bool type([1,2,3,4]) class list type((1,2,3,4)) class tuple type({"elma": "meyve"}) class dict</p> <p>Değişken tanımlama x=3 y=5 x= x+y 8 ilk="Python" son="Dersleri" ilk=ilk+son PythonDersleri ***Python değişken türü atandıktan sonra kendisi tanır.</p>	<pre>>>> type(12) <class 'int'> >>> type(2.34) <class 'float'> >>> type("Python") <class 'str'> >>> type([1,2,3]) <class 'list'> >>> type((1,2,3)) <class 'tuple'> >>> type({"elma": "meyve"}) <class 'dict'> >>> x=35 >>> type(x) <class 'int'> >>> str(3) '3' >>> float(3) 3.0 >>> int(3.5) 3 >>> int("Serkan") Traceback (most recent call last): File "<pyshell#11>", line 1, in <module> int("Serkan") ValueError: invalid literal for int() with base 10: 'Serkan' >>> int("34a") Traceback (most recent call last): File "<pyshell#12>", line 1, in <module> int("34a") ValueError: invalid literal for int() with base 10: '34a' >>> int(348) 348</pre>

Yumurta Kolileri

DERS 07 - LİSTELER

Ders 07: LİSTELER	
Ders Notları	Kod Uygulamaları
<p>Listeler: Değişik veri tiplerini tutan depo(kümler)dir.</p>	<p>Diziler []</p>
<p>a=[12,60,"Python",3.55] **Listeler yapı olarak stringler gibi işlem görür. Listelerin eleman sayısı len(a) ----- > 4 b=["elma","armut","şeftali",35,60] ogrenci=["Akın", "Kandemir",9,308] ogrenci[0] ----- > Akın Index'e değer atama ogrenci[0] = "Baran" append(): Listeye eleman eklemek ogrenci.append("UAL") --- > listenin sonuna Değer ekler. ogrenci= ogrenci+["Urla","İzmir"] ogrenci[:2] --- > Baştan ilk iki elemanı gösterir. Liste'de Eleman Değişliği ogrenci[0:2] = ["Kemal", "ATA"] Liste'de Eleman Silme ogrenci[:2]= [] Listeyi Temizleme ogrenci[:]= []</p>	<pre>>>> a=[12,60,"Python",3.55] >>> a [12, 60, 'Python', 3.55] >>> len(a) 4 >>> b= ["elam","armut","şeftali",35,60] >>> b ['elam', 'armut', 'şeftali', 35, 60] >>> len(b) 5 >>> ogrenci=["Akın","Kandemir",9,308] >>> ogrenci ['Akın', 'Kandemir', 9, 308] >>> ogrenci[0] 'Akın' >>> ogrenci[1] 'Kandemir' >>> ogrenci[0]="Baran" >>> ogrenci ['Baran', 'Kandemir', 9, 308] >>> ogrenci.append("UAL") >>> ogrenci ['Baran', 'Kandemir', 9, 308, 'UAL'] >>> ogrenci+["URLA","İZMİR"] ['Baran', 'Kandemir', 9, 308, 'UAL', 'URLA', 'İZMİR'] >>> ogrenci[:2]= ["Kemal","KARA"] >>> ogrenci ['Kemal', 'KARA', 9, 308, 'UAL'] >>> ogrenci[:2]= [] >>> ogrenci [9, 308, 'UAL'] >>> ogrenci[:]= [] >>> ogrenci []</pre>

DERS 08 - DÜZENLEME İŞLEMLERİ

Ders 008: input() ve .format() Fonksiyonları	
Ders Notları	Kod Uygulamaları
<p>input() : Kullanıcıdan(klavyeden) veri girişi sağlar.</p> <p>.format() : Print ile yazdırma işlemi yapılırken, verilerin düzenli girilmesi ve gösterilmesini sağlar.</p>	<p>input()</p> <p>.format()</p>
<pre>a= input() print(a) a=input("Bir sayı Giriniz : ") print(a) *** <u>input() ile girilen değerler stringdir.</u> *** <u>Biz sayıya çeviririz.</u> int(a) .format() fonksiyonu Print ile yazdırma işlemi yapılırken, verilerin düzenli girilmesi ve gösterilmesini sağlar. print("Merhaba {}".format("Python"))</pre>	<pre>a=input("") print(a) a=input("Bir sayı Giriniz : ") print("Girdiğiniz Sayı : ",a) x= input("Birinci Sayıyı Giriniz : ") y= input("İkinci Sayıyı Giriniz : ") z= input("Üçüncü Sayıyı Giriniz : ") print("Sayılar hesaplanıyor.....") print("Toplam : ", x+y+z) print("Toplam : ",int(x)+int(y)+int(z)) print("İşlem Bitti") Öğrenci Kaydetme Programı print("Öğrenci Kaydetme Programı.....") ad=input("Öğrenci Adını Giriniz :") soyad = input("Öğrenci Soyadını Giriniz :") numara = input("Öğrenci Numarasını Giriniz :") print("Bilgileri Listeye alıyoruz") ogrenci = [ad, soyad, numara] print("Bilgiler Database kaydediliyor.....") print("Öğrenci Adı :",ad," Öğrenci Soyadı : ",soyad," Öğrenci Numarası : ",numara) print("Veriler Kaydedildi....") print("Öğrenci Bilgileri format() fonksiyonu ile kaydetme") print("Öğrenci Adı : {}\nÖğrenci Soyadı : {}\nÖğrenci Numarası : {} ".format(ogrenci[0],ogrenci[1],ogrenci[2]))</pre>

Ders 09:KOŞULLU DURUMLAR	
Ders Notları : (if-elif-else)	Kod Uygulamaları
<p>Programlar yukardan aşağı ve soldan sağa çalışırlar. Bazen belirli bir şartı gerçekleştiğinde çalışmasını istediğimiz komutların çalışması için koşullu durum</p> <p>if : Eğer, else : Değilse</p>	<p>Ders 000:</p>
<p>if koşul ifadesi kullanılır.</p> <p>Mantıksal İfadeler:</p> <p>= = Eşittir != Eşit Değildir > Büyüktür < Küçüktür >= Büyük Eşittir <= Küçük Eşittir</p> <p>if koşul(Mantıksal ifade): komutlar</p> <p>if/else ifadesi:</p> <pre>print("Öğrenci Notuna Göre Geçme/ Kalma Durumu...") ortalama = int(input("Ortalamayı Giriniz: ")) if ortalama>=50: print("Geçti") else: print("Kaldı")</pre> <p>if/elif(ya da eğer)/else ifadesi: Çok Yönlü Koşullu Durum</p> <pre>print("Öğrenci Notuna Göre 1,2,3,4,5 Yazma Durumu...") ortalama = int(input("Ortalamayı Giriniz: ")) if ortalama>=85: print("5") elif ortalama >=70: print("4") elif ortalama >=60: print("3") elif ortalama >=50: print("2") else: print("Kaldı")</pre> <p>*** elif yerine if kullanılırsa(Zincirleme Durum) bütün koşullara tek tek bakılır ve doğru olan cevaplar gösterilir.</p> <p>elif(Çok Yönlü Koşullu Durum) kullanılması durumunda doğru olan koşul çalışır diğerleri atlanır.</p>	<p>Koşullu Durumlar(if-elif-else) a</p> <pre>print("Öğrenci Notuna Göre Geçme/ Kalma Durumu...") ortalama = int(input("Ortalamayı Giriniz: ")) if ortalama>=50: print("Geçti") else: print("Kaldı")</pre> <pre>print("Öğrenci Notuna Göre 1,2,3,4,5 Durumu...") notu = int(input("Ortalamayı Giriniz: ")) if notu>=85: print("5") elif notu >=70: print("4") elif notu >=60: print("3") elif notu >=50: print("2") else: print("1")</pre> <p>*** elifler if' e çevrilirse doğru olan koşullar çalışır</p>

DERS 10 - KOŞULLU DURUMLAR -2 (IF-ELIF-ELSE)

Ders 10: KOŞULLU DURUMLAR -2	
Ders Notları : if/elif/else	Kod Uygulamaları
Kullanıcı Adı ve Parola Kontrolü	Ders 000:
<p># Tekli Yorum satırı “”” çoklu yorum satırı, üç tırnak“”” Birleşik koşul ifadeleri and: ve or: veya not: Değil</p> <p>Örnek: Kullanıcı Adı ve Parola sorgulama programı</p>	<p>Koşullu Durumlar(if-elif-else) <i>#Kullanıcı Adı Parola sorgulama programı</i></p> <pre>defKullanici= "Python" defParola = "1234" kullanici= input("Kullanıcı Adınızı Giriniz : ") parola = input("\$ifrenizi Giriniz : ") if((defKullanici== kullanici) and (defParola== parola)): print("Sisteme Giriş Yaptınız. Merhaba {}".format(kullanici)) elif ((defKullanici!= kullanici) and (defParola== parola)): print("Hatalı Kullanıcı Adı") elif ((defKullanici== kullanici) and (defParola!= parola)): print("Hatalı Şifre") else: print("Tekrar deneyin")</pre>

DERS 11 - DÖNGÜLER / TEKRARLI YAPILAR

Ders 11: DÖNGÜLER WHILE- 1	
Ders Notları	Kod Uygulamaları
While	While
<p>Döngü ler belirli işlemleri, belirlenen sayıda veya bitirme koşulu gerçekleşene kadar devam ettiren yapılara denir. Hesap makinesi sürekli işlem yapar çıkmak istediğimizde program sonlanır.</p> <p>Döngü Değişkeni başlatma while (mantık cümlesi): (Girinti)Yapılacak işlemler(Komutlar) (Girinti)Artırma işlemi</p> <p>Değişkeni başlatma ilk değer ataması---- > i=0 While (i <= 10) --- >True Yapılacak İşlemler--- > print(i) Artırma ---- > i +=1</p> <p>i +=1 (i 1 artırır) i -=1 (i 1 azaltır)</p>	<pre>""" Döngü Değişkeni başlatma while (mantık cümlesi): (Girinti)Yapılacak işlemler(Komutlar) (Girinti)Artırma işlemi """ i = 0 while (i <= 10): print("i'nin değeri", i) i +=1</pre>

DERS 12 - DÖNGÜLER / TEKRARLI YAPILAR

Ders 11: DÖNGÜLER WHILE- 2	
Ders Notları	Kod Uygulamaları
While	While
<p>while/if/elif/else/break <i>#Kullanıcı Adı Parola Kontrolü</i> <i>#break deyimi - Döngünün içinde kullanıldığında döngü sonlanır veya döngüden çıkarılır.</i></p> <pre>defkullanici = "Python" defparola = "1234" while (True): kullanıcı = input("Kullanıcı Adı :") parola = input("Parola :") if((kullanıcı == defkullanici) and (parola == defparola)): print("Hoş geldinizi,", kullanıcı) break elif((kullanıcı != defkullanici) and (parola == defparola)): print("Kullanıcı Adınız yanlış girdiniz !") elif((kullanıcı == defkullanici) and (parola != defparola)): print("Şifrenizi mi unuttunuz? ") print("Şifrenizi değiştirmek ister misiniz? (E/H)") cevap = input("E/H : ") if(cevap == "E"): yenisifre=input("Yeni şifrenizi giriniz : ") defparola=yenisifre print("Şifre başarılı bir şekilde değişti.") else: print("Tekrar deneyiniz.") *** Döngü sürekli çalışıyor(True). Kullanıcıadı ve şifre doğru girildiğinde break deyimi ile döngüden çıkılıyor. Şifre değişikliği için if koşulu ile "E" cevabı kontrol ediliyor.</pre>	<p>While Döngüsü <i>#Kullanıcı Adı Parola Kontrolü</i> <i>#break deyimi - Döngünün içinde kullanıldığında döngü sonlanır veya döngüden çıkarılır.</i></p> <pre>defkullanici = "Python" defparola = "1234" while (True): kullanıcı = input("Kullanıcı Adı :") parola = input("Parola :") if((kullanıcı == defkullanici) and (parola == defparola)): print("Hoş geldinizi,", kullanıcı) break elif((kullanıcı != defkullanici) and (parola == defparola)): print("Kullanıcı Adınız yanlış girdiniz !") elif((kullanıcı == defkullanici) and (parola != defparola)): print("Şifrenizi mi unuttunuz? ") print("Şifrenizi değiştirmek ister misiniz? (E/H)") cevap = input("E/H : ") if(cevap == "E"): yenisifre=input("Yeni şifrenizi giriniz : ") defparola=yenisifre print("Şifre başarılı bir şekilde değişti.") else: print("Tekrar deneyiniz.")</pre>

DERS 13 - DÖNGÜLER / TEKRARLI YAPILAR

Ders 11: DÖNGÜLER WHILE- 3 ALIŞTIRMALAR	
Ders Notları	Kod Uygulamaları
While	While
Alıştırma -1: Klavyeden girilen sayının rakamlarının toplamını yazdıran program.	Alıştırma -1 <pre>sayi=int(input("Sayıyı giriniz : ")) toplam=0 while sayi>0: toplam += sayi %10 sayi //=10 #print("Toplam :",toplam) #print("Sayı : ",sayi) print("Girilen sayının rakamları toplamı : ",toplam)</pre>

TEKRARLI YAPILAR

DERS 13- DÖNGÜLER FOR -1

Ders 13: FOR DÖNGÜSÜ -1	
Ders Notları: For Döngüsü	Kod Uygulamaları
Belirli sayıdaki işlemleri tekrarlar	Ders 000:
Stringler for Döngüsü: string = "Python" for i in string: print(i) Listelerde for döngüsü: liste = ["Elma", "Armut", "Şeftali"] for i in liste: print(i) range() fonksiyonu: *Not: print ile kullanılırken * kullanılır. print(*range()) print(*range(2,10)) print(*range(2,10,3)) for ile range () kullanımı for i in range(1,10): print(i * " ") Ekran alt alta * basar.	For Döngüsü string = "Python" for i in string: print(i) liste = ["Elam", "Armut", "Şeftali"] for i in liste: print(i) <i>#range fonksiyonu</i> >>> print(*range(10)) 0 1 2 3 4 5 6 7 8 9 >>> print(*range(1,10)) 1 2 3 4 5 6 7 8 9 >>> print(*range(1,10,3)) 1 4 7 for i in range(1,10): print(i*" ")

DERS 14- DÖNGÜLER FOR -2

Ders 14: DÖNGÜLER FOR -2	
Ders Notları	Kod Uygulamaları
Faktöriyel Hesaplama	For Döngüsü
<pre>#Faktöriyel Hesaplama #Negatif sayıların Faktöriyeli olmaz #0!=1 faktoriyel =1 while True: sayi = int(input("Pozitif bir sayı giriniz : ")) if(sayi<0): print("Lütfen Negatif sayı girmeyiniz!") else: for i in range(1,sayi+1): faktoriyel *= i print(sayi,"!= ",faktoriyel) break</pre>	<pre>#Faktöriyel Hesaplama #Negatif sayıların Faktöriyeli olmaz #0!=1 faktoriyel =1 while True: sayi = int(input("Pozitif bir sayı giriniz : ")) if(sayi<0): print("Lütfen Negatif sayı girmeyiniz!") else: for i in range(1,sayi+1): faktoriyel *= i print(sayi,"!= ",faktoriyel) break</pre>

DERS 15- DÖNGÜLER ÇIKIŞ

BREAK/ CONTINUE DEYİMLERİ

DERS 15- DÖNGÜLER ÇIKIŞ BREAK/ CONTINUE DEYİMLERİ	
Ders Notları	Kod Uygulamaları
Break : Döngüyü sonlandırır. Kendi bloğunu	Break/Continue
Continue : Döngünün en başına döner	
Break Deyimi #Kullanıcı Adı ve Şifre Kontrol defkullanici = "Python" defparola = "1234" while True: kullanici = input("Kullanıcı Adı : ") parola = input("Parola : ") if ((kullanici != defkullanici) or (parola != defparola)): print("Yanlış Giriş !") else: print("Hoş Geldiniz ...") break	<i>#Kullanıcı Adı ve Şifre Kontrol</i> defkullanici = "Python" defparola = "1234" while True: kullanici = input("Kullanıcı Adı : ") parola = input("Parola : ") if ((kullanici != defkullanici) or (parola != defparola)): print("Yanlış Giriş !") else: print("Hoş Geldiniz ...") break
Continue Deyimi liste =[2,4] for i in range(1,10): if (i in liste): continue print(i)	liste =[2,4] for i in range(1,10): if (i in liste): continue print(i) 2 ve 4 ü geçti i =0 while i<10: if(i==2): print(i) i+=1 continue i +=1 print(i)