30.09.24

Large standard library

Arka planda çok fazla kütüphanesi mevcuttur.

Pycharm güzel bir ide’dir.

Bir stringe hangi işaretle başlarsak o işaretle bitirmemiz gerekir.

- tek tirnak -> single quotes

- cift tirnak -> double quotes

- / -> slash

- - tire - dash

- : colon iki nokta ust uste

- ; semi colon , noktali virgul

01.10.24

Derste yazılanlar mutlaka ertesi gün tekrardan yazılacak.

""Zorluklar, başarıya giden yolun gizli rehberleridir. Her engel, seni bir adım daha ileri taşır."" Çift double code olmaz hata verir. Ancak üç tane olabilir. Ya bir adet olacak ya da üç adet olacaktır.

'"Disiplin, arzularınızı hedeflerinizin önüne koymamaktır." — Abraham Lincoln' bu çalışıyor. Çünkü string olarak kabul ediyor.

Concatenation: farklı stringlerin birleştirilmesidir.

Markdown

# başlık1 ile hash işareti 6 ya kadar artırılarak başlık yazna boyut küçültülebilir.

Pep8 kuralları

3+9 # pep8 kurallarına uygun değildir. estetik yazma kuralları olarak düşünülebilir. zorunlu olmayan ancak daha okunaklı olması için gerekli olan şyedir.

3 + 9 # pep8 kurallarına uygundur. Okunurluk çok önemlidir.

Variables’ta age olması lazım age her zaman yaştır. X yaş olarak atanırsa sorun olur doğal olarak bu pep8 kuralna uygun değildir denilebilir.

Variables

İsim = aman dikkat büyük küçük harf önemlidir.

### string -> 'Jane Blue', 'USA', 'Kuru Fasulye'

### integer -> 78, 9000, -386539 (Tam Sayı)

### float -> 6.9, 35.99, -990.791 (küsuratlı sayı)

### boolean -> isRetired -> True, False

# print kullanarak

print(type(age))

print(name, age)

# tek print kullanarak variables arasına virgül koyarak yazdırılabilir.

plakaNo = camelcase

plaka\_no = snakecase

userFirstname # ilk kelime küçük harflerle yazılı daha sonra gelen kelimeler ise ilk harfleri büyük olur.

Ctrl + python da yazıları büyütür ve küçültür.

Boolean, bool

Normalde adı boolean’dır ancak python da adı bool diye geçer.

# True, False kelimeleri ilk harfi buyuk yazilmalidir pythonda.

isScheduled = false #sistem false kelimesini variable zannediyor ilk harfi kucuk oldugundan

- ctrl +c -> kopyala

- ctrl +v -> yapistir

- ctrl +y -> yeniden yap

- shıft + alt +asagi oku tusu -> uzerinde bulundugunuz satirin kopyasini cikarir

- shift + enter -> hucrelerdeki kodlari calistirmak icin kullandigimiz bir kisayol

- ctrl + a -> hepsini secmek

Öncelikle city\_2’yi variable olarak bolu olarak atıyoruz. Sonrasında yazdırıyoruz sonra variable ı sildikten sonra bile yazdırmaya devam ediyor. Çünkü ram de bir yer kapladığından dolayı devam eder. Bunun için del fonksiyonunu kullanıyoruz.

del city\_2

print(city\_2)

variable oluştururken nelere dikkat etmeliyiz

1. Sayı ile variable isimleri başlayamaz
2. \_under\_score ile ile başlayabilir.
3. @ ile başlayamaz. $ ile & başlayamaz hocam.
4. \_ underscore haricinde herhangi bir işaret kullanamıyoruz. Bu iş bu kadar hocam…..

Collection = çoklu data tutabildiğimiz bir mevzudur.

country\_1, country\_2 = "USA" , "tURKEY"

print(country\_1, country\_2)

USA tURKEY

istanbul = izmir = adana = 30

ankara = trabzon = bursa = 25

van= erzurum = 20

print(van)

tuple ve collection data tiplerine bakmak lazım. İleride bakacağız inş.

02.10.24

jupyter kullandigimizdan dolayi print fonksiyonunu kullanmasak bile type kullandigimiz zaman cikti alabiliriz

fakat alt alta bir kac tane bu sekilde cikti almaya calistigimiz zaman, sistem sadece en sondakini print eder.

Digerlerini print etmez

Type conversion olarak bakılacak

Dersi derste anlamaya bak. Sabah bakmaya fırsatın olmayabilir.

Kalıcı ve geçici değişiklik

height = str(height) olacaktır.

print(type(height) dediğimizde sonuç str olacaktır.

Bu kalıcıdır.

String integer e çevrilebilri mi?

Eğer bir string integer a benziyorsa ancak çevrilebilir. Aksi takdirde olmaz.

Float bir sayı integer a çevrilirken noktadan sonraki kısım tamamen atılır.

Olabildiğince değişken tanımlarsak daha iyi olacaktır.

Hocanın kuralı chat cpt ye yazdırılmayacak hocam

steps\_today = "5000"

steps\_today\_int = int(steps\_today)

steps\_yesterday = steps\_today\_int - 200

toplam\_adim = steps\_today\_int + steps\_yesterday

print('Toplam adım sayisi= ', toplam\_adim)

Kodlamada tekrar sistemi yorduğundan iyi değildir. Olabildiğiince kodu tekrarlamadan yazmak gerekir.

# Boolean to Int

int(True) varlık tır ve birdir.

İnt(False) yokluk tur ve sıfırdır.

- Carpma \*

- Bolme / = işleminin sonucu float tır. Tam sayı olarak gerekli ise tam bölme yanı // kullanılabilir.

- Toplama +

- Cikarma -

- Mod %

- Tam Bolme // = sonucu integer olur. Farkı budur. Adı floor division dur.

- Us alma \*\*

pow(2,2) ikinin karesi hocam.

# kökünü almak için herhangi bir fonksioyn yoktur. O yüzdenkaresinin ters mantığını almak lazım.

İşlem önceliği bu tam anlaşılmadı bununla ilgili videolara bak hocam….

1. Parantez
2. Uslu ifadeler
3. Carpma ve bölme
4. Toplama ve cıkarma

String her zaman “” arasında bulunur.

İnteger bir adet sayıdır. Ancak kenarında “” dalgası bulunmaz.

Float ise mutlaka noktalı bir sayıdır.

String bir ifadeyi integer’a çevirmek istersek ifadenin integer a benzemesi gerekir.

Aynı şekilde string ifadeyi float a çevrimek istersek ya float a ya da integer a benzemesi gerekir.

Python case sensitive dir dolayısıyla false diye bir şey olmayacaktır.

Tuple çoklu data barındırabilecek bir değerdir. Arasında virgül ile yazılır.

# bölme işlemi sonucu her daim flaot değer vereri tam bölünmesinin önemi yoktur.

03.10.24

Python kodları soldan sağa doğru okuyacaktır.

Parametre= ürettiğimiz fonksiyonları dinamik hale getirmek için kullanılır.

help(print) yazdığımız zaman size fonksiyonun ayrıntılarını döndürecektir.

print(\*args, sep=' ', end='\n', file=None, flush=False)

Prints the values to a stream, or to sys.stdout by default.

sep

string inserted between values, default a space.

end

string appended after the last value, default a newline.

Buna bakıldı. Bununla ilgili örnek çözülecektir.

file

a file-like object (stream); defaults to the current sys.stdout.

flush

whether to forcibly flush the stream.

Unicode’lar: bu şekilde bir şey vardır.

<https://www.geeksforgeeks.org/python-program-to-print-emojis/>

burada emojiler vardır hocam. Güzel bir uygulamadır.

Dikkat= parametre nedir onu tamamen öğrenmemiz gerekmektedir. En az 3-4 video izlemek gerekir.

Fonksiyon nedir ve parametre nedir. Bu konuya bakmak lazım. Nasıl yazılır öğrenmek gerekmektedir.

'\n' escape sequence demektir.

Ctrl + shift + L aynı olan yazıları seçip bu kısayolu kullandığınızda dataları aynı anda değiştirebiliriz.

F string

Format() son derece karışıktır.

age = int(input(“Lütfen isminizi giriniz: ”))

input fonksiyonu değeri string şekilde alacaktır.

Kürenin yarı çapını kullanıcıdan alacağız. Sonrasında ise

yaricap = int(input(“lütfen hacmini bulmak istediğiniz kürenin yaricapini giriveriniz: “))

kurenin\_hacmi = (4/3)\*(3.14)\*(yaricap \*\* 3)

print(f”Yarıçapı {yaricap} olan kürenin hacmi {kurenin\_hacmi} cm3’tür efenim.”)

print('Laptop,' 'Computer,' 'Phone', sep='--')

# stringlerin arasinda virgul olmadigindan seperator parametresine ihtiyac duyulmaz,

# bu sebepten -- eklenmez ,

# python bu yan yana yazilmis stringleri birlestirir cikti verir

print("a" , "b" , "c" , sep="--")

seperator parametresi tetiklenmesi için araya virgül konması gerekliliğidir. Virgülsüz seperator çalışmıyor.

04.10.24

import math dediğimizde

matematik kütüphenesini açıyoruz demektir.

Print(f”Yarıçapı {yaricap} olan bir dairenin alanı {int(alan)} cm2’dir.”)

Print(f”Yarıçapı {yaricap} olan bir dairenin alanı {round(alan, 2)} cm2’dir.”)

\ bastığımızda bu kaçış manevrasıdır. Bunu yani string ten say diyoruz.

print('tom\'s shoes') burada tek tırnağı string olarak kabul eyle hocam diyoruz.

print('tomun \n ayakkabıları \n cok güzeldir.')

çıktısı:

tomun

ayakkabıları

cok güzeldir.

Şeklindedir.

Ödev1 \t

Sep = ‘b’

Comparison operators

<

>

>=

<=

== karşılaştırma operatörüdür.

= atama operatörüdür.

!= eşit değil midir diye sormaktır.

# database = veritabanı

sifre = 'hope7.\*'

user\_password = input('şifrenizi giriniz lütfen:')

print(sifre == user\_password) # eşit midir diye soruyor.

İnput dataları string olarak alır.

password = 123

web\_pass = int(input('şifreyi firin'))

print(password == web\_pass)

True > 0 ve False < 1 bu tarz örneklere bakılacaktır.

Ascii values

String kıyaslaması yapılabilir mi?

Ascii nerde işimize yarar? Programın çalışma mantığı bu şekildedir.

Ord diye bir fonksiyon mevcuttur.

ord(‘m’) ascii değerini gösterir yani bir çeşit ebcet hesabı sonuç 109’dur.

Bunun tersi ise chr(65) tir o nun da sonucu A’dır.

#### Falsy and Truthy Values

'ahmet'

'mehmet'

45

0

56

# Bunların hepsinin birer boolean dereği var arkadaşlar.

bool('murat') sonucu True’dur.

print(f' 0 {bool(0)}')

print(f' 0.0 {bool(0.0)}')

print(f' False {bool(False)}')

print(f"' {bool('')}")

bunların neticesi False tur.

And or Logical Expressions

6 and 9 and “melisa” sonucu melisa dır.

# and

# hepsi true ise sondaki elemanı yazdırır.

# bir tane bile false varsa ilk gördügü false i yazdırır.

# bütün şartların sağlandığı durumlarda and kullanacağız.

# mükemmelliyetçidir.

Or operatörü çok iyimser dir tek bir tane True varsa onu döndürür. Ama eğer truthy değer yoksa o zaman en sondaki değeri döndürür.

8 or 0 or ‘ankara’

Or true değeri arar and ise false değeri arar

True and 0 or False or 45 and 3

45 and 3

3

Bunun sonucunu bulmaya çabala hocam

Matematikteki mantık konusu ve veya değil dir.

7 and 0 or False neticesi ne olur?

False or (False and ‘Melisa’ or not ‘ ’ and (0 or 0.1))

Bunun neticesini bulana kadar çalışmak gerekmektedir.

# kutlesi(kg) ve hizini(m/s) kullanicidan aldiginiz bir cismin kinetik enerjisini hesaplayiniz

# Kullanicinin boy ve kilo degerlerini alip kullanicinin bmi degerini hesaplayin

('Ankara' or 'Istanbul' and 0 and not '') or not (True and 679 or '')

Cismin\_kutlesi = m = int(input(“Lütfen cismin kutlesini kg cinsinden giriniz: “)

Cismin\_hizi = v = int(input(“Lütfen cismin hizini metre/saniye cinsinden giriniz: “)

Kinetik\_enerji = ½ \* cismin\_kutlesi \* cismin\_hizi \*\*2

# and = iki veya daha fazla koşulun True olup olmadığını kontrol eder

# or = en az bir koşulun True olup olmadığını kontrol eder

# not = Koşul False ise True, ve tam tersi

('Ankara' or 'Istanbul' and 0 and not '') or not(True and 679 or '')

'Ankara' or 'Istanbul' = ‘Ankara’

05.10.24

sayi = 1

print(f"{sayi} karesi= {sayi \*\* 2}")

print(f"{sayi+1} karesi= {(sayi+1) \*\* 2}")

print(f"{sayi+2} karesi= {(sayi+2) \*\* 2}")

print(f"{sayi+3} karesi= {(sayi+3) \*\* 2}")

print(f"{sayi+4} karesi= {(sayi+4) \*\* 2}")

print(f"{sayi+5} karesi= {(sayi+5) \*\* 2}")

print(f"{sayi+6} karesi= {(sayi+6) \*\* 2}")

print(f"{sayi+7} karesi= {(sayi+7) \*\* 2}")

print(f"{sayi+8} karesi= {(sayi+8) \*\* 2}")

print(f"{sayi+9} karesi= {(sayi+9) \*\* 2}")

hackerrank