# -\*- coding: utf-8 -\*-

"""Day\_4.ipynb

Automatically generated by Colaboratory.

Original file is located at

https://colab.research.google.com/drive/1h4zQ6YOlemX8Rsf8G\_bR30vFqyZ-4-Gs

##Soru

Kullanıcıdan bir tamsayı değeri (stop\_number) giren ve 0'dan stop\_number'a kadar tüm sayıların toplamını yazdıran bir program yazın. Kullanıcının geçerli bir değer gireceğini varsayabilirsiniz.

"""

stop\_number = int(input("Bir bitiş sayısı giriniz: "))

sonuc = 0

for i in range(1, stop\_number):

sonuc += i

print(sonuc)

"""##Soru

Programınızı, toplanacak sayıların başlangıç değerini de (start\_number) girecek şekilde genişletin. Bu durumda, programınız start\_number dan stop\_number a kadar tüm sayıları toplayacaktır. Kullanıcının iki geçerli değer gireceğini varsayabilirsiniz.

"""

start\_number = int(input("Bir başlangıç sayısı giriniz:"))

stop\_number = int(input("Bir bitiş sayısı giriniz: "))

sonuc = 0

for i in range(start\_number, stop\_number):

sonuc += i

print(sonuc)

"""##Soru

Programınızı, start\_number ı 0 ile 100 arasında bir tamsayı olup olmadığını kontrol edecek şekilde genişletin. Eğer öyleyse, programınız ikinci girdiyi sormaya ve hesaplamaları yapmaya devam edecektir; aksi takdirde, bir hata yazdırır ve yürütmeyi durdurur.

"""

start\_number = int(input("Bir başlangıç sayısı giriniz:"))

if 0 <= start\_number <= 100:

stop\_number = int(input("Bir bitiş sayısı giriniz: "))

if start\_number <= stop\_number <= 100:

sonuc = 0

for i in range(start\_number, stop\_number):

sonuc += i

print(sonuc)

else:

print("Geçersiz bitiş sayısı")

else:

print("Geçersiz başlangıç sayısı")

"""##Soru

Bilgisayar 1 ile 100 arasında rastgele bir sayı seçsin, biz de onu tahmin edecek programı yazalım.

###Modüller

Bazı işlevleri kolaylıkla yerine getirmemizi sağlayan birtakım fonksiyonları ve nitelikleri içinde barındıran araçlardır. Barındırdığı bu fonksiyonları farklı dosyalar ve programlar içinde tekrar tekrar kullanma imkanı sağlar. Modüller sayesinde Python çok daha kullanışlı ve uygulaması kolay bir dil haline gelmiştir.

Python’da herhangi bir modülü kullanabilmek için öncelikle onu ‘içe aktarmamız’ gerekir. İçe aktarmak, bir modül içindeki fonksiyon ve nitelikleri başka bir program içinden kullanılabilir hale getirmek demektir.

\*\*import <modül\_adı>\*\*

"""

import random as rnd

gizli = rnd.randint(1, 100)

check = False

tahmin = int(input("Bir tahmin giriniz:"))

print(gizli)

for x in range(5):

tahmin = int(input("Bir tahmin giriniz:"))

if tahmin == gizli:

print("Tebrikler")

check = True

break

elif tahmin < gizli:

print("Daha büyük bir sayı giriniz!!")

else:

print("Daha küçük bir sayı giriniz!!")

if not check:

print("Bilemediniz, belirlenen sayı: ", gizli)

"""#Sözlükler (Dictionary)

\* Python'daki bir veri tipi.

\* Sözlükler,anahtarlar (keys) tarafından indexlenir ve bu anahtarlar String ve sayı tipinde olabilir.

\* key:value şeklinde yazılır ve key ler benzersiz (unique) değerlerdir.

\* {}

"""

d = {}

"isim"-"fethi"

"soyisim"-"tekyaygil"

"yaş"-24

1-"okunan okul sayısı"

"isim2"-"fethi"

print(d)

{"Key":"Value"}

d = {"python":1, "course":2}

print(d)

print(d["python"])

d2 = {"machine":"learning", "artificial":"intelligence" }

d2

d2["artificial"]

d["course"]

d2

d2["deep"] = "12"

d2

#print(d2)

print(d2.keys())

sorted(list(d2.values()))

d2.items()

"""for k in d2.keys():

print(k)"""

[print(i) for i in d2.keys()]

print(d)

for v in d.values():

print(v)

[print(i) for i in d.values()]

d2.items()

for key,value in d2.items():

print(key,value)

d.items()

for k,v in d.items():

if v == 2:

print(k)

print(d)

d["a"] = [3,4,5]

d

d.pop("course")

d

len(d)

"a" in d

"python" in d

d2

d2.get("artificial")

print(d2.get("asli"))

d2

del d2["deep"]

d2

d4 = {"insan":2, "kedi":4, "örümcek":8}

"""

for i in d4:

ayak = d4[i]

print("%s canlısının %d adet ayağı bulunur" % (i,ayak))

print(str(i) + " canlısının " + str(ayak) +" adet ayağı bulunur.")

"""

for key,value in d4.items():

print("%s canlısının %d adet ayağı vardır" % (key,value))

import math as m

f=m.floor(18.7)

f

c=m.ceil(18.7)

c

"""

i=17

flo=float(i)

flo

"""

d4 = {"insan":2, "kedi":4, "örümcek":8}

for i in d4:

print(i)

d4 = {"insan":2, "kedi":4, "örümcek":8}

for i,ayak in d4.items():

print(str(i) + " canlısının " + str(ayak) +" adet ayağı bulunur.")

ilceler = {"İstanbul":["Bostancı", "Beşiktaş", "Kadıköy"],

"Ankara":["Çankaya", "Gölbaşı", "Kızılcahamam"],

"İzmir":["Çeşme","Bornova","Foça"]}

ilceler["İstanbul"]

type(ilceler)

type(ilceler["Ankara"])

ilceler["İzmir"]

ilceler["İzmir"][1:3]

ilceler["İzmir"][2]

#listeden sözlük oluşturma

0-1-2-3-4-5-6-7-8

sayilar = list(range(9))

"""

{"key":"value"}

continue

"""

cift\_kare = {x: x\*\*2 for x in sayilar if x % 2 == 0}

print(cift\_kare)

"""#Kümeler (Sets)

\* Bir küme, listelerden farklı olarak, belli bir sırası bulunmayan bir veri topluluğudur; elemanlarına indexleme ile erişilemez.

\* Tıpkı matematiksel kümeler gibi, aynı elemandan birden fazla barındıramaz.

\* Kümeler içerisine bir öğe ancak bir defa eklenebilir. Yani bir küme iki

aynı elemana sahip olamaz.

\* Boş küme oluşturmak için set() fonksiyonunu kullanıyoruz.

"""

{"Key":"Value"}

s = {"python", 5,6,8,5,6,"abc", "python"}

s

l=[1,2,3,4,5,3,2]

print(type(list(set(l))))

bos = set()

bos

bos2 = {}

type(bos2)

s2 = set(["python", 5,6,8,5,6,"abc", "python"])

print(s2)

an = set("ananas")

print(an)

y = {"a","b",[1,2,3],5,6}

print(y)

s2

6 in s2

9 in s2

len(s2)

list.append

list[4]=""

dict["yeni"]=yenival

s2.add("ai")

len(s2)

s2.add("ai")

print(s2)

len(s2)

s2.remove("ai")

print(s2)

from math import sqrt #squareroot karekök

#import math

print(sorted(list({sqrt(x) for x in list(range(10))})))

"""# Demetler(Tuples)

\* Sıralı veri yapısıdır, listeler gibi index değerine sahiptir, tüm veri tiplerini içerisinde barındırabilir.

\* Listelerden farklı olarak, değiştirilemeyen bir yapıları vardır.Elimizde birden fazla değiştirilemez değerimiz var ise, bunları bir demet içinde toplayabiliriz.

"""

demet = (1,2,3,4,5)

print(demet)

print(type(demet))

{}

set()

demet2 = () #boş bir listeyi de, liste = [] olarak tanımlayabiliyorduk.

type(demet2)

demet

print(demet[3])

demet[-2]

demet[:4]

dm3 = ("asli", 5, 8, "eylul")

dm3.index("eylul")

dm3.count(5)

dm4 = ("elma","armut","çilek")

dm4[0] = "kiraz"

print(dm4)

dm4.remove("armut")

# demetlerin anahtar olduğu bir sözlük oluşturalım.

sozluk = {(x, x+1): x for x in range(10)}

print(sozluk)

demet5 = (5,6)

print(type(demet5))

sozluk[demet5]

sozluk[(3,4)]