



# پایتون

جلسه ہفتم

ذبح

بہار ۱۴۰۳

# فهرست مطالب

بازگردانی متغیر در توابع

01

متغیر های لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

# ساختار توابع در پایتون :

اسم دلخواه برای تابع

دستور ثابت برای ایجاد توابع

```
def hello():  
    print('hello world')
```

دستورات مد نظر برای تابع

نکته : توابع به صورت خودکار در برنامه اجرا نخواهند شد و باید آنها را فراخوانی کنیم.

نکات کاربردی

01

توابع داخلی

02

توابع در پایتون

03

تمرین

04

مثال 1 : تابعی بنویسید که متن hello world را چاپ کند. سپس تابع را فراخوانی کنید و خروجی را ملاحظه کنید.

```
1 def hello():  
2     print('hello world')  
3  
4 hello()
```

hello world

خروجی:

نکات کاربردی

01

توابع داخلی

02

توابع در پایتون

03

تمرین

04

## مثال 2 : تابعی بنویسید که نام شما را به عنوان ورودی دریافت کند و سپس آن را چاپ کند.

```
1 def my_function(name):  
2     print(f'I am {name}')  
3  
4 my_function("iman")
```

I am iman

خروجی:

نکات کاربردی

01

توابع داخلی

02

توابع در پایتون

03

تمرین

04

## انواع توابع:

1- توابعی که ورودی ای دریافت نمیکنند و خروجی ای هم بر نمیگردانند

نکات کاربردی

01

2- توابعی که ورودی ای دریافت نمیکنند اما خروجی بر میگردانند.

توابع داخلی

02

3- توابعی که ورودی دریافت می کنند اما خروجی ای بر نمی گردانند.

توابع در پایتون

03

4- توابعی که ورودی دریافت کرده و خروجی بر می گردانند.

تمرین

04

مثال 3 : برنامه ای بنویسید که ابتدا سن شما را دریافت کند سپس تابعی به نام `convert` تعریف کنید و در تابع سن را به روز تبدیل کند و چاپ نماید.

نکات کاربردی

01

توابع داخلی

02

توابع در پایتون

03

تمرین

04

```
1 def convert(sen):
2     be_rooz = sen * 365
3     print("Sen shoma be rooz: ", be_rooz)
4
5 x = int(input("enter your age: "))
6 convert(x)
```

```
enter your age: 20
Sen shoma be rooz: 7300
```

خروجی:

01	نکات کاربردی
02	توابع داخلی
03	توابع در پایتون
04	تمرین

مثال 4: برنامه ای بنویسید که یک تابع به نام ضرب با دو پارامتر ورودی  $x$  و  $y$  داشته باشد سپس در برنامه ی اصلی  $x$  و  $y$  را بخوانید و با صدا زدن تابع ضرب آنها را در هم ضرب کنید.

```
1
2 def zarb(x, y):
3     z = x * y
4     print(z)
5
6 addad1 = int(input("addad aval: "))
7 addad2 = int(input("addad dovom: "))
8
9 zarb(addad1, addad2)
```

نکات کاربردی

01

توابع داخلی

02

توابع در پایتون

03

تمرین

04

مثال 5 : برنامه ای بنویسید که نام و فامیل و سن شما را در برنامه ی اصلی بخواند. سپس تابعی به نام سن تعریف کنید که سن شما را به ماه حساب کند و چاپ کند. همچنین تابعی به نام mohaseb تعریف کنید که طول نام و طول فامیل شما را محاسبه کند و چاپ کند.

```
1 def sen(x):
2     z = x * 12
3     print(z)
4
5 def mohaseb(esm , famil):
6     print(len(esm))
7     print(len(famil))
8
9     firstname = input("esm ra vared kon: ")
10    lastname = input("famil ra vared kon: ")
11    age = int(input('sen ra vared kon: '))
12
13    sen(age)
14    mohaseb(firstname, lastname)
15
16
```

نکات کاربردی

01

توابع داخلی

02

توابع در پایتون

03

تمرین

04

**مثال 6 : برنامه ای بنویسید که ابتدا یک عدد را دریافت کند سپس تابعی به نام دلخواه تعریف کنید و در فاکتوریل آن عدد را محاسبه و سپس چاپ کند.**

نکات کاربردی

01

توابع داخلی

02

توابع در پایتون

03

تمرین

04

```
1 def factor(num):
2     f = 1
3     for i in range(1, num + 1):
4         f = f * i
5
6     print("factoriel: ", f)
7
8 x = int(input("adad ra vared kon: "))
9 factor(x)
```

```
adad ra vared kon: 5
factoriel: 120
```

خروجی:

نکات کاربردی

01

توابع داخلی

02

توابع در پایتون

03

تمرین

04

**نکته:** با استفاده از دستور return می توانیم داده ی مد نظر خود را از تابع به برنامه ی اصلی برگردانیم.

**مثال 7:** ابتدا تابعی بنویسید که دو عدد را به عنوان ورودی دریافت کند و آنها را در هم ضرب کند و به عنوان خروجی به برنامه ی اصلی برگرداند سپس در برنامه ی اصلی خروجی را چاپ کند

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

```
1 def zarb(x, y):
2     m = x * y
3     return m
4
5 num1 = int(input("addad aval ra vared kon: "))
6 num2 = int(input("addad dovom ra vared kon: "))
7
8 hasel = zarb(num1, num2)
9 print("Hasel Zarb: ", hasel)
```

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

**نکته: همانطور که تا الان متوجه شدید توابع تا زمانی که آنها را فراخوانی نکنیم اجرا نخواهند شد. و می توانیم هر زمان که خواستیم در برنامه به هر مقداری که لازم است تابع خود را فراخوانی کنیم.**

نکات کاربردی

01

توابع داخلی

02

توابع در پایتون

03

تمرین

04

مثال 8: دو عدد به نام های num1 و num2 از کاربر دریافت و سپس num1 را به توان num2 برساند. سپس نتیجه را با تابع pow مقایسه کنید. نام تابع خود را tavan قرار دهید و تابع از نوع چهارم باشد.

```
temp.py* x
1 num1 = int(input('Enter Num1?'));
2 num2 = int(input('Enter Num2?'));
3 def tavan(num1,num2):
4     t = num1**num2;
5     return t
6 print(tavan(num1,num2));
7 powNum = pow(num1,num2);
8 print('pow' , powNum);
9
```

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

```
temp.py* x
1 num1 = int(input('Enter Num1?'));
2 num2 = int(input('Enter Num2?'));
3 def tavan(num1,num2):
4     t = num1**num2;
5     return t
6 print(tavan(num1,num2));
7 powNum = pow(num1,num2);
8 print('pow' , powNum);
9
```

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

```
temp.py* x
1 num1 = int(input('Enter Num1?'));
2 num2 = int(input('Enter Num2?'));
3 def tavan(num1,num2):
4     t = num1**num2;
5     return t
6 print(tavan(num1,num2));
7 powNum = pow(num1,num2);
8 print('pow' , powNum);
9
```

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

همانطور که در مثال بالا می بینید دو عدد به نام های num1 و num2 از کاربر گرفته شده و سپس در لاین 8 ام کد به تابع zarb به عنوان ورودی داده و تابع فراخوانی شده.

و در آخر حال ضرب دو عدد در لاین 3 ام return شده و حال مقدار متغیر hasel برابر با حاصل ضرب دو عدد (m) خواهد بود.

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

مثال 9: ابتدا تابعی بنویسید که به عنوان ورودی یک رشته را دریافت کرده و تعداد کرکتر های آن رشته را محاسبه و به برنامه اصلی برگرداند. سپس در برنامه ی اصلی از کاربر اسمش را دریافت کنید و تابع را برای اسم فراخوانی کرده و خروجی تابع را چاپ کنید.

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

```
1  ✓ def worker(txt):
2      |     m = len(txt)
3      |     return m
4
5  x = input("esm khod ra vared kon: ")
6
7  tedad = worker(x)
8  print("Tedad horoof: ", tedad)
```

```
esm khod ra vared kon: erfane
Tedad horoof: 5
```

خروجی:

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

**نکته: همانطور که می توانیم به میزان مدنظر برای تابعی ورودی تعریف و دریافت کنیم. برای خروجی نیز میتوانیم به هر میزان تعریف و return کنیم.**

**مثال 10: در برنامه ی خود تابعی بنویسید که 2 عدد را به عنوان ورودی دریافت کند سپس حاصل ضرب و حاصل جمع آن دو عدد را به برنامه ی اصلی بازگرداند. و در برنامه ی اصلی از کاربر دو عدد دریافت کنید و تابع را برای آن دو عدد فراخوانی کرده و سپس خروجی را چاپ کنید.**

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

```
1 def worker(x , y):
2     z = x * y
3     j = x + y
4     return z, j
5
6 adad1 = int(input("adad aval ra vared kon: "))
7 adad2 = int(input("adad dovom ra vared kon: "))
8
9 zarb, jame = worker(adad1, adad2)
10 print(f"Hasel zarb: {zarb}\nHasel jame: {jame}")
```

```
adad aval ra vared kon: 8
adad dovom ra vared kon: 6
Hasel zarb: 48
Hasel jame: 14
```

خروجی:

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

مثال 11: تابعی به نام mohaseb تعریف کنید به طوری که سه متغیر از کاربر دریافت میکند (در برنامه ی اصلی به نام  $x, y, z$ ) سپس حاصل ضرب هر جفت عدد در تابع محاسبه میکند و به برنامه ی اصلی بر می گرداند.

$$A = x * y$$

$$B = x * z$$

$$C = y * z$$

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

**مثال 12: برنامه ای بنویسید و در برنامه ی خود دو تابع تعریف کنید که تابع اول یک ورودی دریافت کرده و فاکتوریل آن را حساب کرده و به برنامه ی اصلی برگرداند و تابع دوم نیز یک عدد ورودی دریافت کرده و حاصل جمع از 1 تا عدد را حساب کرده و به برنامه ی اصلی برگرداند.**

**سپس از کاربر یک عدد دریافت کرده و هر دو تابع را برای آن فراخوانی کنیم و سپس خروجی ها را چاپ کنید.**

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

```

1 def factor(x):
2     f = 1
3     for i in range(1, x + 1):
4         f = f * i
5     return f
6
7 def jame(x):
8     s = 0
9     for i in range(1, x + 1):
10        s = s + i
11    return s
12
13
14 adad = int(input("adad ra vared kon: "))
15
16 factorial = factor(adad)
17 majmoo = jame(adad)
18 print(f"Factorial {adad}: {factorial}")
19 print(f"Hasel jame ({1} ta {adad}): {majmoo}")
20

```

```

adad ra vared kon: 5
Factorial 5: 120
Hasel jame (1 ta 5): 15

```

خروجی:

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

مثال 13: برنامه ای بنویسید به کمک توابع که یک عدد را دریافت کند و تعداد ارقام آن را چاپ کند در تابع اول با نام (targham) و نام تابع دوم ما (targham2) سپس در تابع دیگری مجموع ارقام آن را چاپ کند؟

01 بازگردانی متغیر

02 لوکال و گلوبال

03 مثال های ترکیبی

04 تمرین

```
def targham(x):  
    return len(str(x))  
  
def targham2(x):  
    c = 0 # shomarande  
    while x > 0:  
        x = int(x / 10)  
        c = c + 1  
    return c  
  
def majmoe(x):  
    sum = 0;  
    while x > 0:  
        b = x % 10  
        sum = sum + b  
        x = int(x / 10)  
    return sum  
x = int(input('enter Num?'))  
print('targham1 => ',targham(x))  
print('targham2 => ',targham2(x))  
print('majmoe adad => ',majmoe(x))
```

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04



مثال 14: برنامه ای بنویسید و در برنامه ی خود یک تابع تعریف کنید که سن را به سال به عنوان ورودی دریافت کرده و تبدیل به ماه هفته و روز بکند و خروجی را به برنامه اصلی برگرداند. از کاربر سنش را به سال بگیرد و تابع را برای سن کاربر فراخوانی و سپس خروجی را چاپ کند.

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

```
1  def sen(year):
2      month = year * 12
3      week = round(365 / 7 * year)
4      day = year * 365
5      return month, week, day
6
7
8  x = int(input("sen khod ra vared konid: "))
9
10 to_months, to_weeks, to_days = sen(x)
11 print("Sen be mah: ", to_months)
12 print("Sen be hafte: ", to_weeks)
13 print("Sen be rooz: ", to_days)
```

```
sen khod ra vared konid: 20
Sen be mah: 240
Sen be hafte: 1043
Sen be rooz: 7300
```

خروجی:

بازگردانی متغیر

01

لوکال و گلوبال

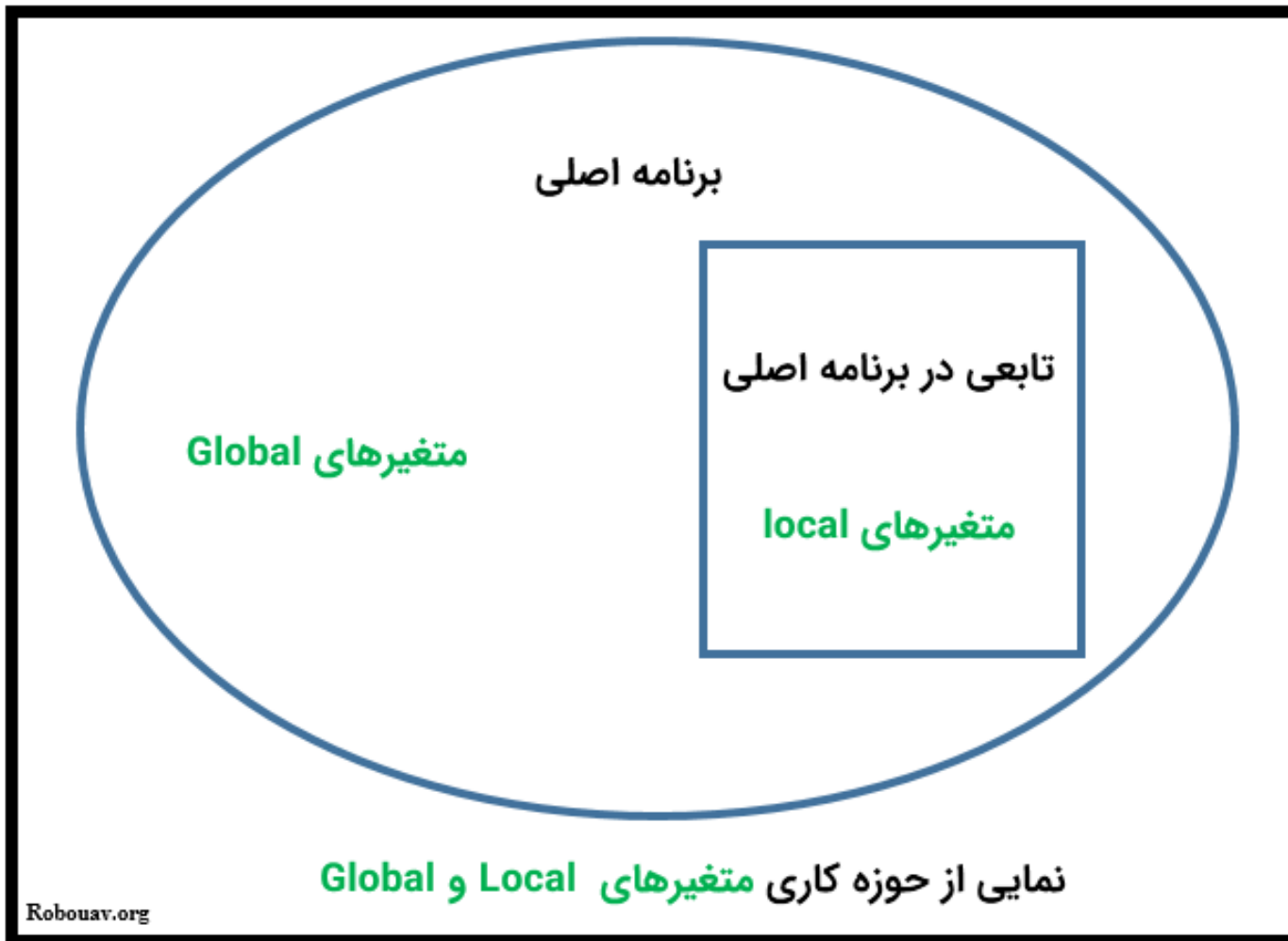
02

مثال های ترکیبی

03

تمرین

04



- 01 بازگردانی متغیر
- 02 لوکال و گلوبال
- 03 مثال های ترکیبی
- 04 تمرین

برای درک بهتر این دو مفهوم فرض کنید یک محله داریم و در آن محله آقای "محمدی" فرد معروفی است. آقای محمدی اگر به یک سوپر مارکتی برود ، همه او را می شناسند و به او نسبه می دهند. چون یک فرد آشنا است میتواند با تمامی افراد آن محله سلام و علیک داشته باشد.

اما اگر همین آقای محمدی به خارج از محله برود و بخواهد از سوپر مارکت های شهر نسبه بگیرد. به نظر شما به او نسبه می دهند ؟ قطعه نه ! چون او را نمی شناسند. اگر به افراد غریبه سلام کند ممکن است کسی پاسخ سلام او را ندهد. علت این است که آقای محمدی صرفا در محله خود آقای محمدی است و در جاهای دیگر کسی او را نمی شناسد.

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

اما فرض کنید که یک فرد مشهور داریم. مثلا "مهران مدیری" را در نظر بگیرید. به این فرد در تمامی شهر نسبه می دهند. چون همه او را می شناسند. علاوه بر آن اگر آقای مدیری به محله آقای محمدی برود نیز ، همچنان میتواند از مغازه داران نسبه بگیرد و به همه سلام دهد. چون تمامی افراد آن محله نیز آقای مدیری را می شناسند.

مثال بالا دقیقا مفهوم متغیرهای local و global را بیان می کند. در مثال بالا آقای محمدی نقش یک متغیر local و آقای مدیری نقش یک متغیر global را دارد.

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

به عنوان مثال در کد زیر متغیر `name` یک متغیر گلوبال (global) است و در هر جای برنامه که بخواهیم می توانیم به آن دسترسی داشته باشیم. اما متغیر `age` که در داخل تابع تعریف شده یک متغیر لوکال (local) است و فقط در داخل تابع `test` می توانیم به آن دسترسی داشته باشیم یا تغییراتی در آن ایجاد کنیم.

```
1 def test():
2     |     age = 20
3
4     name = "ali"
```

مثال: کدهای بالا را در برنامه ی خود بنویسید و یکبار در تابع تست متغیر `name` را چاپ کنید و تابع را در برنامه فراخوانی و خروجی را ملاحظه کنید.

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

مثال 15: در همان برنامه ی قبلی این بار متغیر age را در برنامه ی اصلی و خارج از تابع چاپ کنید و خروجی را ملاحظه کنید.

```
1 def test():
2     age = 20
3     print(name)
4
5 name = "ali"
6 test()
7
8 print(age)
```

خروجی:

```
NameError: name 'age' is not defined
```

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

همانطور که دیدید برنامه برای چاپ age در خارج از تابع ارور داد و این به این معنیست که متغیر age و تمامی متغیر هایی که داخل توابع تعریف می شوند لوکال هستند

**نکته: لوکال بودن متغیر های تابع مزایایی من جمله اشغال کمتر حافظه رم و ایجاد سطح دسترسی و بهینه بودن و ... برنامه را دارد.**

**اما در صورتی که نیاز باشد متغیر لوکالی را تبدیل به گلوبال کنیم تا در تمام برنامه قابل دسترسی باشد میتوانیم از دستور global استفاده کنیم.**

**مثال: در برنامه ی قبلی این بار متغیر age را گلوبال کرده تابع را فراخوانی و سپس در خارج از تابع دوباره چاپ کنید و نتیجه را ملاحظه کنید.**

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

```
1  def test():
2      global age
3      age = 20
4      print(name)
5
6  name = "ali"
7  test()
8  print(age)
```

ali  
20

خروجی:

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

همانطور که می بینید این بار متغیر `age` با موفقیت چاپ شد.

توجه: دستور `global` حتما باید قبل از تعریف متغیر صدا زده شود (مانند کد بالا) و در غیر این صورت ارور خواهد داد.

## چند نکته:

1- حتی در صورت گلوبال شدن متغیرهای تابع اگر تابع را در برنامه فراخوانی نکنیم به ارور بر خواهیم خورد!

2- استفاده از دستور global از بهینه بودن کد می‌کاهد و ترجیحا اگر الزامی نیست سعی کنید از آن استفاده نکرده و از دستور return استفاده کنید.

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

مثال 16: برنامه ای بنویسید که حاصل جمع ارقام یک عدد دریافتی از کاربر را در تابعی حساب کند و به برنامه اصلی بازگرداند و سپس در برنامه ی اصلی خروجی چاپ شود.

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

```
1 def worker(num):
2     s = 0
3     while num > 0:
4         b = num % 10
5         num = num // 10
6         s = s + b
7     return s
8
9
10
11 x = int(input("adad ra vared kon: "))
12 majmoo = worker(x)
13 print("Majmoo argham: ", majmoo)
```

```
adad ra vared kon: 10024
Majmoo argham: 7
```

خروجی:

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

مثال 17: برنامه ای بنویسید که تعداد هر حرف صدا دار اسم دریافتی از کاربر را داخل تابعی بدست آورده و نتیجه را در برنامه ی اصلی چاپ کند.

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

```
1
2 def worker(txt):
3     seda = ["a", "e", "i", "o", "u"]
4     found = {}
5     for i in seda:
6         found[i] = 0
7
8     for i in txt:
9         if i in seda:
10            found[i] += 1
11    return found
12
13 x = input("esm khod ra vared konid: ")
14 f = worker(x)
15 for i in f:
16     print(f"{i} found {f[i]} times")
```

```
esm khod ra vared konid: iman zabah
a found 3 times
e found 0 times
i found 1 times
o found 0 times
u found 0 times
```

خروجی:

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

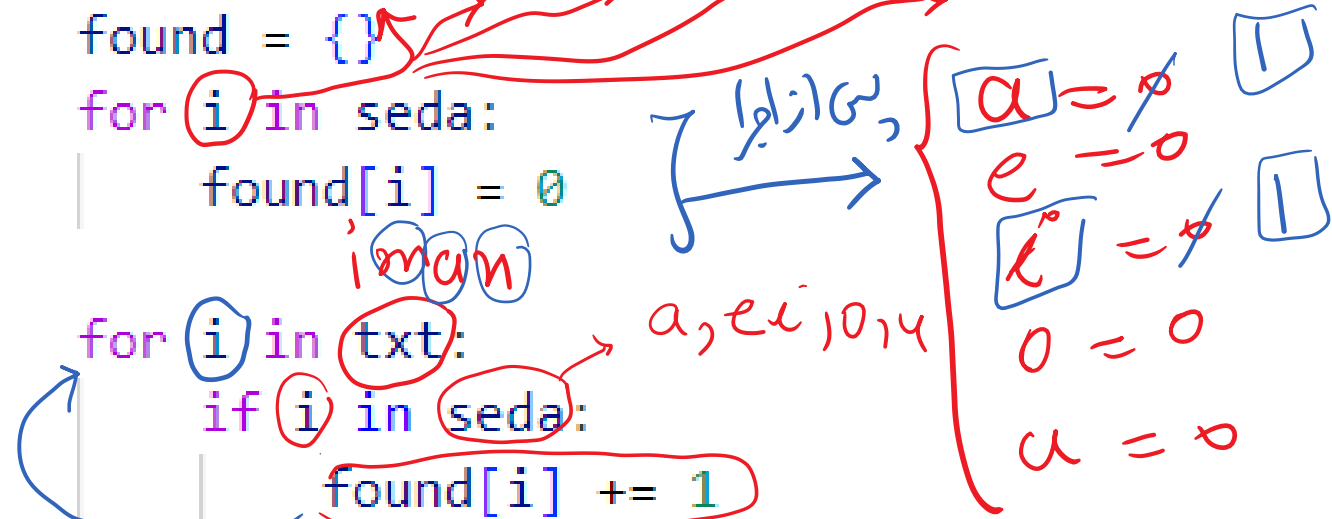
تمرین

04

```

1
2 def worker(txt):
3     seda = ["a", "e", "i", "o", "u"]
4     found = {}
5     for i in seda:
6         found[i] = 0
7
8     for i in txt:
9         if i in seda:
10            found[i] += 1
11    return found
12
13 x = input("esm khod ra vared konid: ")
14 f = worker(x)
15 for i in f:
16     print(f"{i} found {f[i]} times")

```



بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

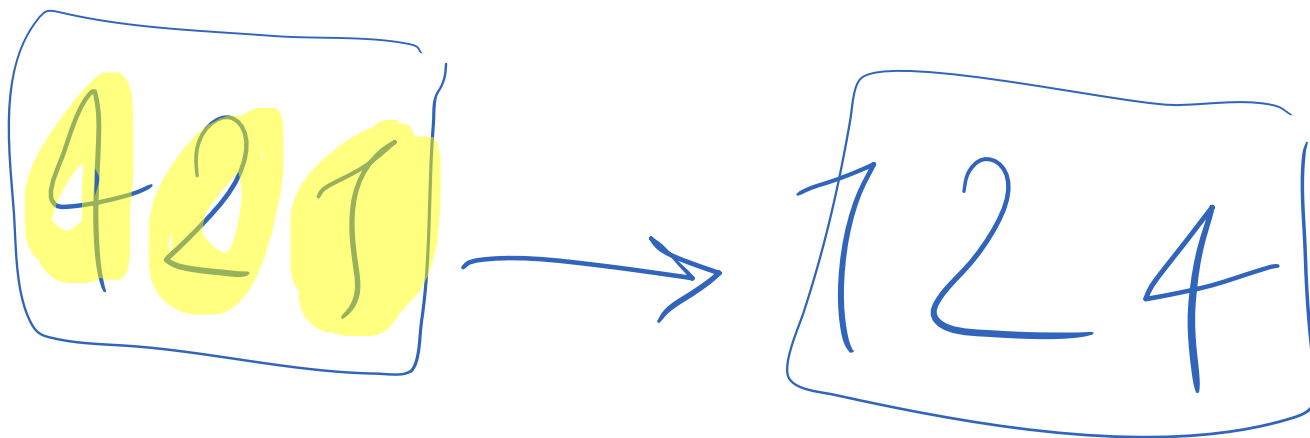
تمرین

04

مثال 18: برنامه ای بنویسید که وارون عدد دریافتی از کاربر را در تابعی محاسبه کند و خروجی را در برنامه ی اصلی چاپ کند.

421 --> 124  
1008 --> 8001

مثلا:



بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

۳ ۲ ۵

سید و سید و سید →

۳ ۲ ۵ ۲

سه هزار و سیصد و پنجاه →

۳ ۲ ۵ ۲ ۶ →

۳ ۲ ۵ ۲ ۶ ۳ ۴ ۵ ۶ ۷ ۸ ۹ ۱۰

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

## روش اول:

```
1 def worker(num):
2     temp = []
3     while num > 0:
4         b = num % 10
5         num = num // 10
6         temp.append(str(b))
7
8     varoon = ""
9     for i in temp:
10        | varoon = varoon + i
11
12        varoon = int(varoon)
13        return varoon
14
15 x = int(input("esm khod ra vared konid: "))
16 v = worker(x)
17 print(f"Varoon {x}: {v}")
```

```
esm khod ra vared konid: 123098
Varoon 123098: 890321
```

خروجی:

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

```
1  ✓ def worker(num):
2      temp = list(str(num))
3      temp.reverse()
4      varoon = ""
5
6  ✓  for i in temp:
7      |     varoon = varoon + i
8      |     varoon = int(varoon)
9
10     return varoon
11
12  x = int(input("esm khod ra vared konid: "))
13  v = worker(x)
14  print(f"Varoon {x}: {v}")
```

```
esm khod ra vared konid: 123098
Varoon 123098: 890321
```

خروجی:

## روش دوم:

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

```
2 x = int (input('adad ra vared kon: '))
3 temp = []
4 for i in str(x):
5     temp.append(i)
6
7 temp.reverse()
8 varoon = ""
9 for i in temp:
10     varoon = varoon + i
11
12 print("varoon: ", int(varoon))
13
```

```
esm khod ra vared konid: 123098
Varoon 123098: 890321
```

خروجی:

روش سوم:

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

**نکته:** در روش دوم مثال 3 در لاین دوم با استفاده از تابع داخل str ابتدا عدد را به رشته تبدیل و سپس با استفاده از تابع داخلی list رشته را به لیست تبدیل کردیم و بدین ترتیب تمامی ارقام عدد تبدیل به آیتم هایی در لیست temp شدند.

سپس با استفاده از تابع reverse() لیست را معکوس کرده و سپس آن را بهم به صورت رشته چسبانندیم و در نهایت عدد وارون شده را به int تبدیل کردیم.

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

تمرین

04

**تمرین 1: برنامه ای بنویسید که موارد مطلوب زیر را درباره ی عدد دریافتی از کاربر در داخل توابع جدا چک کند و نتایج را چاپ کند.**

- 1- تعداد ارقام عدد**
- 2- اول بودن یا نبود عدد**
- 3- زوج یا فرد بودن عدد**

بازگردانی متغیر

01

لوکال و گلوبال

02

مثال های ترکیبی

03

**تمرین**

**04**

# پایان جلسه هفتم

---