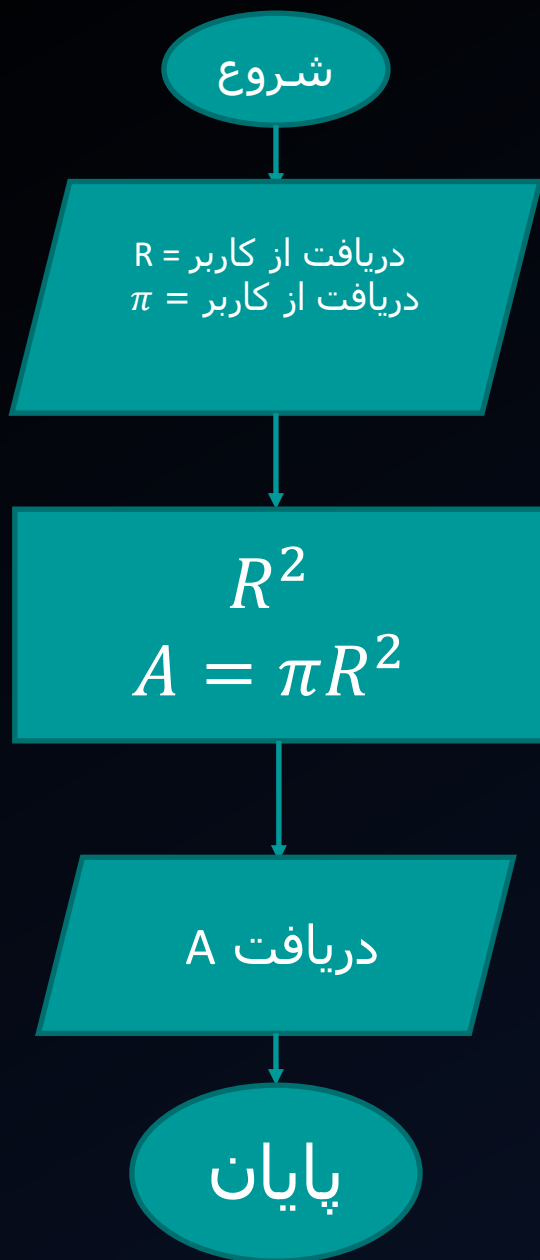


2. Operators and Variables

2.1 Variables



متغیر چیست ؟

به زبان ساده متغیر ظرفی است برای نگه داشتن برخی اطلاعات



در این مثال R و A و π متغیر های ما هستند
یعنی هر مقدار ممکن می توانند داشته باشند
و در حافظه کامپیوتر مقدار دلخواه ما به آنها نسبت داده میشود

حال با فراخوانی R و A و π در هر جای دلخواه از برنامه
این مقادیر در اختیار ما قرار خواهند گرفت

یک متغیر میتواند انواع و تعداد مختلفی از داده ها را در خود ذخیره کند
از جمله عدد صحیح یا اعشاری کاراکتر و کلمه و جمله یا
مقادیر منطقی صحیح یا غلط و... که در ادامه مورد بحث قرار خواهد گرفت

اما قواعدی برای نامگذاری بر متغیر ها وجود دارد

1. نام متغیر ها میتوانند شامل حروف (A-Z) اعداد و _ باشد
2. به هیچ وجه نام متغیر نباید با عدد آغاز شود

`_Age_1 = 23` 
`2Age = 44` 

3. متغیر ها نسبت به حروف کوچک و بزرگ حساس هستند
برای مثال `Age` و `age` دو متغیر متفاوت به حساب خواهند آمد که مقادیر خاص خود را خواهند گرفت
4. متغیر نباید هم نام با کلمات کلیدی پایتون باشد

False	await	else	import	pass
None	break	except	in	raise
True	class	finally	is	return
and	continue	for	lambda	try
as	def	from	nonlocal	while
assert	del	global	not	with
async	elif	if	or	yield

انواع داده ها

در پایتون انواع مختلفی از داده ها و متغیرها وجود دارد که هر کدام خصوصیات خود را دارد برای اطلاع از نوع داده ها یا نوع متغیرها از تابع `type()` استفاده میکنیم و داده یا متغیر را بین پرانتز قرار میدهیم

Variable types:

1. Int
2. Float
3. Str
4. Bool
5. Dict
6. List
7. Tuple
8. Set
9. ...

Data type	Example	Example	Example	Example
Bool	True	False		
Str	'Hello'	"Hello_12"	'-12.01'	"True"
Int	23	-10	0	5000000
Float	23.0	-12.25	0.0	5e6
.

پس وقتی به یک متغیر داده از نوع `int` داده شود نوع آن متغیر نیز همان `int` در نظر گرفته میشود دیدن چند مثال از تابع `type()` میتواند مفید باشد

2.2 Operators, operands and data casting

عملگر چیست؟
عملوند چیست؟
عملگر ها نماد هایی هستند که بر روی عملوند ها عملیاتی خاص انجام میدهند
داده ها و متغیر هایی که بر روی آنها عملیات صورت میگیرد

پایتون operator ها را در دسته بندی های زیر قرار میدهد:

Arithmetic operators	عملگر های ریاضیاتی
Assignment operators	عملگر های مقدار دهی
Comparison operators	عملگر های قیاسی
Logical operators	عملگر های منطقی
Identity operators	عملگر های هویت
Membership operators	عملگر های عضویت
Bitwise operators	عملگر های باینری (تدریس نمیشود)

عملگر های ریاضیاتی (Arithmetic operators)

Operator	Name	نام	مثال	نتیجه
+	Addition	جمع	5+2	7
-	Subtraction	تفریق	58-12	46
*	Multiplication	ضرب	8*5	40
/	Division	تقسیم	12/4	3
%	Modulus	پیمانه(باقی مانده)	12%8	4
**	Exponentiation	توان	3**2	9
//	Floor division	تقسیم صحیح	13//3	4

عملگر های مقدار دهی (Assignment operators)

برای مثال اگر فرض کنیم $x=5$

نتیجه	معادل	مثال	عملگر
5	$x = 5$	$x = 5$	$=$
8	$x = x + 3$	$x += 3$	$+=$
2	$x = x - 3$	$x -= 3$	$-=$
15	$x = x * 3$	$x *= 3$	$*=$
1.666	$x = x / 3$	$x /= 3$	$/=$
2	$x = x \% 3$	$x \% = 3$	$\% =$
1	$x = x // 3$	$x //= 3$	$// =$
125	$x = x ** 3$	$x ** = 3$	$** =$
عملگر های مقدار دهی باینری نیز وجود دارند که از بحث این کارگاه خارج اند			

عملگر های قیاسی (Comparison Operators)

برای مثال فرض کنیم $x=5$ و $y=3$

Operator	Name	نام	مثال	نتیجه
<code>==</code>	Equal	برابر	<code>x == y</code>	False
<code>!=</code>	Not equal	نا برابر	<code>x != y</code>	True
<code>></code>	Greater than	بزرگتر	<code>x > y</code>	True
<code><</code>	Less than	کوچکتر	<code>x < y</code>	False
<code>>=</code>	Greater than or equal to	بزرگتر یا مساوی	<code>x >= y</code>	True
<code><=</code>	Less than or equal to	کوچکتر یا مساوی	<code>x <= y</code>	False

عملگر های منطقی (Logical operators)

قبل از ورود به این بحث مروری بر جبر بول (بولی) میتوانم مفید باشد

جبر بولی مقوله ای از جبر است که در آن مقادیر متغیر، درست و غلط است (True,False)

سه عملگر مهم در این بحث وجود دارد:

1. نقیض

2. و

3. یا

در این جبر اولویت ها به ترتیب بدین شکل هستند:

1. پرانتز

2. نقیض

3. و

4. یا

در پایتون این عملگر ها به این شکل نمایش داده می شوند: (فرض کنیم $x=6$)

نتیجه	مثال	توضیح	عملگر
True	$\text{Not}(x < 5 \text{ and } x < 10)$	نتایج را معکوس میکند	not
False	$x < 5 \text{ and } x < 10$	فقط وقتی درست است که هر دو شرط درست باشد	and
True	$x < 7 \text{ or } x < 4$	وقتی نتیجه درست است که حداقل یکی از شروط درست باشند	or

توضیح بیشتر:



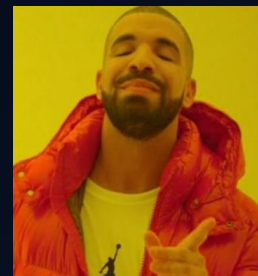
or



or



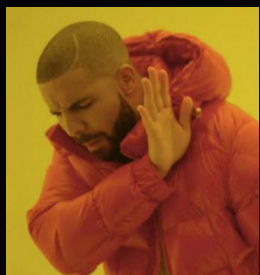
or



or



توضیح بیشتر:



and



and



and



and



توضیح بیشتر:

not



not



عملگر های هویت و عضویت بعد از تدریس دیکشنری ها بحث خواهد شد

تابع print()

تابعی است که هر تعداد پارامتر دلخواه از هر نوع داده (int,str,...) را (چه جدا جدا چه به صورت ترکیب) دریافت کرده و آن را در یک خط در خروجی برنامه چاپ میکند

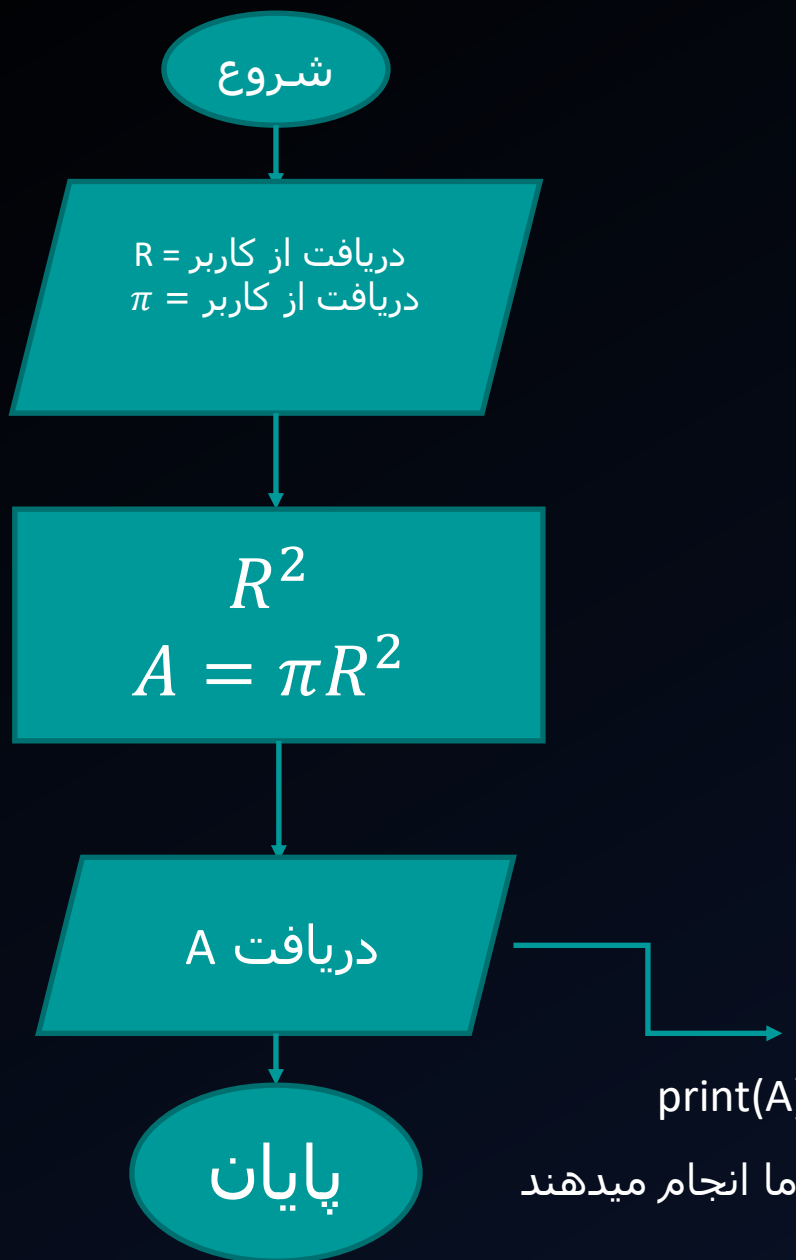
```
print(12,'Hello world',True,'True')
```

این تابع در انتها یک '\n' نیز از خود اضافه میکند و به خط جدید میرود (new line)
این تابع حتی میتواند ورودی نداشته باشد آنگاه به صورت یک '\n' رفتار میکند
در این تابع '\t' نیز به معنای یک Tab یا سه فاصله میباشد
توجه: '\t' و '\n' هر دو باید داخل ' ' قرار بگیرند.

به طور پیشفرض پارامتر های این تابع با فاصله (space) از هم جدا می شوند
اما با پارامتر step میتوان این پیشفرض را تغییر داد

```
print(12,'Hello world',True,'True',sep='**')
```

اگر تعداد زیادی متغیر را بخواهیم در یک تابع پرینت جای دهیم از f استفاده میکنیم به این صورت:
`print(f" some strings {variable} some other strings {another variable} string ... ")`



پس برای نشان دادن مقدار متغیرها به کاربر از تابع `print()` استفاده میکنیم
به این شکل که متغیر مورد نظر را در بین پرانتز مینویسیم
برای مثال برای دیدن مقدار `A` به این شکل عمل میکنیم :

`print(A)`

توابع `function` برنامه های از قبل نوشته شده ای هستند که کارهای مشخصی برای ما انجام میدهند
برای مثال تابع `print` وظیفه نمایش مقدار متغیرها را بر عهده دارد

تبدیل داده ها و تابع input()

تابع input() به ما کمک میکند تا برنامه را تعاملی تر کرده و از کاربر ورودی هایی بآب میل خودش بگیرد. این تابع یک پیغام (str) به عنوان پارامتر از برنامه نویسنده میگیرد و در اجرای برنامه آن را هنگام دریافت ورودی به کاربر نشان میدهد. '\n' و '\t' در این تابع نیز کاربرد دارند.

پس از دریافت مقدار از کاربر در متغیر مربوط به طور str ذخیره خواهد شد (حتی اگر از نوع int) باشد.

پس در هنگام استفاده از این تابع باید نوع آن به نوع مورد نیاز تغییر کند (data casting).

نتیجه	مثال	وظیفه	تابع
23	int("23.9")	تبدیل پارامتر به int	int()
14.0	float(14)	تبدیل پارامتر به float	float()
'120000.0'	str(12e4)	تبدیل پارامتر به str	str()

تبدیل از **str** به int یا float وقتی امکان پذیر است که پارامتر حتما عدد باشد. برای مثال '12' یک str است در صورتی که 12 یک int میباشد.

تمرین

1. یک برنامه بنویسید که از کاربر نام و نام خانوادگی اش را گرفته و در غالب یک پیام خوش آمد گویی چاپ کند (هم به طور معمول و هم با f امتحان کنید)
2. برنامه ای بنویسید که شعاع یک کره را (به متر) با یک متن مناسب و مفهوم از کاربر گرفته و حجم آن کره را با واحد cm^3 چاپ کند ($V = \frac{4}{3}\pi R^3$)
3. هواپیمایی با سرعت $450km/h$ از شهر A به شهر B که فاصله بین آنها $1200km$ است حرکت میکند برنامه ای بنویسید که زمان طی شده در این سفر را بر حسب دقیقه محاسبه کرده و نتیجه را با پیام مناسب چاپ کند
4. فرض کنیم یک باکتری در هر تولید مثل بتواند به 2 باکتری تبدیل شود. این باکتری در هر دقیقه 2 بار تکثیر می شود. با فرض نامیرا بودن و امکان تولید مثل بینهایت برای هر باکتری برنامه ای بنویسید که تعداد این باکتری ها را پس از n دقیقه (که n توسط کاربر انتخاب می شود) محاسبه کرده و با پیام مناسب چاپ کند
5. دنباله $R_n = n(n + 1)$ را در نظر بگیرید
برنامه ای بنویسید که جمله n ام دنباله زیر را با گرفتن n از کاربر محاسبه کرده و نتیجه را با پیام مناسب چاپ کند

تمرین

1. یک برنامه بنویسید که سه عدد از کاربر گرفته و میانگین آنها را محاسبه کند و با پیام مناسب چاپ کند