Sign in Get started

You have **2** free member-only stories left this month.
Sign up for Medium and get an extra one

Prashant Lakhera Follow
May 6, 2019 · 3 min read · Member-only · Listen

100 Days of DevOps — Day 93-Python Functions

To view the updated DevOps course(101DaysofDevOps)

Course Lists Register: https://www.101daysofdevops.com/register/

Course Link: https://www.101daysofdevops.com/courses/101-days-of-devops/

YouTube lisk: https://www.youtube.com/user/laprashant/videos

Welcome to Day 93 of 100 Days of DevOps, Focus for today is Python Functions

We can think function as a reusable piece/chunk of code

Built-in functions

The modules that come with Python are called the ***standard library****.*

for eg: ***print()/len()*** *is a built-in function*

But we can install third-party modules using ***Python Package Index(pip)***

```
# pip3 install pandas
Collecting pandas
100%     |################################| 11.8MB 142kB/s
Requirement already satisfied: python-dateutil>=2 in
/usr/local/lib/python3.6/site-packages (from pandas)
Requirement already satisfied: pytz>=2011k in
/usr/local/lib/python3.6/site-packages (from pandas)
Requirement already satisfied: numpy>=1.7.0 in
/usr/local/lib/python3.6/site-packages (from pandas)
Requirement already satisfied: tzlocal; in
/usr/local/lib/python3.6/site-packages (from python-dateutil>=2-
>pandas)
Installing collected packages: pandas
Successfully installed pandas-0.19.2
```

Functions

We can write our own functions, functions are like mini-program inside the program and their main aim is to get rid of duplicate code.

```
def hello():
    print("Hello ")
hello()
```

- *the def statement defines a function*
- hello is the name of the function

Output

```
Hello
```

Passing Argument to function

```
def hello(name):
    print("Hello " + name)
hello("Prashant")
```

Output

```
Hello Prashant
```

- *Arguments: The value passed in the function call(eg:Prashant)*
- *Parameter: The variable inside the function(name)*

```
def sum(num1,num2):
    return num1+ num2
output = sum(5,6)
print(output)
```

Every function has a return value. If our function doesn't have a return statement, the default return value is None.

Output

```
11
```

Global and Local Scope

In the case of Python variables inside the function can have the same name as a variable outside the function but they are considered as two separate variables.

The scope can be thought of as an area of the source code and as a container of variables

- **Global Scope:** *is code outside of all functions. Variables assigned here are global variables. Global scope is created when a program starts and is destroyed when it terminated. The code in global scope can't access a local variable.*
- **Local Scope:** *Function code is in its own local scope. Variables assigned here are local variables. Local scope is created when the function call and all the variables assigned during this function call exist and when the function returns all these variables are forgotten. The Code in local scope can access a global variable.*
- *The code in one function's local scope cannot use variables in another local scope*

```
x = 1 #local variable
def test():
x = 2 #global variable
```

One example

```
def test():
    x = 2

test()
print(x)
```

Output

As mentioned above x is local variable so as soon as the function returns all the variables are forgotten

```
/usr/local/Cellar/python3/3.6.8/Frameworks/Python.framework/Versions/3
.6/bin/python3.6 /Users/ijaleera/Documents/python/firstprogram.py
Traceback (most recent call last):
  File "/Users/ijaleera/Documents/python/firstprogram.py", line 5, in
<module>
    print(x)
NameError: name 'x' is not defined
```

If we want this code to work, then we need a keyword global

```
def test():
    global x
    x = 2

test()
print(x)
```

The Cause of scope is to limit the scope so the impact of a bug in the code will be limited.

Name	Scope
Must be only starting letter a-z,letter	Can use any part outside of function
Only one word allowed in the desired Code a function	Many word allowed, can be separate under a function
Can be Upper case letter, lower case letter, number, underscore	Can be Upper case letter, lower case letter, number, underscore
Value associated with it lost in the value	Value underwritten is accurate

Let's take one more example to reinforce the concept, here we are trying to access x before defining it

```
def myfun():
    x = 5

x = 4
myfun()
print(x)
```

Output

```
/usr/local/Cellar/python3/3.6.8/Frameworks/Python.framework/Versions/3
.6/bin/python3.6 /Users/ijaleera/Documents/python_testing/test.py
Traceback (most recent call last):
  File "/Users/ijaleera/Documents/python_testing/test.py", line 5, in
<module>
    myfun()
  File "/Users/ijaleera/Documents/python_testing/test.py", line 2, in
myfun
    x = 5
UnboundLocalError: local variable 'x' referenced before assignment
Process finished with exit code 1
```

But if this will work if?

```
def myfun():
    x = 5

x = 4
myfun()
print(x)
```

Looking forward from you guys to join this journey and spend a minimum an hour every day for the next 100 days on DevOps work and post your progress using any of the below medium.

- *Twitter: @100daysofdevops OR @ijaleera2015*
- *Facebook: https://www.facebook.com/groups/2323829430810645/*
- *Medium: https://medium.com/@devopslearning*
- *Slack: https://devops-myworld.slack.com/messages/C2F41LEFG4S/*
- *GitHub Link:https://github.com/100daysofdevops*

Reference

100 Days of DevOps — Day 0
D-day is just one day away and finally, this is a continuation of the post(I posted a month earlier)
medium.com

100 Days of DevOps — Day 92-Choosing Right EC2 Instance Type
Welcome to Day 92 of 100 Days of DevOps, Focus for today is Choosing Right EC2 Instance Type
medium.com

4

Get an email whenever Prashant Lakhera publishes.
Your email Subscribe
By signing up, you will create a Medium account if you don't already have one. Review our Privacy Policy for more information about our privacy practices.

More from Prashant Lakhera Follow
AWS Community Builder, Ex-Redhat, Author, Blogger, YouTuber, RHCA, RHCDS, RHCE, Docker Certified,KAMAS, CCNA, MCP, Certified Jenkins, Terraform Certified, NSCP

May 13, 2019

100 Days of DevOps — Day 92-Choosing Right EC2 Instance Type

To view the updated DevOps course(101DaysofDevOps) Course Registration link: https://www.101daysofdevops.com/register/ Course Link:...

AWS · 4 min read

Share your ideas with millions of readers. Write on Medium

May 12, 2019

100 Days of DevOps — Day 91-How to check if the file exists (Bash/Python)

To view the updated DevOps course(101DaysofDevOps) Course Registration link: https://www.101daysofdevops.com/register/ Course Link:...

Dev Ops · 4 min read

May 11, 2019

100 Days of DevOps — Day 90- Try and Except Statement Python

To view the updated DevOps course(101DaysofDevOps) Course Registration link: https://www.101daysofdevops.com/register/ Course Link: https://www.101daysofdevops.com/courses/101-days-of-devops/ YouTube link:...

Python · 2 min read

May 9, 2019

100 Days of DevOps — Day 89-Python Files I/O

To view the updated DevOps course(101DaysofDevOps) Course Registration link: https://www.101daysofdevops.com/register/ Course Link: https://www.101daysofdevops.com/courses/101-days-of-devops/ YouTube link:...

Programming · 2 min read

May 8, 2019

100 Days of DevOps — Day 88-Lists in Python

To view the updated DevOps course(101DaysofDevOps) Course Registration link: https://www.101daysofdevops.com/register/ Course Link: https://www.101daysofdevops.com/courses/101-days-of-devops/...

Programming · 5 min read

Love podcasts or audiobooks? Learn on the go with our new app. Try Knowable

Medium
About Help Terms Privacy