

U1BP42MRS : That's a bit rude, but I challenge you to show me how this case above is not the same, vs just saying it's not. Here is proof it is:

```
>>> def foo():
...     print("foo")
...
>>> bar = lambda: print("bar")
>>> foo()
foo
>>> bar()
bar
>>>
```

You can pass around `bar` or `foo` to any function that needs a callable (with no args in this case)

```
U1BP42MRS : ``>>> type(foo)
<class 'function'>
>>> type(bar)
<class 'function'>
>>>
```

U5NMSURQA : something like this, in a 2x2 grid?

U5NMSURQA : rude? man, I specifically joked about a cup-of-coffee so that doesn't seem rude :disappointed:

U1BP42MRS : > nooo waaay

U4BMZ90T0 : I legitimately just have two frames of a 2 labels, a dropdown and a button

U1BP42MRS : Is what I was referring to

U4BMZ90T0 : and I can't get them to do anything lol

U4BMZ90T0 : Format wise

U5NMSURQA : > You can pass around `bar` or `foo` to any function that needs a callable (with no args in this case)

U1BP42MRS : this is actually what I should have showed

```
>>> def foo():
...     print("foo")
...
>>> bar = lambda: foo()
>>>
>>> foo()
foo
>>> bar()
foo
>>>
```

U5NMSURQA : Yes, you're right 100%

U5NMSURQA : but

U1BP42MRS : effectively foo and bar are the same here

U5NMSURQA : tkinter callbacks don't accept arguments, they are all callables compatible with `lambda: x`

U5NMSURQA : but you almost always need to pass something that takes a whole lot of arguments

U1BP42MRS : This talk wasn't directly about tkinter

U5NMSURQA : therefore it's a common pattern to hide the stuff that you actually need to do behind a simple unassuming lambda

U1BP42MRS : that was the general context, but not to my comment about `lambda: foo()` and `foo` being the same

U4BMZ90T0 : alright I got it. Now I need a progress bar.

U5NMSURQA : alright then :slightly_smiling_face:

U5NMSURQA : we're both wrong and right in 50% of contexts :slightly_smiling_face:

U0LSCQQNR : solve one problem, another pops up

U5NMSURQA : tkinter.ttk has a progressbar widget

U1BP42MRS : :stuck_out_tongue:

U1BP42MRS : I was being too literal, my apologies (sorry, had to run into a quick meeting)

U371MGS2H : ^^ Could anyone explain what I'm doing wrong to get the error there?

U371MGS2H : i have a feeling it's just a weird typo that i'm not seeing

U486CN4VD : I'm noticing an immediate syntax error
U0LSCQQNR : can you even use comments in a dict?
U486CN4VD : `replication`, can't happen
U371MGS2H : you mean `replication`, `?` or am i missing where there's a space?
U486CN4VD : shouldn't it be `replication`:
U371MGS2H : ooooooh yes
U0LSCQQNR : yeah, in your second dict in the list
U371MGS2H : thank you for the eyeballs :smile:
U0LSCQQNR : should be `replication`: { ...}
U486CN4VD : ^^
U371MGS2H : facepalm :stuck_out_tongue:
U0LSCQQNR : you have it in two locations
U0LSCQQNR : both `replication` keys
U371MGS2H : fixed!
U486CN4VD : :boom:
U371MGS2H : duhhhhhhhhh
U0LSCQQNR : sometimes its the little things
U0LSCQQNR : and things like this is where using an editor or IDE helps
U0LSCQQNR : because it can check your code for syntax validity as you write it
U0LSCQQNR : not so much with the shell or ipython
U371MGS2H : <@U0LSCQQNR> actually i use vim with syntax highlighting and it runs flake8 on everything when saving, and somehow it just decided i must be trying to do some other valid thing in python
U371MGS2H : at least my test suite figured out it was a problem :stuck_out_tongue:
U0LSCQQNR : oh, interesting
U0LSCQQNR : :slightly_smiling_face:
U371MGS2H : test suites FTW
U0LSCQQNR : so flake8 said that was valid python?
U0LSCQQNR : interesting
U371MGS2H : yeah idunno what it thought i was looking for
U0LSCQQNR : you may want to write a bug report about that on their gitlab
U0LSCQQNR : include that dict structure as an example
U371MGS2H : that's not a bad idea
U371MGS2H : worst case there really is some half valid reason for syntax like that and they close it
U0LSCQQNR : yup
U0LSCQQNR : don't see anything on the tracker about dictionaries that applies
U0LSCQQNR : so, maybe an edge case?
U371MGS2H : <<https://gitlab.com/pycqa/flake8/issues/353>>
U4BMZ90T0 : Finally finished this stupid gui. It should work
U4BMZ90T0 : omg it works
U47HK8BS8 : :awweya:
U53MQAY3A : Hey everyone. I need some help with django, particualrly behave-django. I have some tests, some of which are dependent on there being some entries in a specific table. I thought Django would run migrations when it sets up the test database and created one but to no avail. I'd like to know if there's a way to add those table entries after the test database has been created. I'm using DRF.
U1BP42MRS : You may want to ask in <#C0LMFRMB5|django> <@U53MQAY3A> instead of here
U53MQAY3A : Cool, thanks.
U2BS4M1RV : Anyone update PyCharm in the last couple of days and have your colors change?
U2BS4M1RV : I don't think function names were colored before.
U1BP42MRS : Mine have been for as long as I can recall - but maybe it accidentally overrode your config
U2BS4M1RV : It looks like many of the colors changed a bit. I don't remember what color functions were, maybe they were highlighted, but it is different. The TODO: colors also look a little different.
U44Q0N7JT : has anyone used the attribute "placeholder" when using pyramid and deform?
U44Q0N7JT : I need to add a placeholder to a field but Im not actually coding an html page. so this would need to be done while constructing the schema of the table fields
U29163YQH : <@U2BS4M1RV> yeah my colors got changed too
U2BS4M1RV : <@U29163YQH> Are they easy to change back? I think I liked the old ones better.
U29163YQH : Yeah color are easy to edit. But you need to figure out what they where before. I launched the previous version and compared some
U1BP42MRS : Oh I guess I am on the EAP, I guess mine have been _different_ for a while :facepalm:
U29163YQH : Mine were changed with 2017.2

U2BS4M1RV : I tried the IntelliJ theme, I can't stand it, it's too bright and the colors don't stand out much.
U29163YQH : I use darcula with small modification
U2BS4M1RV : Yeah, that's what I use. Only issue I have with the prior to this week colors is that I want multi-line TODOs.
U2BS4M1RV : That's more of how it parses those than the color though.
U1BP42MRS : multiline todos would be nice - have you had that working before?
U2BS4M1RV : I haven't.
U2BS4M1RV : Just something I would like to see.
U1BP42MRS : Aww, I thought you were about to make my week!
U2BS4M1RV : It would be something worth requesting, I would think.
U2BS4M1RV : Maybe I'll offer to pay for a year if they add it.
U4WG0KTNJ : Appreciate any help to point me to the right direction, not familiar with os class, how the files look like is in the comment :pray:
U5VGKQ2SY : `if f.filename in os.listdir(dir_name)` Maybe?

U5VGKQ2SY : yeah.. this works for me if I have zip file that extracts to csv if csv not there. But print "csv-filename already exists" if csv w/ same name already exists.
U4WG0KTNJ : Thanks! This is what I have so far, not working as I thought it would be, instead of file_a_1, file_a_2, file_b_1, file_b_2. It gives file_a_1, file_a_2, file_b_3, file_b_4
U5VGKQ2SY : update line 1 to:``
import os
import os.path
from zipfile import ZipFile
``

U5VGKQ2SY : this `zip_ref = zipfile.ZipFile(file_name)` to `zip_ref = ZipFile(file_name)`
U5PJK7JHE : quick question assume I have a list `["abc","ab","cda" ..]` I want to remove all elements that are substring i.e. in this case it would be `["abc","cda"]`
U5PJK7JHE : what is the quickest way to do that?
U4WG0KTNJ : len?
U5PJK7JHE : huh?
U1BP42MRS : <@U5PJK7JHE> what have you tried?
U1BP42MRS : We like to see what you've tried before just giving answers :)
U5PJK7JHE : ``for e in li:any([x in e for x in e if x !=e])
``

U5VGKQ2SY : <@U4WG0KTNJ> moving our convo as comments/thread into your last code snippet