

U1BP42MRS : <@U6G0F77F1> did you follow the example from the docs  
(<[https://docs.djangoproject.com/en/1.11/topics/db/examples/one\\_to\\_one/#>](https://docs.djangoproject.com/en/1.11/topics/db/examples/one_to_one/#>))

U1BP42MRS : Basically:

...

```
>>> r = Restaurant(place=p1, serves_hot_dogs=True, serves_pizza=False)
```

```
>>> r.save()
```

...

Where `p1` is the instance of the "model which is already saved"

U6G0F77F1 : <@U1BP42MRS> i am going to try this and see if that works .. thanks again .I'll update here once i am done

U6B5YDMHT : I see in a lot of stackoverflow questions and blog posts that to change the default admin UI, you have to change things in the directory ./templates/admin. in 1.11 that doesn't seem to exist...

What's the updated way to change the admin template? In particular, to either change the /change/ url or to disable editing after initial creation.

U6B5YDMHT : I just don't see a `templates` directory at all...

U0LSCQQNR : <<https://docs.djangoproject.com/en/1.11/ref/contrib/admin/#overriding-admin-templates>>

U5NMSURQA : > The required arguments are an RFC 822 from-address string, a list of RFC 822 to-address strings (a bare string will be treated as a list with 1 address), and a message string.

U5NMSURQA : sendmail's "signature" is something like: `def sendmail(from\_addr: str, to\_addrs: list, msg: str):`

U5NMSURQA : ``#Change recipients string to list. to = [to]``

U5NMSURQA : you should transform your string of emails ('<mailto:email1@mail.com|email1@mail.com>',

<<http://email2.mail.com>|email2.mail.com>'), so that it becomes a list with each element being a proper email address

U4EALKV2M : So instead of reassigning to a list, perform a split? Like this:``

```
#Change recipients string to list.
```

```
to = to.split(',')
```

...

U5NMSURQA : yes, that's it :slightly\_smiling\_face:

U4EALKV2M : Which would still allow for cc and bcc to be added

U4EALKV2M : Cool! I'll try that out.

U5NMSURQA : I'd also `strip` them of whitespace, otherwise that leading space will become a part of one email address, and I'm not sure if `sendmail` handles that

U5NMSURQA : better safe than sorry

U4EALKV2M : Ahh good point

U4TJSS0V7 : hi guys. anyone here know any solutions to having a python script create a list of objects parallel to another script consuming those objects as they are created?

U4TJSS0V7 : i've got some test automation where each test requires a brand new account, but the accounts take anywhere between 30s-2min to create. i'd like to create the accounts parallel to the tests running to speed up execution time. i've found a way using `multiprocessing.Process` and continuously adding accounts to a `multiprocessing.Queue`, but this only works if that process is started from within the same script that executes the tests. i have a use case that requires the test execution to be able to consume a pre-made account as soon as the test starts executing, so `multiprocessing.Process` does not fully solve my problem, since the first test to execute still has to wait for the first account to be created and added to the queue

U4TJSS0V7 : what would be ideal is a way to create a queue of accounts from a `create\_accounts.py` script that i can execute at any time, and then be able to consume that queue at any arbitrary point in time \_during\_ or \_after\_ `create\_accounts.py`

U0LSCQQNR : an account takes up to two minutes to create?

U0LSCQQNR : :open\_mouth:

U0LSCQQNR : no chance you can spend time trying to optimize that process?

U4EALKV2M : <@U5NMSURQA> That worked perfectly, thank you for the help!``

```
#Change recipients string to list and strip whitespace.
```

```
to = [x.strip() for x in to.split(',')]
```

...

U5NMSURQA : :slightly\_smiling\_face: I'm glad

U1BP42MRS : <@U4TJSS0V7> I wrote a service at a job with similar issues, it basically kept a fresh list of users at all times and tests could "check one out", later the service would clean them up

U4BMZ90T0 : Anyone here who has used `ttk.progressbar`

U4BMZ90T0 : If I never have to use tkinter in my life again then it's a good day

U1BP42MRS : This is why electron is so popular :(

U4BMZ90T0 : Why did no one tell me of these other beautiful things

U4BMZ90T0 : Attempting to run this damn progress bar is gonna be the end of me

U4BMZ90T0 : honestly I'm just gonna say screw it and just click a button and you have to wait

U1BP42MRS : What's the app do?