

U6FNX78AJ : I'm having a hard time distilling this into a code example. Maybe it would help me to just talk about it abstractly. If you don't mind? For one, what is a socket, and what would cause it to close/how could I keep it open? I have a feeling that this is the problem

U6FNX78AJ : Thank you for your help

U6FNX78AJ : reading about sockets here <https://www.tutorialspoint.com/python/python_networking.htm> right now. if you know of any good resources off the top of your head I would love to have them

U1BP42MRS : Sorry it's late my time, so I may be out shortly. I'm hoping someone else familiar with the architecture comes in to help too

U6FNX78AJ : I understand, thanks anyway

U6G0F77F1 : hello people ..

U1BP42MRS : <@U6G0F77F1> if you don't see answers here, in the future use the channel dedicated to your topic (<#C0LMFRMB5|django> or <#C0W0D17MJ|django_learning>) No need to move, but it's generally best to scope questions

U6G0F77F1 : thanks <@U1BP42MRS> . I surely will ask them in these dedicated channes.

U1BP42MRS : No worries :slightly_smiling_face:

U1BP42MRS : <@U6G0F77F1> did you follow the example from the docs (<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?