COMP 8347: Internet Applications and Distributes Systems
FALL 2020 LAB #3

PART 1: Install necessary software, Python, Django, and PyCharm

1. Follow the setup instructions given in the installation Guides for Python, Django, and PyCharm for Windows and MAC OS X. The guides are posted on BB under **Additional Resources**, if you do not have the necessary software installed.

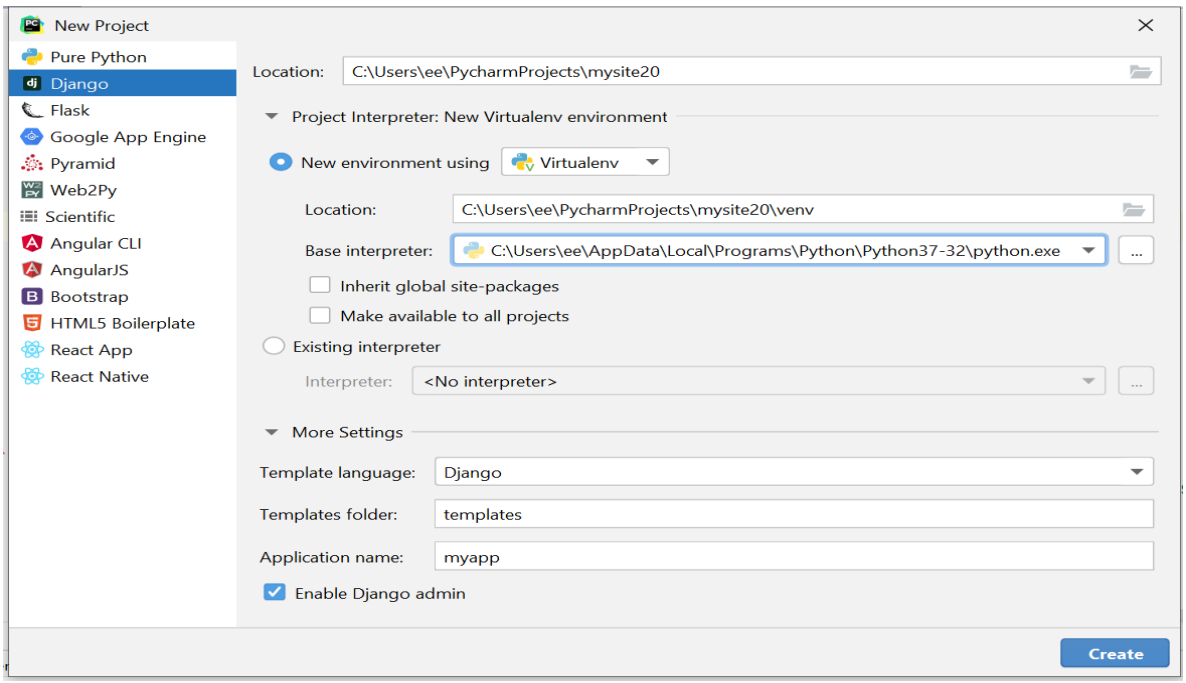
2. Use this command to check Django version:

```
$ python -c "import django; print(django.get_version())"
```

PART 2: Run Django from PyCharm

1. Start a new project in PyCharm as shown in the dialogue box below.

   a) Open PyCharm and click on **File → New Project**.
   b) In left pane select "**Django**"
   c) In dialog box enter path and project name (***mysiteF20***)
   d) Enter the **path to Python** as the Base interpreter.
   e) Click "**More Settings**" and enter Application name (***myapp***)
   f) Click ***Create***.



2. Inside the project, click **Tools → Run manage.py Task …**

a) Create the project database (db).

- Type the command **`migrate`** in **manage.py** console.
- Verify that the database is created, open migration folder

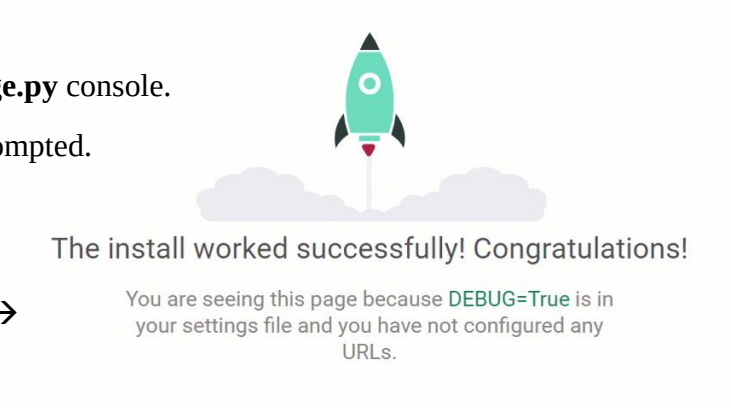
b) Create admin user.

- Type **`createsuperuser`** in **manage.py** console.
- Enter username, email, password as prompted.

c) Start server.

- Click Run → **Run 'mysiteF20'**.

d) Go to 127.0.0.1:8000/ you should see this →



PART 3: Edit *models.py* to create the necessary models

In this course you will be developing an **E-learning WebApp** in Django. The first step is to build the models/tables in the webapp database. Follow the instructions given below to create the first two models. In the next lab you will add more models to your database and modify the existing models.

1. Add the following import statements:

```
from django.db import models
import datetime
from django.contrib.auth.models import User
from django.utils import timezone
```

2. Add the following models.

a. ***Topic*** with the following fields:

```
class Topic(models.Model):
    name = models.CharField(max_length=200)
```

b. ***Course*** with the following fields:

```
class Course(models.Model):
    title = models.CharField(max_length=200)
    topic = models.ForeignKey(Topic, related_name='courses',
on_delete=models.CASCADE)
    price = models.DecimalField(max_digits=10, decimal_places=2)
    for_everyone = models.BooleanField(default=True)
```

3. Create db tables (make sure *myapp* is included under INSTALLED_APPS in *settings.py*). See what happens after each step.

a. **Tools → Run manage.py Task…** (opens a window where you can type ***manage.py*** commands)
b. In ***manage.py*** window: Type **makemigrations myapp** in dialog box.
c. In ***manage.py*** window: Type **sqlmigrate myapp 0001** #Check latest file in *migrations* dir
d. In ***manage.py*** window: Type **migrate**

PART 4: Enter data through Admin interface

1. Update *admin.py* as follows:

```
from django.contrib import admin
from .models import Topic, Course

# Register your models here.
admin.site.register(Topic)
admin.site.register(Course)
```

2. Start your server (**Run → Run 'mysiteF20'**) and navigate to admin site (127.0.0.1:8000/admin).

3. Login using *superuser* name and password (from **PART 2**).

4. Enter the information for each *Topic* and *Course*, as given in ***lab3dataF20.txt*** through the admin interface. How is the data being displayed? Would it be more useful to display additional information?