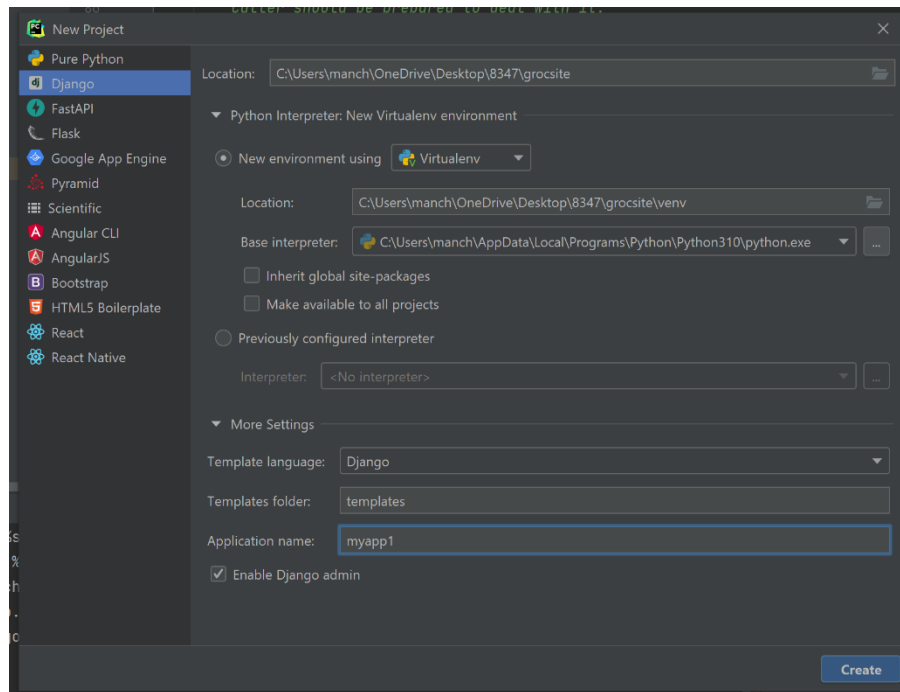# LAB 3 – Introduction to Django – Simple Models

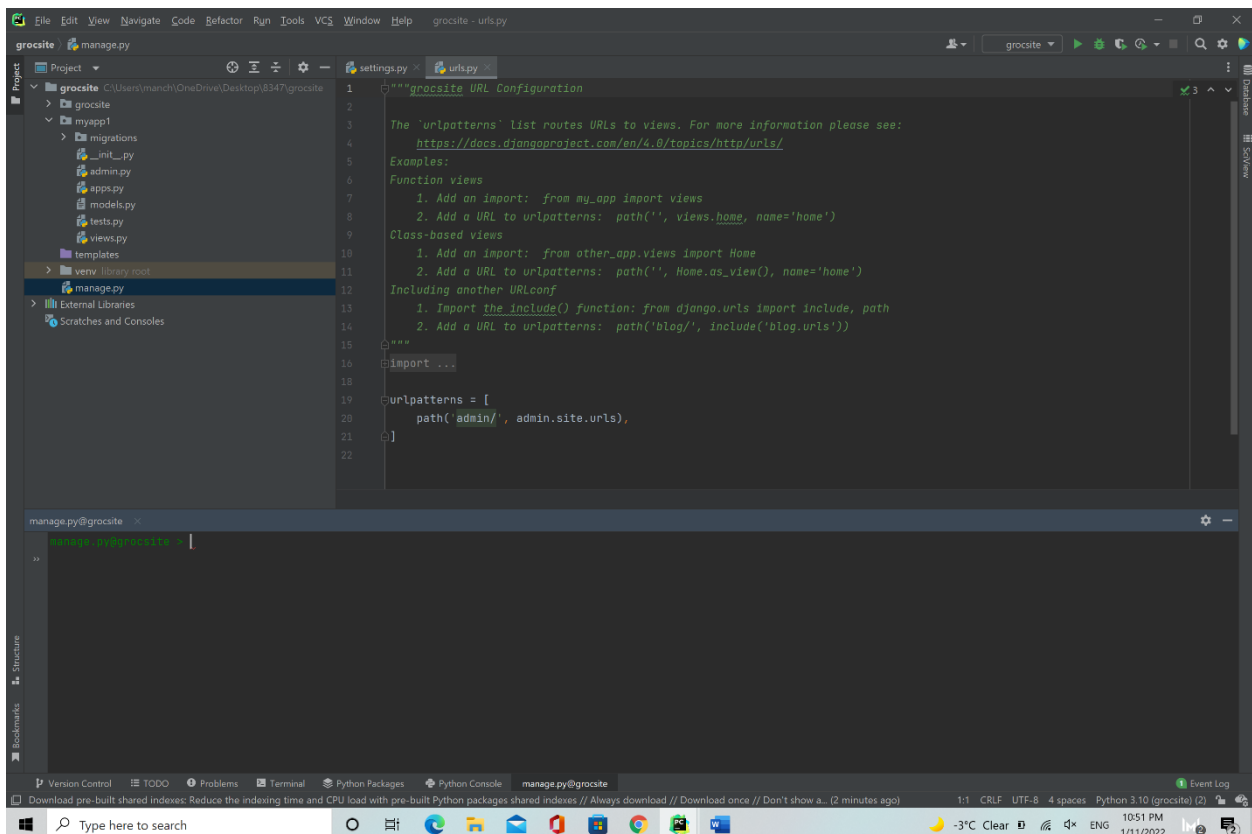**Install necessary software, Python, Django, and PyCharm**

Follow the setup instructions given in the installation guide for Python, Django, and PyCharm for Windows/MAC OS X. The guides are available on Brightspace.

PART 1: 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 (*carsite*)

    d) Enter the **path to Python** as the Base interpreter.

    e) Click "**More Settings**" and enter Application name (*carapp*)

    f) Click *Create*.

2. Once the project "carsite" is created, click on **Tools → Run manage.py**. After this, you will see a console as follows:

In the console shown at the bottom of the above figure, do the following:

a) Create the project database

- Type the command **migrate**
- Verify that the database is created by checking the **migration** folder

b) Create admin user

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

c) Start the server

- Click Run → **Run 'carsite'**.

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

The install worked successfully! Congratulations!

You are seeing this page because DEBUG=True is in your settings file and you have not configured any URLs.

PART 2: Edit *models.py* to create the necessary models for car buying website (Ex. ford.ca). We will build this car website step by step throughout the labs.

Write meaningful

comments using the # sign wherever necessary

from django.db import models

from django.contrib.auth.models import User

1. Add the following models

a. *CarType* with fields below:

```
class CarType(models.Model):
    name = models.CharField(max_length=150)
```

b. *Vehicle* with fields below:

```
class Vehicle(models.Model):
    car_type = models.ForeignKey(CarType, related_name='vehicles', on_delete=models.CASCADE)
    car_name = models.CharField(max_length=200)
    car_price = models.DecimalField(max_digits=10, decimal_places=6)

    inventory = models.PositiveIntegerField(default=10)
    instock = models.BooleanField(default=True)
```

c. *Buyer* with fields below:

```python
class Buyer(User):

AREA_CHOICES = [
        ('W', 'Windsor'),
        ('LS', 'LaSalle'),
        ('A', 'Amherstburg'),
        ('L', 'Lakeshore'),
        ('LE', 'Leamington'),]


    shipping_address = models.CharField(max_length=300, null=True, blank=True)
    area=models.CharField(max_length=2, choices=AREA_CHOICES, default='A')
    interested_in = models.ManyToManyField(CarType)
```

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

    a. **Tools ⟶ Run *manage.py***
    b. In the console, type **makemigrations carapp**. Then, type **sqlmigrate carapp 0001** #Check latest files in *migrations* dir. Finally, type **migrate**
    c. Check the migrations folder and notice the changes stored in "**0001_initial**" file

3. Upload your ***models.py*** file on Brightspace. Also, upload a Word document with related screenshots and mentioning all the primary and foreign keys in the "0001_initial" file. Moreover, specify which foreign key relates to which tables.

PART 3: Repeating the above process with the choice of your website.

- Go to the following link https://www.netguru.com/blog/django-apps-examples
- Choose any application you like from NASA, Venmo, Dropbox, Pinterest, or the Washington Post.
- Repeat all the steps described in PART 2 for your selected application such as:
    - Creating at least 3 classes in models.py
    - One of the above classes should be the user or the client.
    - Each class should have variety of fields (Ex. Boolean, decimal, choice). At least 4 fields in two classes
    - Use foreignkey, default, null, and blank for the fields.
- Upload your ***models.py*** file on Brightspace.