* I am at “Dynamic Data in Template”
* There is a lot of stuff that happens in the “Your first django project” section, not all of which I understand. Revisit that section (or for now, just follow it.)
* In the “Django Models” section, we create an application called blog with manage.py startapp blog. Startapp creates an app with a number of files in it.

Django Girls Blog files:

* **blog/admin.py** imports the Post model and makes it visible on the admin page (of the website.
  + from django.contrib import admin
  + from .models import Post
  + admin.site.register(Post)
* **manage.py** contains script that helps with management of the site.
* **settings.py** contains configuration of site.
  + Set timezone, language, static path
  + Set the allowed hosts.
  + The database is set up here.
  + This is all found in “Your first django project” section.
  + Contains all the installed apps in INSTALLED\_APPS. When we create a new blog app in “Django Models” section, we need to add it to the INSTALLED\_APPS.
* **urls.py** contains a list of patterns to be resolved by urlresolver
* **models.py** defines the Post model.
  + Post is a class.
  + A model is an object of a class that is saved in a database.
  + from django.db import models. I probably need to look at that.
  + Defines variables
    - author = models.ForeignKey(settings.AUTH\_USER\_MODEL, on\_delete=models.CASCADE)
    - title = models.CharField(max\_length=200)
    - text = models.TextField()
    - created\_date = models.DateTimeField(default=timezone.now)
    - published\_date = models.DateTimeField(blank=True, null=True)
  + defines “publish” method, which takes self and which sets the published\_date to now and does self.save()
  + defines \_\_str\_\_ method, which takes self and returns self.title
  + when I do my FYS thing, I will create an FYS class. Each object will be saved in my database. It will contain the following fields:
    - title
    - course number
    - description
    - day and times
    - dates
    - tags
    - open/ closed
* **views.py** defines a post\_list method
  + takes request as an argument.
  + from django.shortcuts import render
  + return render(request, 'blog/post\_list.html', {})
  + I am not totally sure I understand how this works. Look at docs for render.
* **post\_list.html** is the html file that contains the list of posts.
  + I need to go back and look more carefully at how this works.