Django REST Framework Notes

General Terms:

* API is a middleman that allows two software that is differently written to communicate to each other. Think of it as like a waiter in a restaurant.
* REST is a way in which a server responds to Create, Read, Update, Delete request in a standard way. It is to treat a server URL as access point to its resources. For example, when you send a GET request to a URL example.com/users, you are requesting get the “state” of that point.
* Therefore, a RESTful API is simply to create a middleman between to application to respond to Create, Read, and Delete request.

Development Notes:

* We follow the same way on how to create a Django project except now we use the rest framework.
* Create project.
* Create app.
* Add app to installed project apps.
* According to CFE, its preferred to create an app called “api” so that all main URLs routes that are placed in there are understood to be part of the API just in case the Django project will be used as the regular website with MVT pattern.
* Create models.
* Create serializers.
* Create API views.
* We can use either function-based API view class-based generic API view which inherits through the generic API view classes which corresponds to the actions REST API can take such as (assuming the associated model is named “Products”):
  + ProductCreateAPIView – generics.CreateAPIView
  + ProductRetrieveAPIView – generics.RetriveAPIView
  + ProductListAPIView – generics.ListAPIView
  + ProductListCreateAPIView – generics.ListCreateAPIView
  + ProductUpdateAPIView – generics.UpdateAPIView
  + ProductDeleteAPIView – generics.DestroyAPIView
* CRUD is generally what you always need to do in the database. By creating the API, we can override the functionality to do additional actions in each view.
* Aside from the generic API views, we can use mixins directly, but the code will be cluttered so the preferred one is using the generic API views
* Verify if the endpoints are working via the built in DRF UI. Or you can use Python requests library to emulate how a client will talk to our API. Or use 3rd party API platform such as Postman.
* Add permissions.
* Add authentications.