1. django architecture
2. django installation
3. git(how to setup, clone, add, commit, push, pull)
4. sample project(content:)
5. design database(excel, what is the need of relations)
6. basic flow(request and response process with django dev server)
7. how to create project, app, database structure creation(makemigrations, migrate), runserver
8. django admin
9. models(abstract models, proxy models, models, inherited models)
10. CRUD operations of model
11. image, doc file upload.
12. relations in models(primarykey, foreignkey, manytomany, and how to achieve one2many.)
13. views(function based views)
14. static, html, css, bootstrap, django template language
15. forms(forms, modelsforms). form validations
16. Basic project completed.
17. problems in the project
18. views(class based views and generic views.)
19. authentication(login, session, logout)
20. django rest framework:

need of APIS

API view

serializers(validate request, convert request data)

authentication: Basic and token based

1. unitest cases
2. pagination
3. unittest cases
4. middleware( error handling, request tracking)
5. signaling concept
6. cache management
7. maintain multiple databases for your application
8. formset
9. deploying application using gunicorn as webserver, nginx as proxy server
10. working with some other apps which has API support
11. email templating
12. write own management commands
13. how to run management command from setup.py
14. New topics added for this batch.
15. containerizing the services in project using docker and docker-compose
16. automate the continuous deployment using ansible
17. provide highavailability and resiliency for your project using docker-swarm