Q1 What is the difference between framework, library and module?
Ans

library: collection of related functionality
framework: Inversion of Control

module: abstract interface with explicit exports and imports, implementation and interface are separate, there may be multiple implementations and the implementation is hidden

Q2 Explain IoC with an example.
Ans   In software engineering, inversion of control (IoC) is a programming principle. IoC inverts the flow control as compared to traditional control flow. In IoC, custom-written portions of a computer program receive the flow of control from a generic framework. A software architecture with this design inverts control as compared to traditional .

In the mail client example, the framework could follow both the keyboard and mouse inputs and call the command invoked by the user by either means, and at the same time monitor the network interface to find out if new messages arrive and refresh the screen when some network activity is detected. The same framework could be used as the skeleton for a spreadsheet program or a text editor.

Q3 What is MVT Architecture? How is it different from MVC?
Ans
The MVT (Model View Template) is a software design pattern. It is a collection of three important components Model View and Template. The Model helps to handle database. It is a data access layer which handles the data.
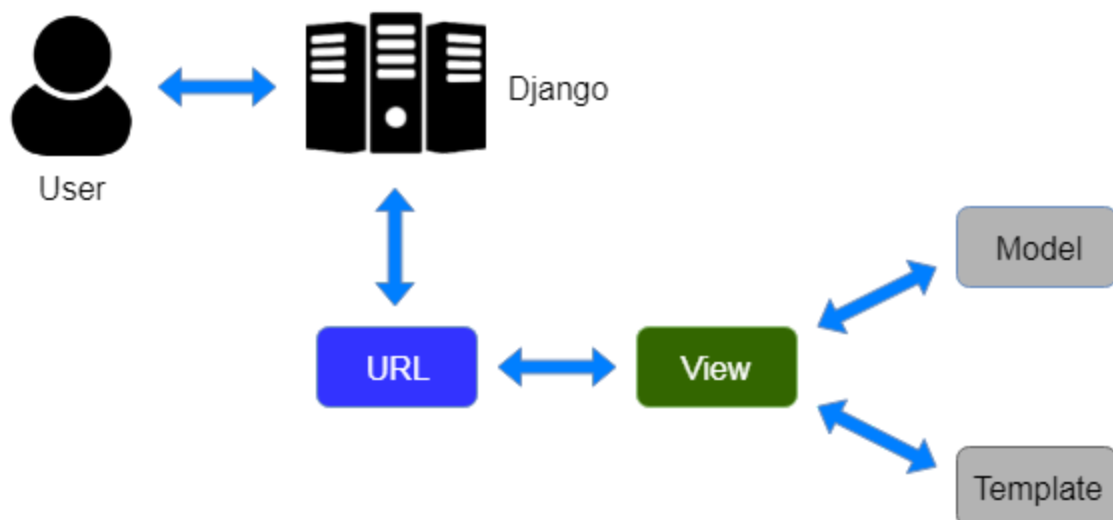
The main difference between the two patterns is that Django itself takes care of the Controller part (Software Code that controls the interactions between the Model and

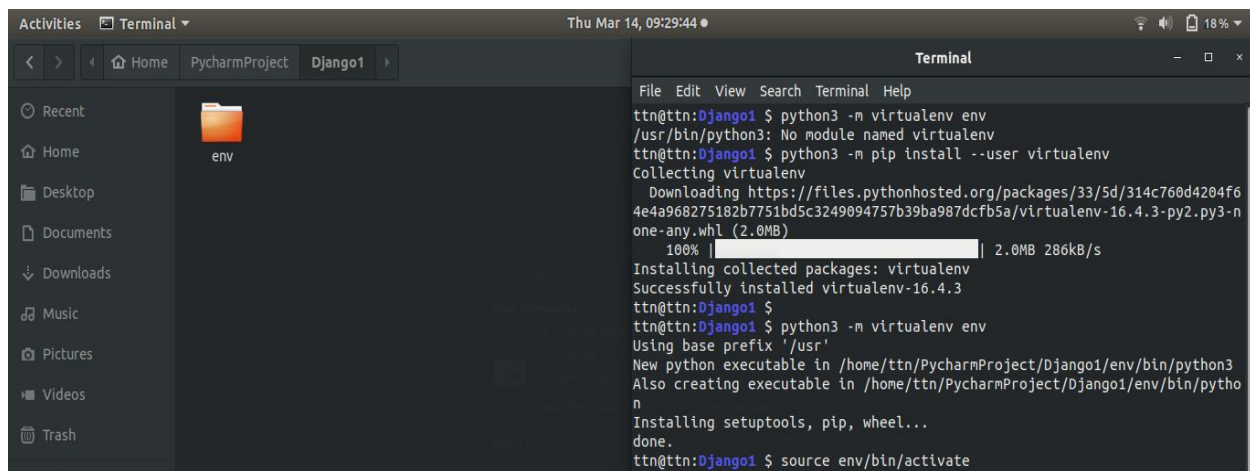View), leaving us with the template. The template is a HTML file mixed with Django Template Language (DTL).

Q4 What is virtual environment? Why is it needed?
Ans A virtual environment is a tool that helps to keep dependencies required by different projects separate by creating isolated python virtual environments for them. This is one of the most important tools that most of the Python developers use.
Imagine a scenario where you are working on two web based python projects and one of them uses a Django 1.9 and the other uses Django 1.10 and so on. In such situations virtual environment can be really useful to maintain dependencies of both the projects.

Q5 Create a virtual environment for your first Django application.
Ans

Q6 Install Django through package manager.
Ans

**Q7 Create a project 'greetings' through django admin and run it.**

Ans
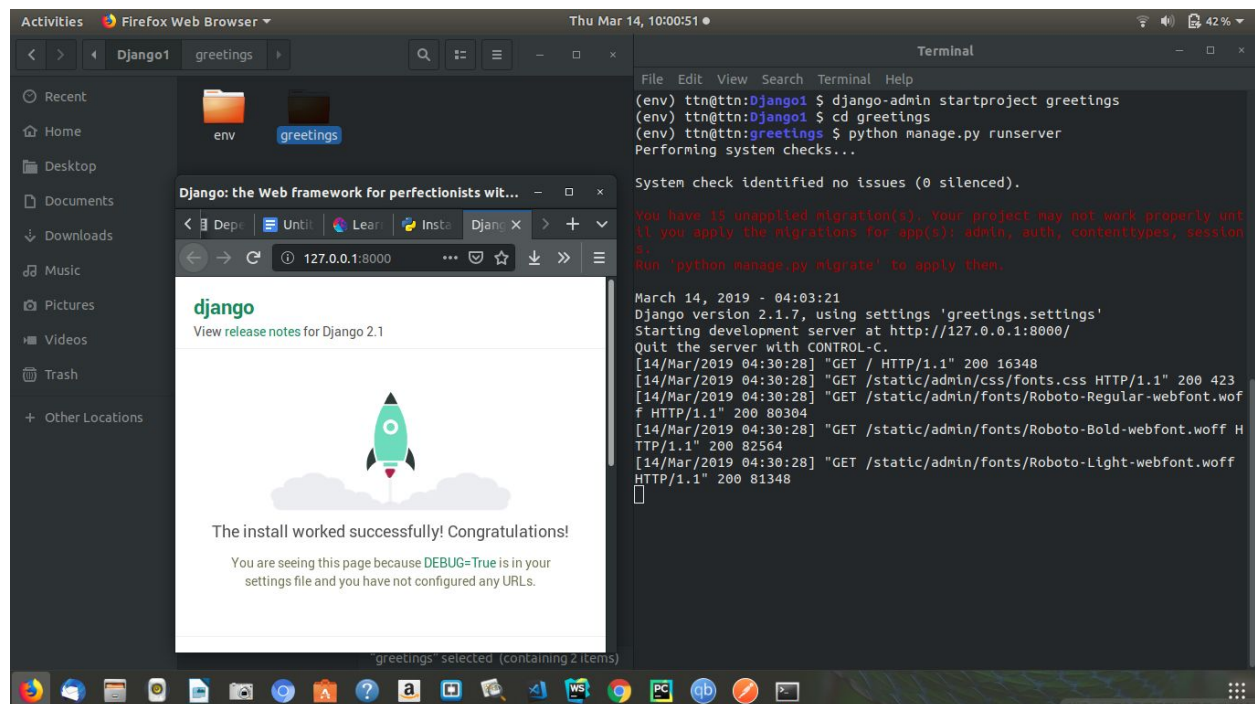
Q8 What is the difference between an app and a project? Create an app named greet and add it into your project greetings.

Ans A *project* refers to the entire application and all its parts.
An *app* refers to a submodule of the project. It's self-sufficient and not intertwined with the other apps in the project such that, in theory, you could pick it up and plop it down into another project without any modification. An *app* typically has its own *models.py* (which might actually be empty). You might think of it as a standalone python module. A simple project might only have one app.
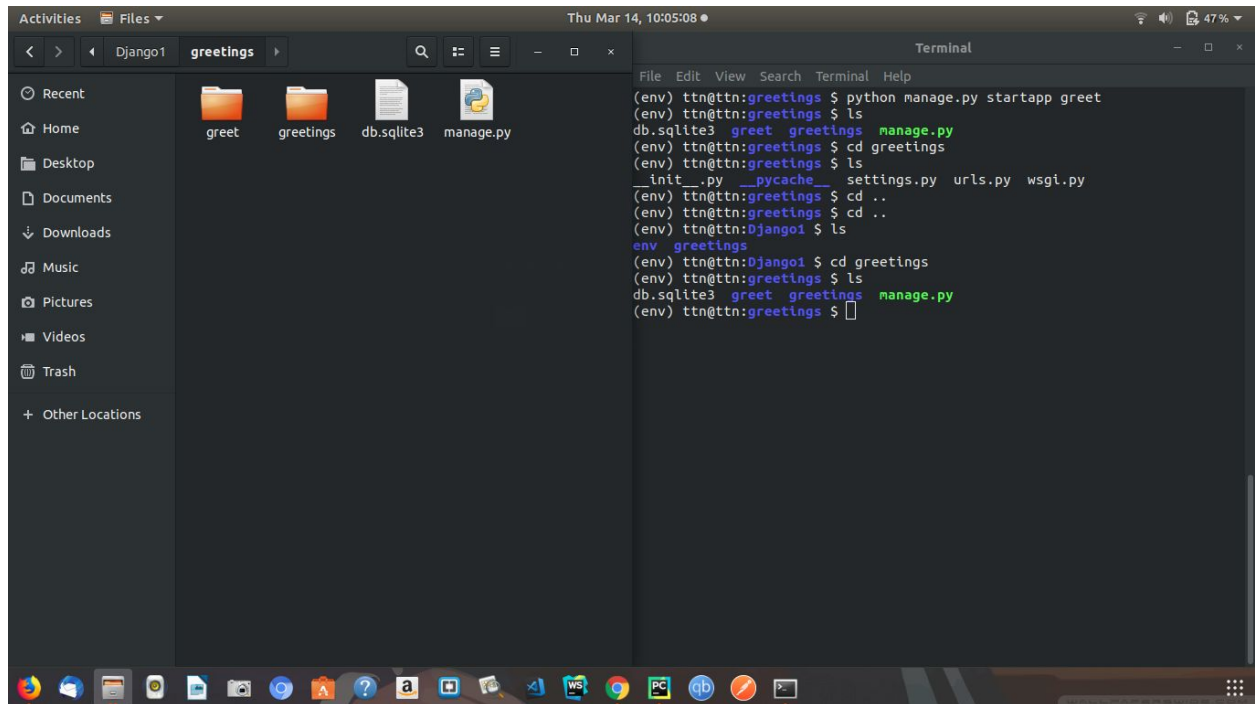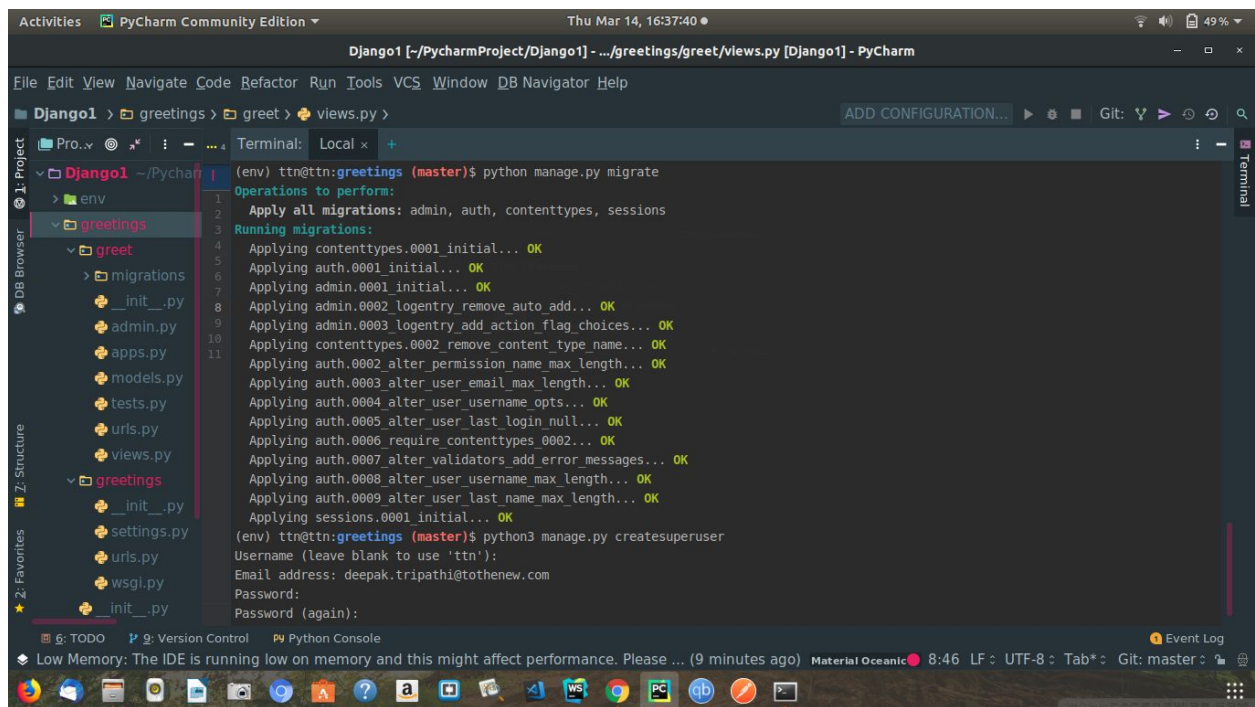
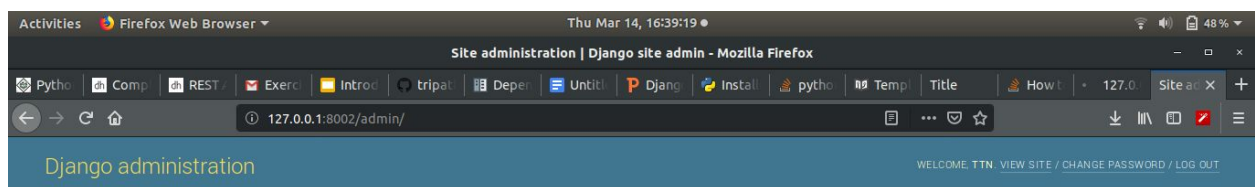Q9 Define a view which will render Good Morning!! or Good Evening!! as per current system time.

Ans

https://github.com/tripathideepak1997/Django1/tree/master
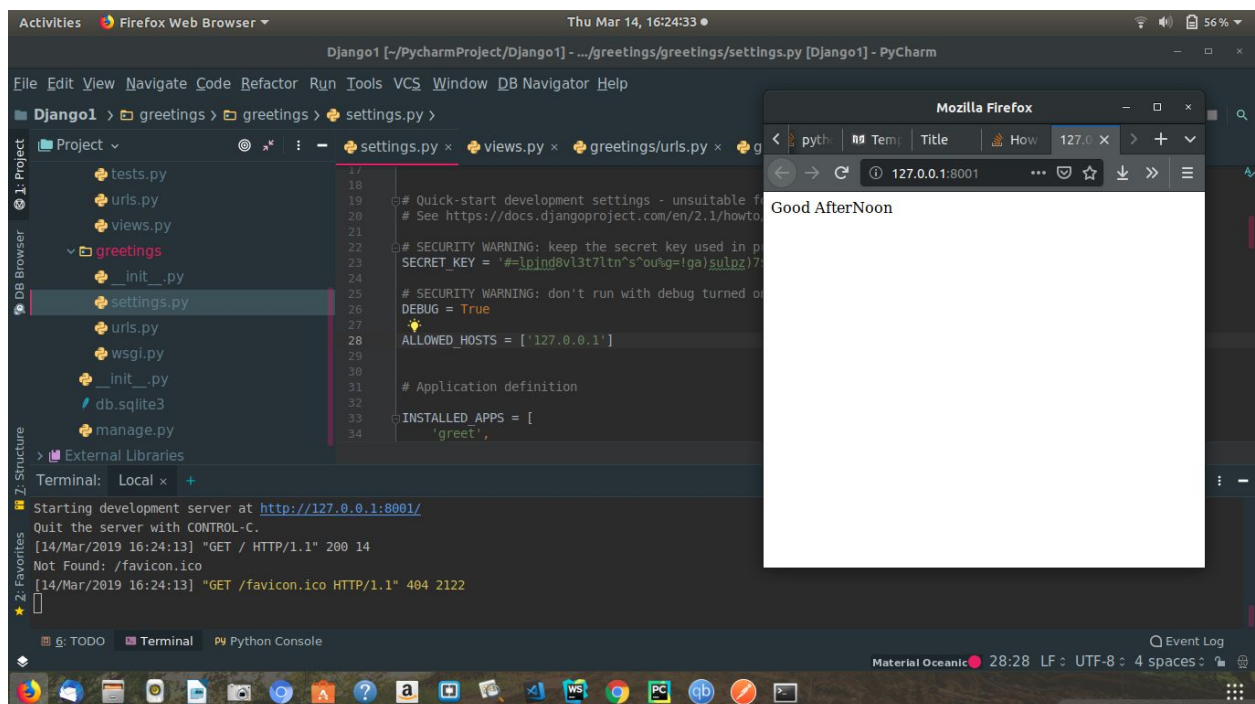
Q10 Register urls of admin portal at url mycustomadmin/, create super user and run it.
Ans