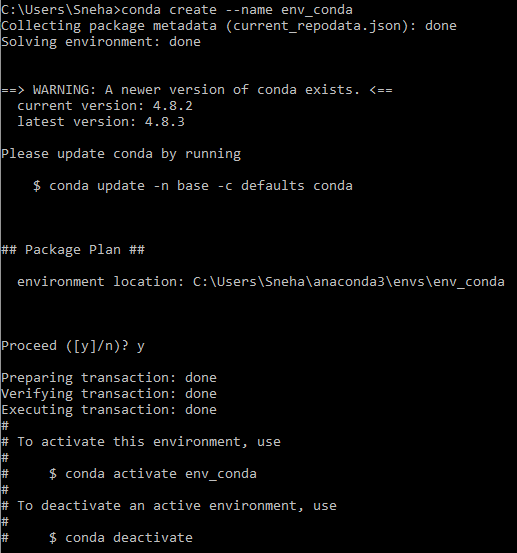
**DAY -10 ESTIMATE  (1-2 hours)**

**TASK 01:**

* Create **virtual environment** using
  + Anaconda
  + Virtualenv
  + venv

Name them as **env\_conda**, **env\_virtual\_env** and **env\_venv** respectively.

**env\_conda**

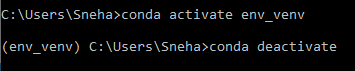




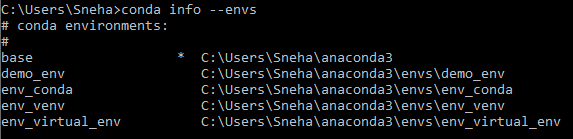
**env\_virtual\_env**



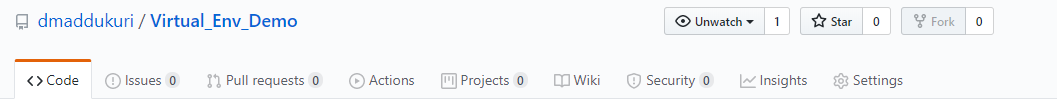
**env\_venv**



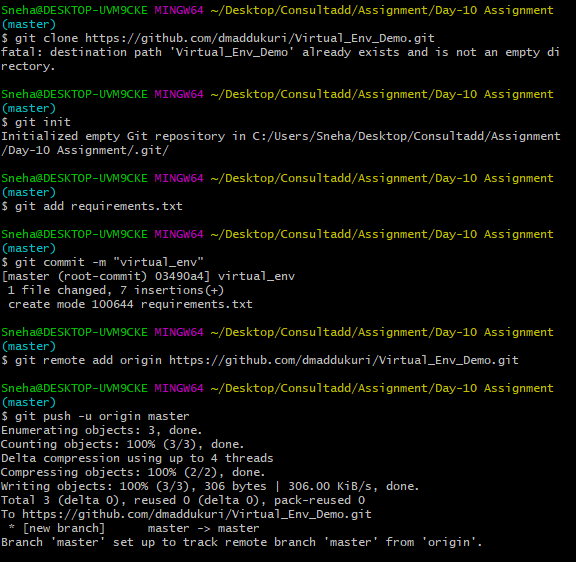
**Conda info --envs**



* Create a Git Repository with the name Virtual\_Env\_Demo

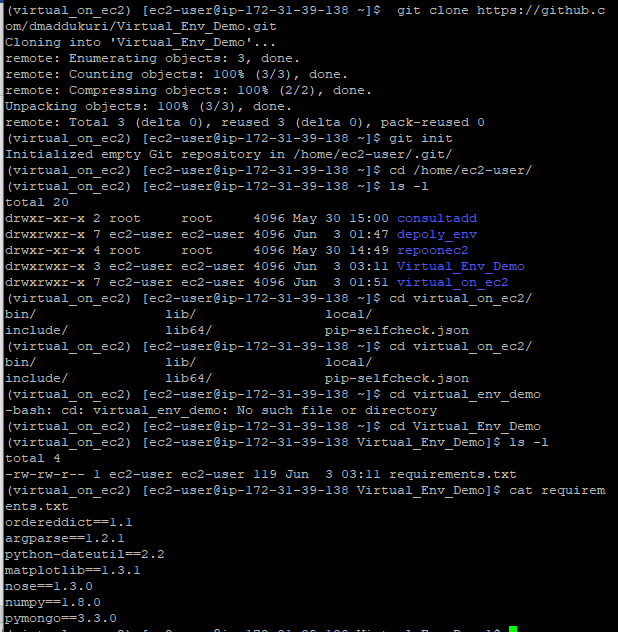


* + Make sure to have one file inside the repo with the name called requirements.txt
    - The content of requirements.txt is

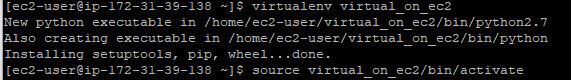


|  |
| --- |
| ordereddict==1.1  argparse==1.2.1  python-dateutil==2.2  matplotlib==1.3.1  nose==1.3.0  numpy==1.8.0  pymongo==3.3.0 |

* + Clone the repository on EC2 Machine.



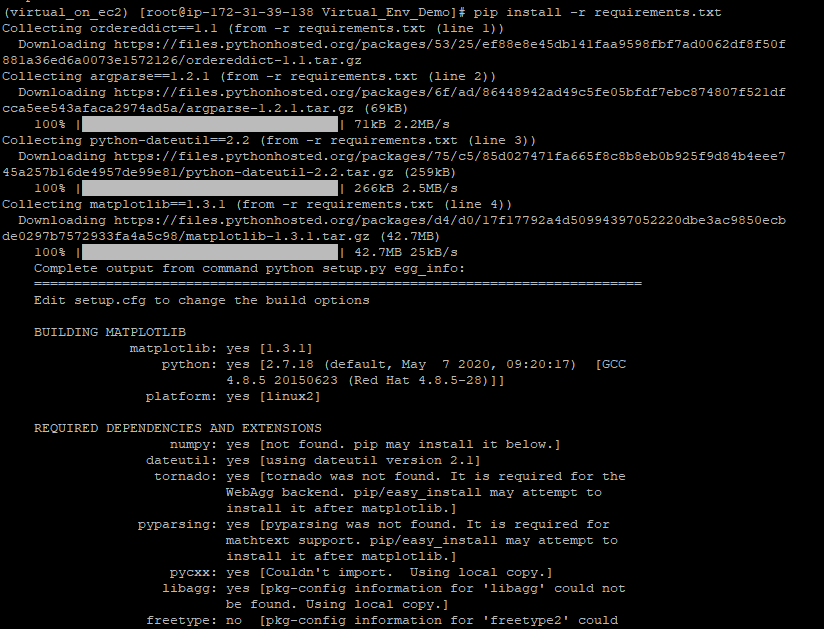
* + Make sure to create a virtual environment on EC2 using virtualenv and name it as Virtual\_on\_ec2

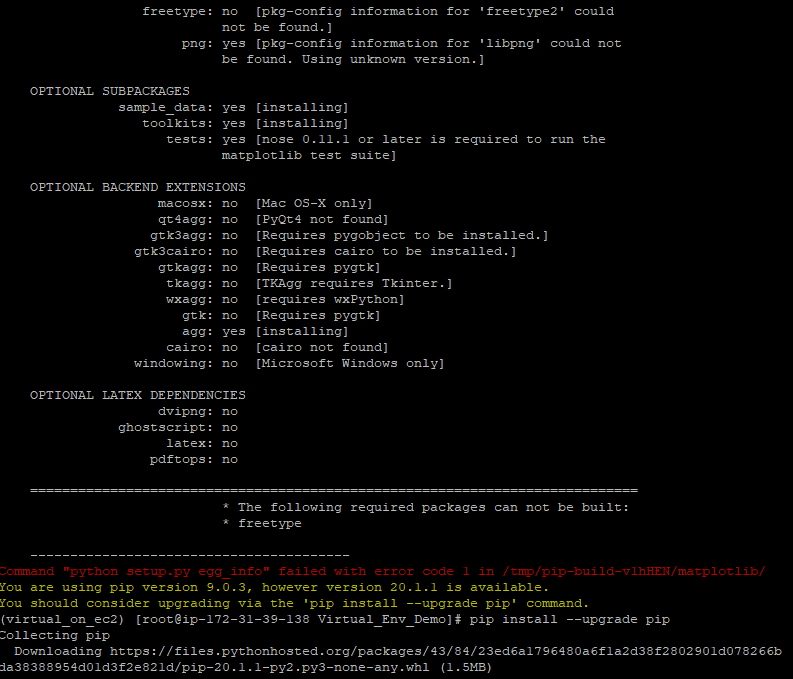


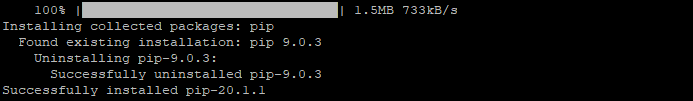
* + Activate the virtual environment



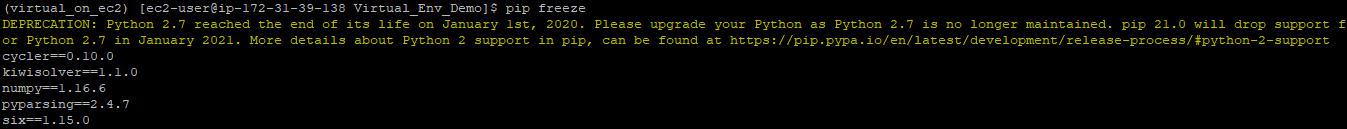
* + Install all the dependencies from cloned requirements.txt



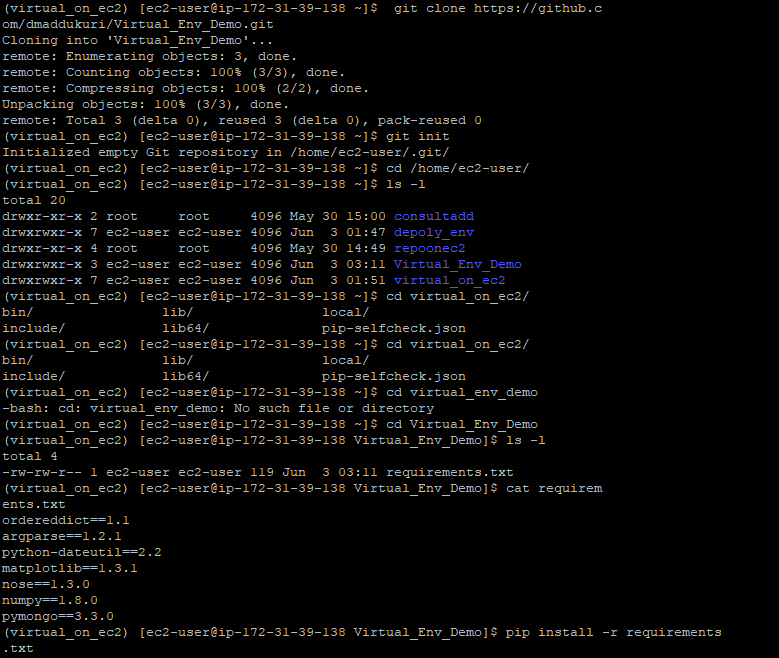




* + do **PIP FREEZE**



* + **Share the Screenshot :)**



**TASK 02:**

* Learn About STATIC AND DYNAMIC WEBSITE

**Introduction to STATIC WEBSITES**

**DEFINITION:**

* Static web pages are made of “fixed code,” and unless the site developer makes changes, nothing will change on the page. Think of it like a brochure for a business. Static sites give a lot of the same type of information that you could get from a brochure, but it can’t just change itself. In order to do this, someone has to create a new page. That’s why static websites are sometimes referred to as brochure sites.

Nothing is stored but the actual pages of a static site. There are:

* No users
* No comments
* No blog posts
* No interactivity

A static website is delivered to a user exactly the way it’s stored. That means that nothing on the page will change by the user or even the site administrator unless there’s a redesign of the site, or the site administrator goes directly into the code to change it.

You’ll often hear static sites called websites and dynamic sites called web apps. It all goes back to the idea that there are websites and web applications. A web application is a website, but a lot of websites can’t be web applications. For example, Facebook is a website and a web application. However, a business’s simple website is not a web application.

A static site is the most basic kind of website, and the easiest to create. It requires no server-side (also called back-end) processing, only client-side. Client-side technologies are [HTML](https://www.pluralsight.com/search?q=html), [CSS](https://www.pluralsight.com/search?q=css), and [JavaScript](https://www.pluralsight.com/search?q=javascript&clm_id=5dfbf8f72fc35849006fea27&CLM_Id__c=5dfbf8f72fc35849006fea27). No programming languages, including JavaScript, are required to make a static site. However, if a site utilizes JavaScript, but no [PHP](https://www.pluralsight.com/search?q=php) or any other programming language, it’s still considered a static site (since JavaScript is a client-side language).

So, if you want a site only to give information that doesn’t need to be updated regularly, creating a static website is a simple and effective way to go.

**Introduction to DYNAMIC WEBSITES**

**DEFINITION:**

* There’s a simple way to determine if a site is dynamic. If you can interact with it, it’s a dynamic site. So, most of the sites you probably visit are dynamic sites because they are interactive. For example, dynamic sites allow you to create a user profile, comment on a post, or make a reservation.

Dynamic site examples include:

* E-commerce sites
* Blogs
* Calendars, or to-do sites
* Any site with information that must be updated regularly

You might hear that PHP and ASP.NET are used to generate HTML dynamically. That means that those programming languages can, with direction, change and write HTML without a person having to go into the code and change it.

Dynamic sites use languages like PHP to interact with information stored in databases. These types of languages used to create dynamic sites are also much more complicated than the client-side languages. Plus, not only is web hosting required, but database or servers must be created as well.  For this reason, dynamic sites are much more complicated and expensive to create.

Most dynamic sites utilize a Content Management System to, you guessed it, manage their content. Often, developers will create a custom CMS for their clients (using PHP and MySQL), but that’s not necessary. There are a lot of free systems available for your use, such as WordPress, Drupal, and Joomla.

A term often associated with dynamic sites is CRUD, which stands for the following four things:

* Create
* Read
* Update
* Delete

All four of these things happen when you’re working with a dynamic site because they refer to the functionality of a database. Think about a blog. In any successful blogging platform, you have the ability to create content, then be able to read or view that content on the page. You can update or edit your posts and have the option to delete them as well. All that work happens in the database. Content management systems make this process possible.