Week 4:

Day1:
Started with the basics of tableau the goal was to create a dashboard from a dataset named discount Mart. We created six individual sheets of total sales,total profit, total quantity sold, sales by region, sales by category, and a bin chart of quantity to observe which customers bought two or more items together. We created a bar chart for sales by category, a map chart for sales by region, tiles for profit,sales and quantity. We applied filter to only one sheet. Then proceeded to create a new dashboard and clubbed all the sheets together. And applied the filter to every chart. We added the logo of discount Mart with the image object and then hosted the dashboard on our tableau public account.

Day2:
We revisited the crud operations of mongodb in the first half in the second half we learned how to deploy a mongodb cluster on the cloud and how to connect it with the mongo shell in our local environment. We were tasked to connect mongo db atlas with tableau.

Day3: On day 3rd we started with node.js we learned about npm packages and how to include them in our project. We installed node.js and from npm installed packages like express and body- parser in our package.json file. We created a templates folder and made a index.html file and rendered a form in it. And through the post method we sent the form data to the app.post method. Now by importing the body parser module we just installed we were able to catch the data on the command line. Now we were tasked to store the form data in either atlas or mongo db shell. We were also tasked with a newsletter functionality which prevents duplicate entries.

Day4: We created an api using node.js and a inbuilt database within the code itself. We used the four basic methods of Rest Api which are get,post,put and delete. Using get we were able to fetch the data,using post we were able to add data to the database, delete was used to remove database entries and put was to update them. We made use of postman for testing our api. We made use of joi for validation and applying restrictions on the input the body parser to catch the input and express for the boiler plate code. We were tasked to improve the api by creating a database in mongo shell.

Day5: We were tasked to create a voting system which will have a voting page where we will have four buttons to vote for our favourite party like A,B,C,D. We created a single record in our database with some default values for the teams and then every time a button is pressed the votes of that particular team should get incremented by one and then on a chart. html page we had to show the analysis done for the same. We used the axios library to fetch the data from the get request in our index.js file and then storing the response in variables we created a Google chart and replaced the data with the variables. At the end of the day we were given a challenge to implement signup and login page and store the password in hash format as well as implement a forgot password functionality and a social media login as well.

Week 3

Day 1:
Started basics of react, learned about the folder structure of react and that everything is

rendered in root div. You can make individual components and call them in the main index.js which is the entry point.

Day 2:
Learned about the browser router, routers and route to link different pages. We use link to to link the different routes. The browser router is the parent tag. Inside it are the routes and route tag we use either path aur exact path to link. We also learned about the usestate hook which captures the state of an element. The usestate hook use a variable and a function that operates on the variable. We use setstate to update or change the variable.

Day3:
In the first half we had our open house in the second half we proceeded with the installation of mongodb we installed the mongo shell as well as it's community version. We learned the insert one, insert many, update one, update many, find one or find all as well as delete one delete all commands.

Week2

Day1
Started with a new project which is a blogging website. Created a model with cover image and gave parameter upload to in it. To upload images on server side we made some changes in the settings.py and urls.py which creates a media folder when we add a image to the server. We were able to dynamically render the blogs using objects.all() method and also limited the content display in the card using slice operator. And on click the full blog can be visible.

Day2:
Learned pagination using paginator class and started a new project a portal for blood bank. Created a model and displayed it in the form of a table and added pagination to it.

Day 3:
Learned how to make use of built in django models to use the login logout functionality and made use of usercreationform to create user and render signup form,using authenticationform rendered login form, made use of decorators to restrict access to particular content made use of login required decorator. For the forms we use django.contrib.auth module and also used authenticate function to authenticate the user. Later on we used the ngrok platform for deploying the website on a temporary basis and made use of social-share module to allow the sharing of blogs on various social media platforms.

Day4:
Learned Api handling using python with the urlib.request module and json module. Made use of urlopen function to fetch the json data from server then jsonified it using the json.loads function and then decode the json data to get what we needed and rendered it on the template. In this Api no api key was required. In the second we learned how to deploy our project on python anywhere which allows us to host the website for three months. We made use of git bash commands to first upload our project on GitHub and used git clone to copy it the files directory on python anywhere. We then made some changes in the wsgi file and and setting.py file by setting the allowed hosts to everyone and then finally deployed the website.

Day5:
Today sir challenged us with two projects one was a feedback system and second one was a suggestion Box project. In the first project we have to display three emojis good, bad and best and the user has to give it's feedback. The admin should be able to view how many people have chosen good,bad or best and it should be displayed in a visualised form like a bar chart or pie chart.

Full Stack Internship

Week1
Day1:
On the first day we started with the basics of html CSS and JavaScript. We created projects on calculator, weather API, Made use of stocks api and Google charts, created a stopwatch and to-do list. Learned how to get data through an api and how to use it in the code

Day2:
Started with django framework and made use of Bootstrap snippets to render the html files. Used virtual environment and installed django in it. We created a website for travel and added a newsletter which we created in the models.py. learned how to create a superuser a model, how to register it. Added templates folder

Day3:
Created a contact form using crispy library to render forms easily whole day was spent on this only.

Day4:
We added a carousel and learned how to load static files in django. We rendered cards and parallax also from bootstrap snippets. Learned how to handle web interactions

Day5:
Completed the whole website added navbar, footer, testimonials and a aboutus page. Ready to be hosted on pythonanywhere.com