## Creating the Model

### 14:30 Saving the model

## Model Prediction Function

0:30 Explaning about deploying models

0:50 Explaining what JSON files are

1:20 Setting up the dictionary

3:15 Making a function to return predictions from JSON file

3:30 Grabbin information from JSON file to work with

7:30 Finishing the function

8:10 Writing code for deployment

## Running a Basic Flask Applicatoin

0:50 Explaninig how it all works with Flask

2:45 Downloading and installing Sublime editor so that Jupyter notebook doesn’t interfere with web properties of Flask

4:25 Writing the script in Sublime

8:00 Running the script from the console

10:00 Taking the local IP address to test on local web browser

## Flask Postman API

1:00 import libraries

2:00 Load model that was trained before

2:30 Load scaler

5:15 Running the application from python console

6:00 Downloading Postman App – Nice graphic interface for sending JSON commands

7:00 Sending request through the Postman App

8:00 Selecting the Post request as an example

8:45 Putting the IP address for the Post command

9:30 Setting up the command

10:15 Sending the request

## Flask API – Using Requests Programmatically

0:30 Need 2 instances of anaconda consoles – setting up the correct environment

1:45 Opening up Jupyter notebook with commands

3:00 Checking the result

## Flask Front End

0:30 This is for normal people to interact with the application

1:00 Explaining the things that are required to do this

1:15 Explaining the App and what files you need

2:30 Editing the python code

3:15 New imports

4:15 Setting up secret key

5:15 Setting up a new class for FlowerForm that will take in some parameters to transfer to a Flask form

7:15 Writing function to use the Flask form

9:45 Redirect command to URL

11:30 Changing the prediction function to make an empty dictionary and then input the parameters collected from the form

15:00 Looking at the HTML file for Home

17:00 Looking at the HTML file for Prediction

17:30 Using Anaconda Prompt to run the Flask application through the python file

19:00 Trying out the HTML file

## Live Deployment to the Web

1:00 Explaining main steps that will be done

2:00 Grabbing the necessary files from the course

2:30 Changing the name of a file to app.py

3:00 Create Heroku account

5:00 Making a deployment environment in Anaconda Prompt

6:45 Install libraries through pip to the new environment

8:30 Making a requirements.txt required for Heroku

9:30 Making another necessary Procfile

10:30 Creating new app through Heroku

11:30 Go to Anaconda Prompt and login to Heroku

12:00 Create new Git repository and sync/push to Heroku

13:30 Explaining why a Push might get rejected error

14:30 Fixing the problem by changing the list of libraries through a text editor

16:45 Copy the URL supplied through the console and use it on a browser