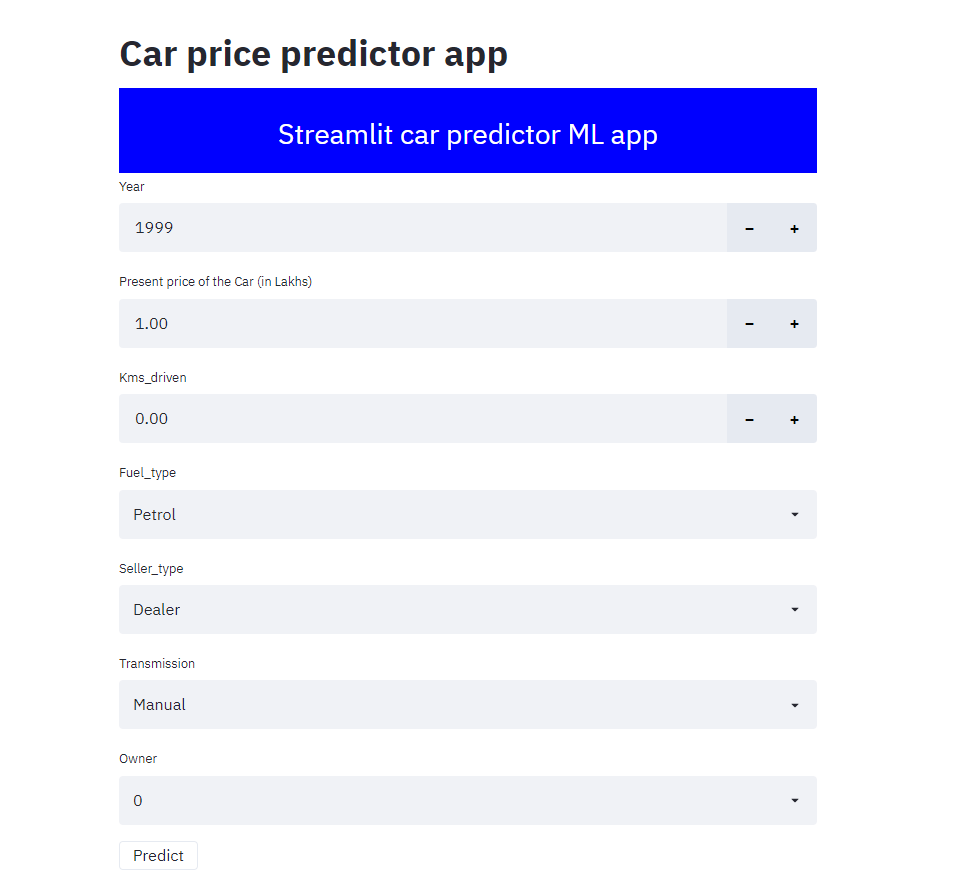
# Streamlit Web app development



(Above is the screenshot of our streamlit car price predictor app)

**What is Streamlit?**

[Streamlit](https://streamlit.io/) is an open-source Python library that makes it easy to build beautiful custom web-apps for machine learning and data science.

In order to install and work with Streamlit, please follow this below steps –

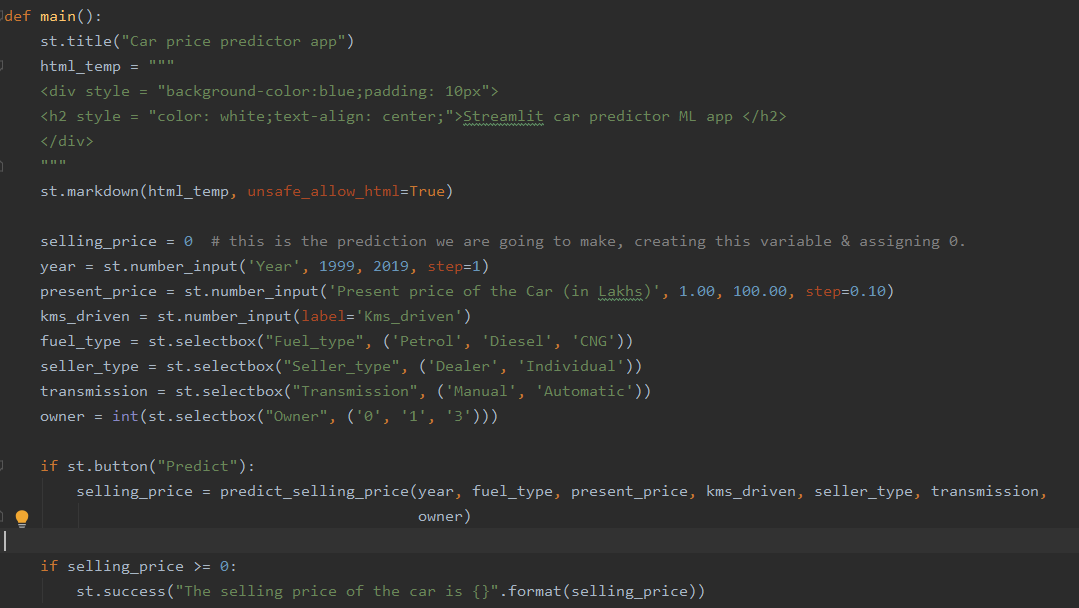
1. Make sure you have **Python 3.7 or above**.
2. Install Streamlit using pip, use the command pip install streamlit (\*\*\*tip: please do not install streamlit into your base environment, it is always better to create one separate environment first then install streamlit using the above command)
3. Once the installation is done, run the below command to check whether streamlit is working fine or not –

streamlit hello

1. That’s it! In the next few seconds the sample app will open in a new tab in your default browser.

Now we will discuss about our streamlit app in detail –

The basic structure that a streamlit app follows is there should be a main function and which will execute the streamlit app



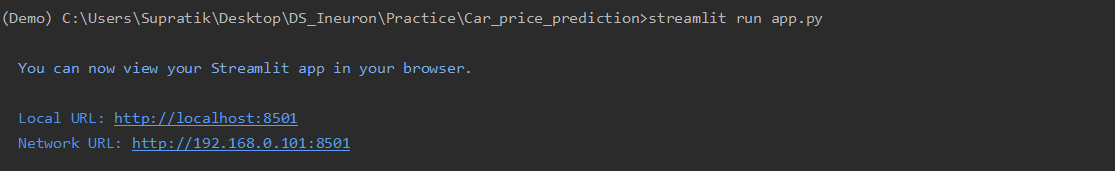


You can see from the above screenshots that we are calling only main() function to execute the app, and from our main() function we are calling different functions and also in the main function we are building our basic UI.

You can’t run the streamlit application in the same way as you run a python application, to run the streamlit app there are two ways –

* Open your pycharm project and go to the terminal and execute the below command –

streamlit run [filename]



* If you are using anaconda distribution, then open anaconda prompt and go to your project location and run the same above command.

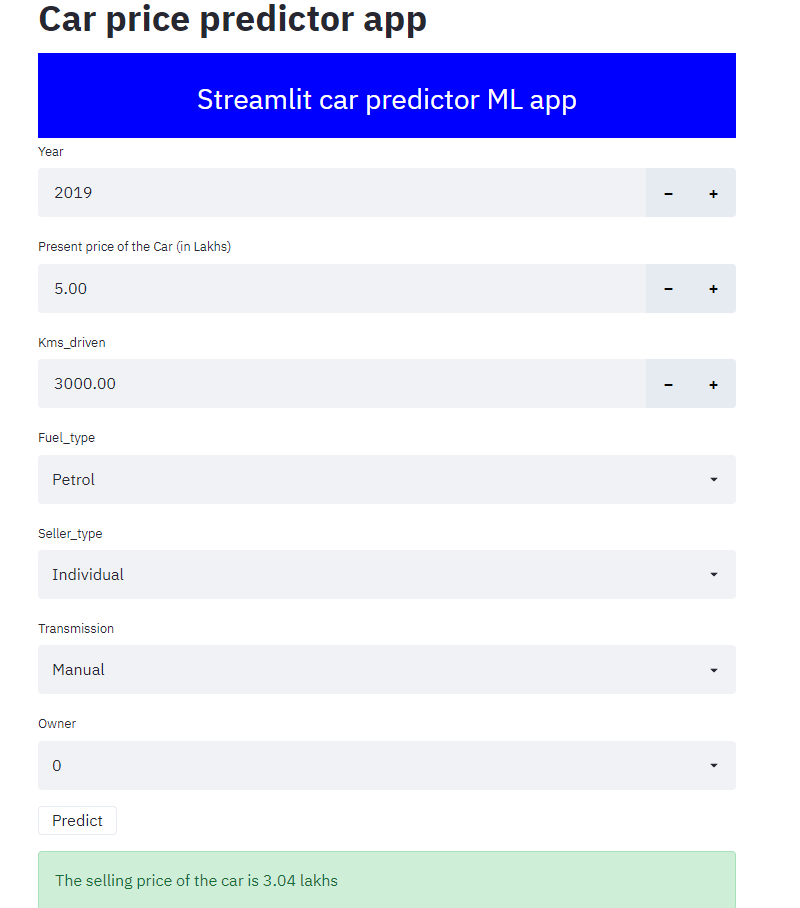
**Demonstration of the application**:

(link of the app: <https://immense-bastion-82042.herokuapp.com/>)

Basically our app approximately predicts the selling price of a car, for that purpose it takes some inputs from the user regarding the car then it predicts the possible selling price. Now, we will discuss the input parameters that are being taken from the end-users.

* **Year** – year of buying of the car, here we are not considering cars bought before 1999, so enter any year between 1999 and 2020
* **Present price of the Car (in Lakhs) –** accepted values for this field is any 2 decimal place value between 1 lakhs to 99 lakhs.
* **Kms\_driven –** Kilometers driven by the car, generally it can take any number (this is also a 2 decimal place value)
* **Fuel\_type –** this is drop-down list, possible values are ‘Petrol’, ‘Diesel’ and ‘CNG’
* **Seller\_type –** this is also a drop-down list containing two values i.e. ‘Dealer’ or ‘Individual’
* **Transmission -** this is also a drop-down list containing two values i.e. ‘Manual’ or ‘Automatic’
* **Owner –** possible values of this field is 0/1/3. User can select any one of the values from the drop-down list.

Once all the required values of the above input parameters is given by the user then they can press the ‘Predict’ button to see the possible selling price of the car (a sample screenshot is given below).



This app is deployed in Heroku cloud platform, the steps of deploying Streamlit web app into Heroku cloud platform is given below, please follow the same to deploy any streamlit app into Heroku cloud platform.

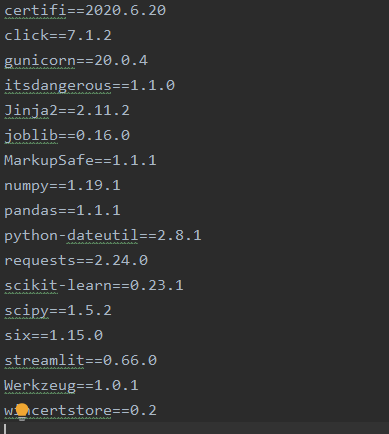
**Heroku app creation and deployment**

A few steps need to be done before the cloud deployment they are –

1. Add a file called ‘Procfile’ inside the project folder. This folder contains the command to run the flask application once deployed on the server:

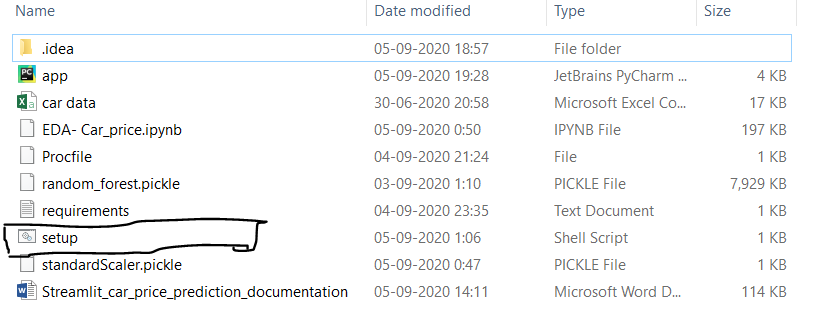


1. Open a command prompt window and navigate to your project folder. Enter the command ‘pip freeze > requirements.txt’. This command generates the ‘requirements.txt’ file. My requirements.txt looks like:



‘requirements.txt’ helps the Heroku cloud app to install all the dependencies before starting the webserver

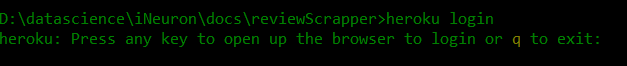
1. Now in order to deploy a streamlit app into Heroku cloud platform, you have to create a unix shell file or bash file named ‘setup.sh’ and place it your project folder.



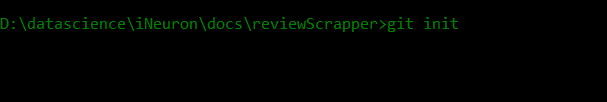
Once all the above steps are completed you can start the steps of deployment, also you need to have an account in Heroku cloud platform & need to install Heroku CLI in your machine.

**Steps for cloud deployment is given below** –

1. After installing the Heroku CLI, open a command prompt window and navigate to your project folder.
2. Type the command ‘heroku login’ to login to your heroku account as shown below:



1. Before deploying the code to the Heroku cloud, we need to commit the changes to the local git repository.
2. Type the command ‘git init’ to initialize a local git repository as shown below:



1. Enter the command ‘git status’ to see the uncommitted changes
2. Enter the command ‘git add .’ to add the uncommitted changes to the local repository.
3. Enter the command ‘git commit -am "make it better"’ to commit the changes to the local repository.
4. Enter the command ‘heroku create’ to create a heroku app. It will give you the URL of your Heroku app after successful creation.
5. Enter the command ‘git push heroku master’ to push the code to the heroku cloud.
6. After deployment, heroku gives you the URL to hit the web API.