

Text  Description automatically generated with low confidence

**Prediction of Blood donation Project**

**Submitted by- Uday Kumar**

Blood Donation Using ML

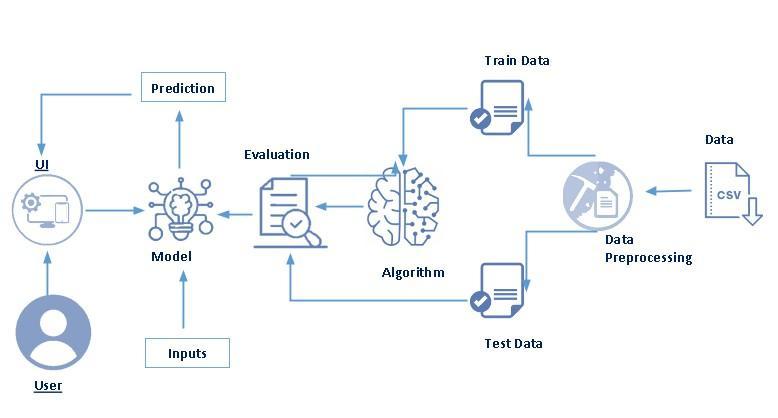
# Project Description:

The goal of this project is to develop a machine learning model that can predict the likelihood of individuals donating blood based on various factors. By analyzing historical data of blood donors, we aim to identify patterns and create a predictive model that can assist blood banks and organizations in targeting potential donors and improving blood supply management.

The dataset used in this project consists of records of blood donors, including features such as recency (months since last donation), frequency (number of times donated), monetary (volume of blood donated), time (months since first donation), and the target variable indicating whether the donor donated blood in a specific period (e.g., a particular month).

The main objective of this project is to build a robust machine learning model that can accurately predict blood donation behavior. By analyzing the donor's historical data, including recency, frequency, monetary value, and time, the model will learn the patterns associated with blood donation and make predictions on whether a person is likely to donate blood in a given period.

# Technical Architecture:



# Pre requisites:

**To complete this project, you must required following software’s, concepts and packages**

* **Anaconda navigator and pycharm:**
  + Refer the link below to download anaconda navigator
  + Link : <https://youtu.be/1ra4zH2G4o0>
* **Python packages:**
  + Open anaconda prompt as administrator
  + Type “pip install numpy” and click enter.
  + Type “pip install pandas” and click enter.
  + Type “pip install scikit-learn” and click enter.
  + Type ”pip install matplotlib” and click enter.
  + Type ”pip install scipy” and click enter.
  + Type ”pip install pickle-mixin” and click enter.
  + Type ”pip install seaborn” and click enter.
  + Type “pip install Flask” and click enter.

# Prior Knowledge:

You must have prior knowledge of following topics to complete this project.

* **ML Concepts**
  + Supervised learning: <https://www.javatpoint.com/supervised-machine-learning>
  + Unsupervised learning: <https://www.javatpoint.com/unsupervised-machine-learning>
  + KNN: <https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning>
  + Logistic Regression: [https://www.analyticsvidhya.com/blog/2018/09/an-end-to-end-guide-to-understand-the-math-behind-logistic-regression/](https://www.analyticsvidhya.com/blog/2018/09/an-end-to-end-guide-to-understand-the-math-behind-xgboost/)
  + Evaluation metrics: <https://www.analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/>
* **Flask Basics** : <https://www.youtube.com/watch?v=lj4I_CvBnt0>

# Project Objectives:

By the end of this project you will:

* Know fundamental concepts and techniques used for machine learning.
* Gain a broad understanding about data.
* Have knowledge on pre-processing the data/transformation techniques on outlier and some visualization concepts.

# Project Flow:

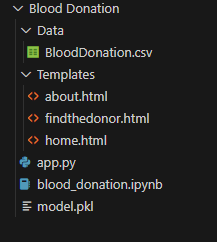
* User interacts with the UI to enter the input.
* Entered input is analyzed by the model which is integrated.
* Once model analyses the input the prediction is showcased on the UI

To accomplish this, we have to complete all the activities listed below,

* Data collection
  + Collect the dataset or create the dataset
* Visualizing and analyzing data
  + Univariate analysis
  + Bivariate analysis
  + Multivariate analysis
  + Descriptive analysis
* Data pre-processing
  + Checking for null values
  + Handling outlier
  + Handling categorical data
  + Splitting data into train and test
* Model building
  + Import the model building libraries
  + Initializing the model
  + Training and testing the model
  + Evaluating performance of model
  + Save the model
* Application Building
  + Create an HTML file
  + Build python code

# Project Structure:

Create the Project folder which contains files as shown below



* We are building a flask application which needs HTML pages stored in the templates folder and a python script app.py for scripting.
* Model.pkl is our saved model. Further we will use this model for flask integration.
* Data folder contains dataset and Templates folder contains html files.

# Milestone 1: Define Problem / Problem Understanding

**Activity 1: Specify the business problem**

The problem statement for the blood donation machine learning project is to develop a predictive model that can accurately forecast the likelihood of individuals donating blood based on their historical data. The aim is to create a model that can identify patterns and factors that influence blood donation behavior, allowing blood banks and organizations to target potential donors more effectively and improve blood supply management.

The ultimate goal of this project is to develop a machine learning model that can assist blood banks and organizations in making informed decisions, optimizing resource allocation, and designing targeted campaigns to encourage blood donation.

## Activity 2: Business requirements

Here are some potential business requirements for Share price predictor.

## Accurate forecasting: The predictor must be able to accurately forecast Blood Donation behaviour for a given information about blood donors. The accuracy of the forecasting is crucial for blood banks and other organizations to make informed decisions on the blood donation.

## Real-time data acquisition: The predictor must be able to acquire real-time data on donors information and other relevant factors that affect blood donation prediction. The data acquisition must be seamless and efficient to ensure that the predictor is always up-to-date with the latest information.

## User-friendly interface: The predictor must have a user-friendly interface that is easy to navigate and understand. The interface should present the results of the predictor in a clear and concise manner to enable doctors and other medical staff to make informed decisions.

* 1. Report generation: Generate a report outlining the predicted best blood donor in a particular region, along with an analysis of the factors that have contributed to their performance. The report should be presented in a clear and concise manner, with appropriate visualizations and insights to help stakeholders make informed decisions.

## Activity 3: Literature Survey (Student Will Write)

A literature survey would involve researching and reviewing existing studies, articles, and other publications on the topic of project. The survey would aim to gather information on current systems, their strengths and weaknesses, and any gaps in knowledge that the project could address. The literature survey would also look at the methods and techniques used in previous projects, and any relevant data or findings that could inform the design and implementation of the current project.

**Activity 4: Social or Business Impact.**

The Blood Donation Analysis project can have both social and business impacts.

**Social Impact:**

The accurate prediction of Blood Donation behaviour can help individual acceptors,blood donation organisations particularly those who may not have extensive knowledge on donation and its bahaviour, make informed decisions, thereby enabling them to achieve their goals and potentially saves a life.

The project can also contribute to blood supply management by providing more accurate information, helping them make informed decisions, and reducing blood volatility and uncertainty.

**Business Impact:**

The project can be particularly useful for blood donation organizations, who make donation decisions on behalf of acceptors. Accurate prediction of best donors can help these organizations improve their donating strategies and potentially decrease the hurdles in finding suitable donors.

Accurate blood donation prediction can also be useful for companies operating in the medical sector, enabling them to anticipate changes in the need for blood and adjust their strategies accordingly. This can help to maintain ecological balance and reduce blood scarcity.

Overall, the project can have a positive impact on the medical industry by improving the accuracy of health and increasing life efficiency.

# Milestone 2: Data Collection and Visualizing and analyzing the data

ML depends heavily on data, It is most crucial aspect that makes algorithm training possible. So this section allows you to download the required dataset.

**Activity 1: Download the dataset**

There are many popular open sources for collecting the data. Eg: kaggle.com, UCI repository, etc.

In this project we have used BloodDonation.csv data. This data is downloaded from kaggle.com. Please refer the link given below to download the dataset.

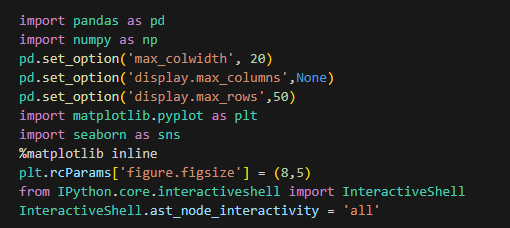
Link: https://www.kaggle.com/code/alperkaraca1/blood-donor-prediction/input

As the dataset is downloaded. Let us read and understand the data properly with the help of some visualization techniques and some analysing techniques.

**Note: There is n number of techniques for understanding the data. But here we have used some of it. In an additional way, you can use multiple techniques.**

**Activity 2: Importing the libraries**

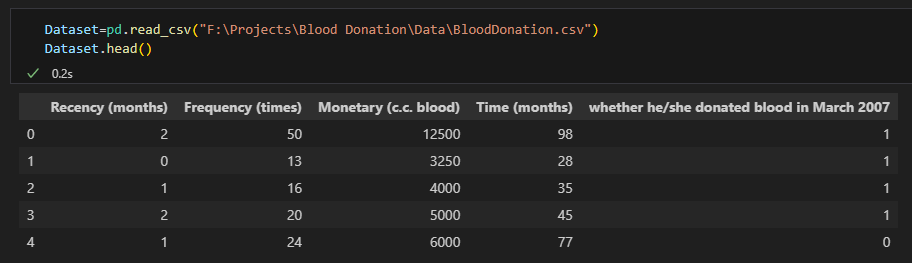
Import the necessary libraries as shown in the image. Here we have used visualization style as five thirty eight.



**Activity 3: Read the Dataset**

Our dataset format might be in .csv, excel files, .txt, .json, etc. We can read the dataset with the help of pandas.

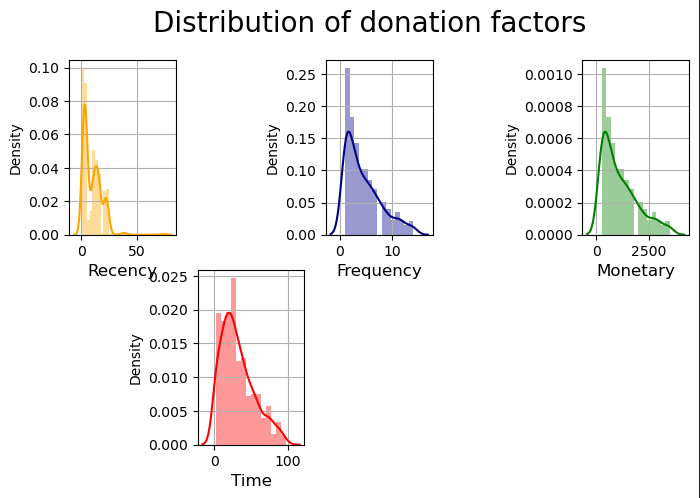
In pandas we have a function called read\_csv() to read the dataset. As a parameter we have to give the directory of csv file.



**Activity 4: Univariate analysis**

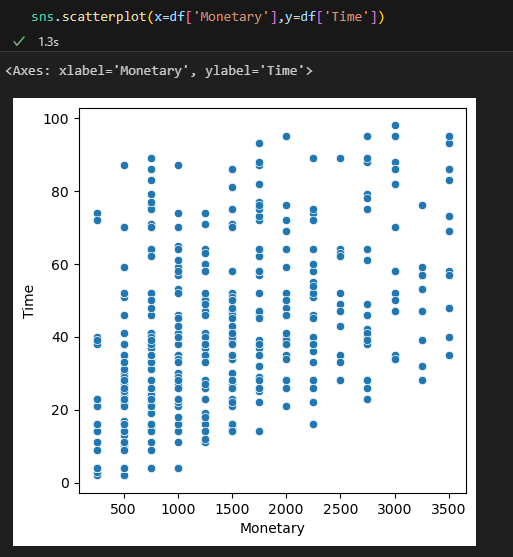
In simple words, univariate analysis is understanding the data with single feature. Here we have displayed two different graphs such as distplot and countplot.

* Seaborn package provides a wonderful function distplot. With the help of distplot, we can find the distribution of the feature. To make multiple graphs in a single plot, we use subplot.



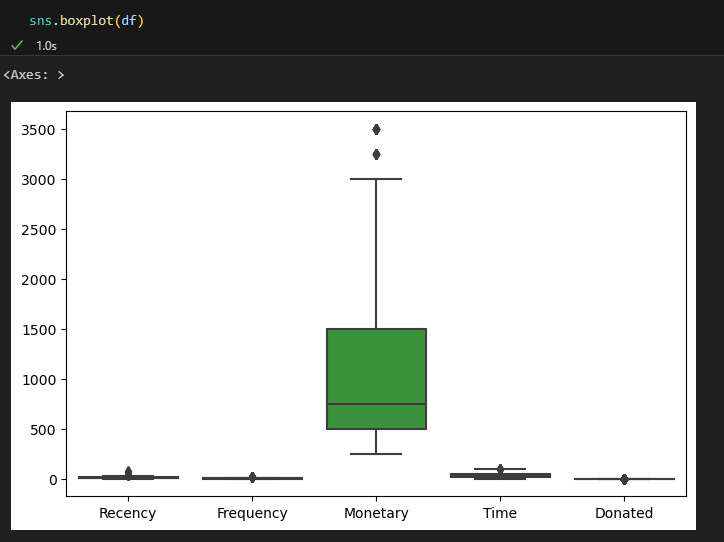
**Activity 5: Bivariate analysis**

To find the relation between two features we use bivariate analysis. Here we are visualizing the relationship between ‘Monetary’ and ‘Time’ variables.



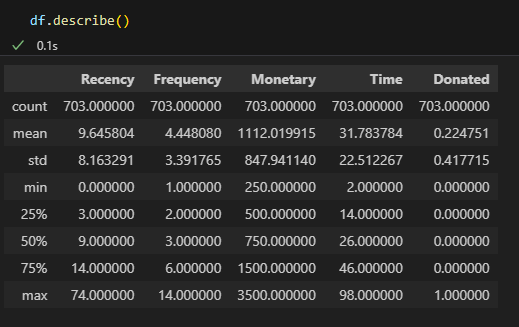
**Activity 6: Multivariate analysis**

In simple words, multivariate analysis is to find the relation between multiple features. Here we have used boxplot from seaborn package.



**Activity 7: Descriptive analysis**

Descriptive analysis is to study the basic features of data with the statistical process. Here pandas has a worthy function called describe. With this describe function we can understand the unique, top and frequent values of categorical features. And we can find mean, std, min, max and percentile values of continuous features.



# Milestone 3: Data Pre-processing

As we have understood how the data is lets pre-process the collected data.

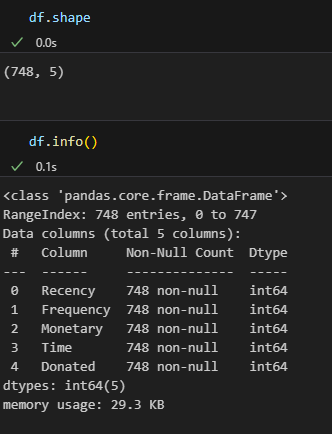
The download data set is not suitable for training the machine learning model as it might have so much of randomness so we need to clean the dataset properly in order to fetch good results. This activity includes the following steps.

* Handling missing values
* Handling categorical data
* Handling outliers
* Scaling Techniques
* Splitting dataset into training and test set

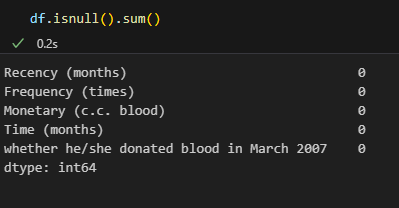
Note: These are the general steps of pre-processing the data before using it for machine learning. Depending on the condition of your dataset, you may or may not have to go through all these steps.

**Activity 1: Checking for null values**

* Let’s find the shape of our dataset first, To find the shape of our data, df.shape method is used. To find the data type, df.info() function is used.



* For checking the null values, df.isnull() function is used. To sum those null values we use .sum() function to it. From the below image we found that there are no null values present in our dataset. So we can skip handling of missing values step.

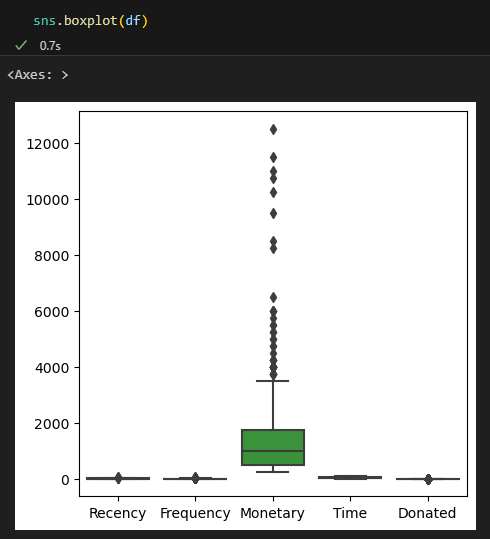


Let’s look for any outliers in the dataset

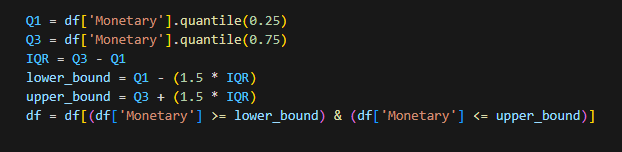
**Activity 2: Handling outliers**

With the help of boxplot, outliers are visualized. And here we are going to find upper bound and lower bound of all features with some mathematical formula.

* From the below diagram, we could visualize that Monetary feature has outliers. Boxplot from seaborn library is used here.

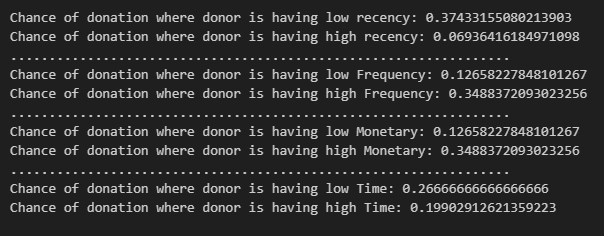


* To find upper bound we have to multiply IQR (Interquartile range) with 1.5 and add it with 3rd quantile. To find lower bound instead of adding, subtract it with 1st quantile. Take image attached below as your reference.



* To handle the outliers transformation technique is used. Here log transformation is used. We have created a function to visualize the distribution and probability plot of Monetary feature.

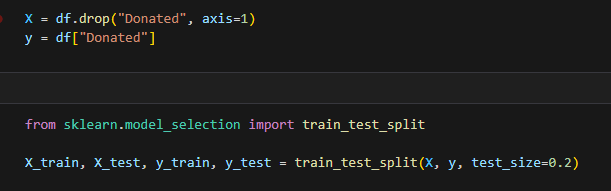
**Activity 3: Finding donation rate with high and low factors**



**Activity 4: Splitting data into train and test**

Now let’s split the Dataset into train and test sets. First split the dataset into x and y and then split the data set

Here x and y variables are created. On x variable, df is passed with dropping the target variable. And on y target variable is passed. For splitting training and testing data we are using train\_test\_split() function from sklearn. As parameters, we are passing x, y, test\_size, random\_state.

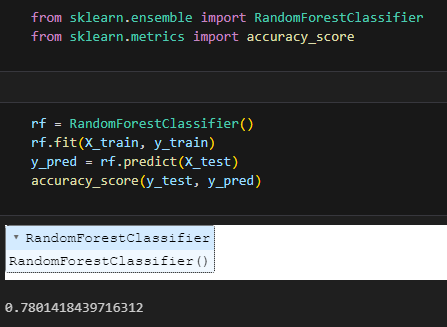


# Milestone 4: Model Building

Now our data is cleaned and it’s time to build the model. We can train our data on different algorithms. For this project we are applying four classification algorithms. The best model is saved based on its performance.

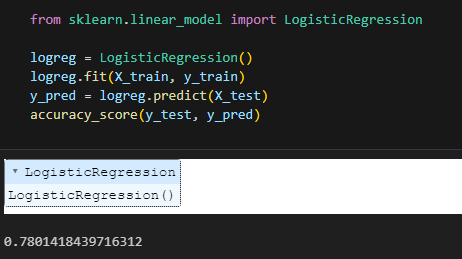
**Activity 1: Random Forest Classifier**

Random Forest Cassifier algorithm is initialized and training data is passed to the model with .fit() function. Test data is predicted with .predict() function and saved in new variable. For evaluating the model, confusion matrix and classification report is done.

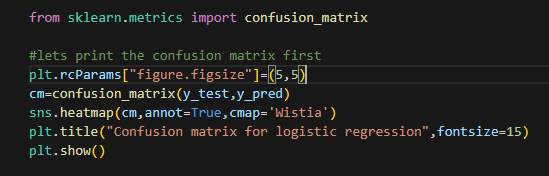


**Activity 2: Logistic Regression model**

LogisticRegression algorithm is initialized and training data is passed to the model with .fit() function. Test data is predicted with .predict() function and saved in new variable. For evaluating the model, confusion matrix and classification report is done.

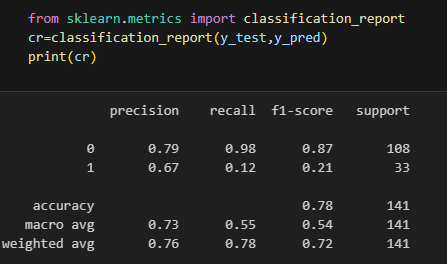


**Activity 3: Evaluating performance of the model and saving the model**

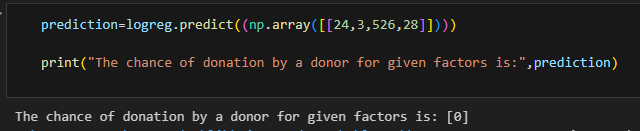


Now let’s see the performance of all the models and save the best model.

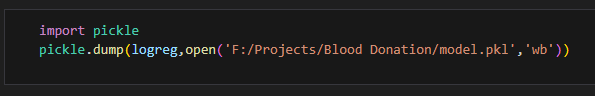




**Activity 4: Predict the chance of donation according to given factors**



Our model is performing well. So, we are saving the model by pickle.dump().



# Milestone 5: Application Building

In this section, we will be building a web application that is integrated to the model we built. A UI is provided for the uses where he has to enter the values for predictions. The enter values are given to the saved model and prediction is showcased on the UI.

This section has the following tasks

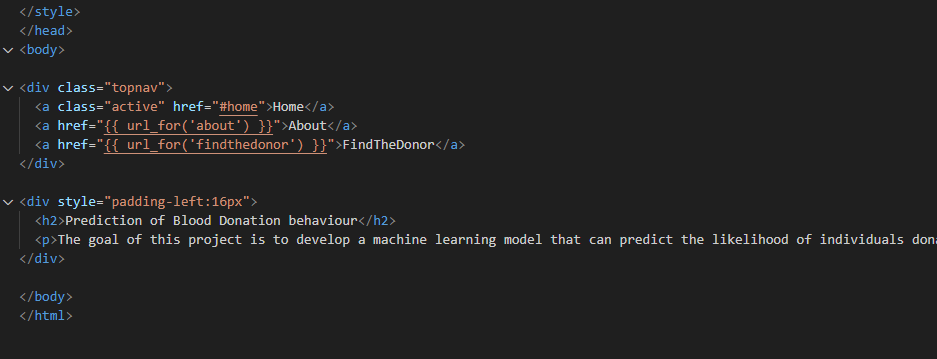
* Building HTML Pages
* Building serverside script

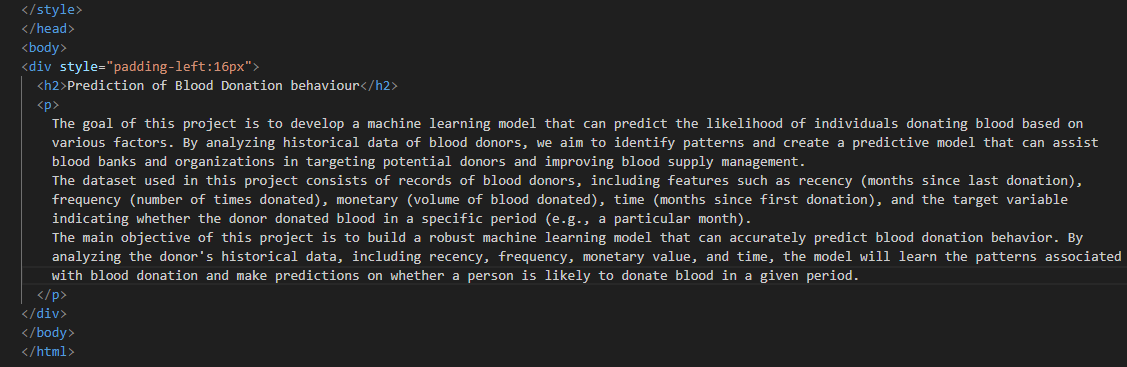
**Activity1: Building Html Pages:**

For this project create three HTML files namely

* home.html
* about.html
* findthedonor.html

and save them in Templates folder.

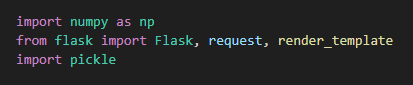




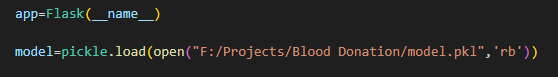


**Activity 2: Build Python code:**

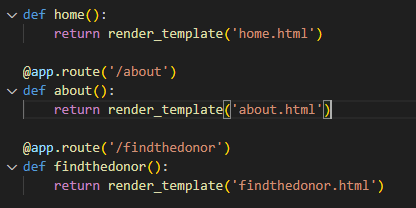
Import the libraries



Load the saved model. Importing flask module in the project is mandatory. An object of Flask class is our WSGI application. Flask constructor takes the name of the current module (\_\_name\_\_) as argument.



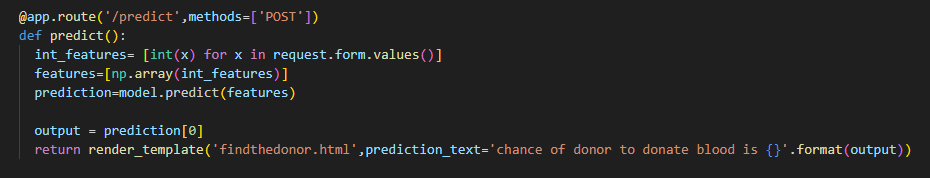
Render HTML page:



Here we will be using declared constructor to route to the HTML page which we have created earlier.

In the above example, ‘/’ URL is bound with home.html function. Hence, when the home page of the web server is opened in browser, the html page will be rendered. Whenever you enter the values from the html page the values can be retrieved using POST Method.

Retrieves the value from UI:



Here we are routing our app to predict() function. This function retrieves all the values from the HTML page using Post request. That is stored in an array. This array is passed to the model.predict() function. This function returns the prediction. And this prediction value will rendered to the text that we have mentioned in the submit.html page earlier.

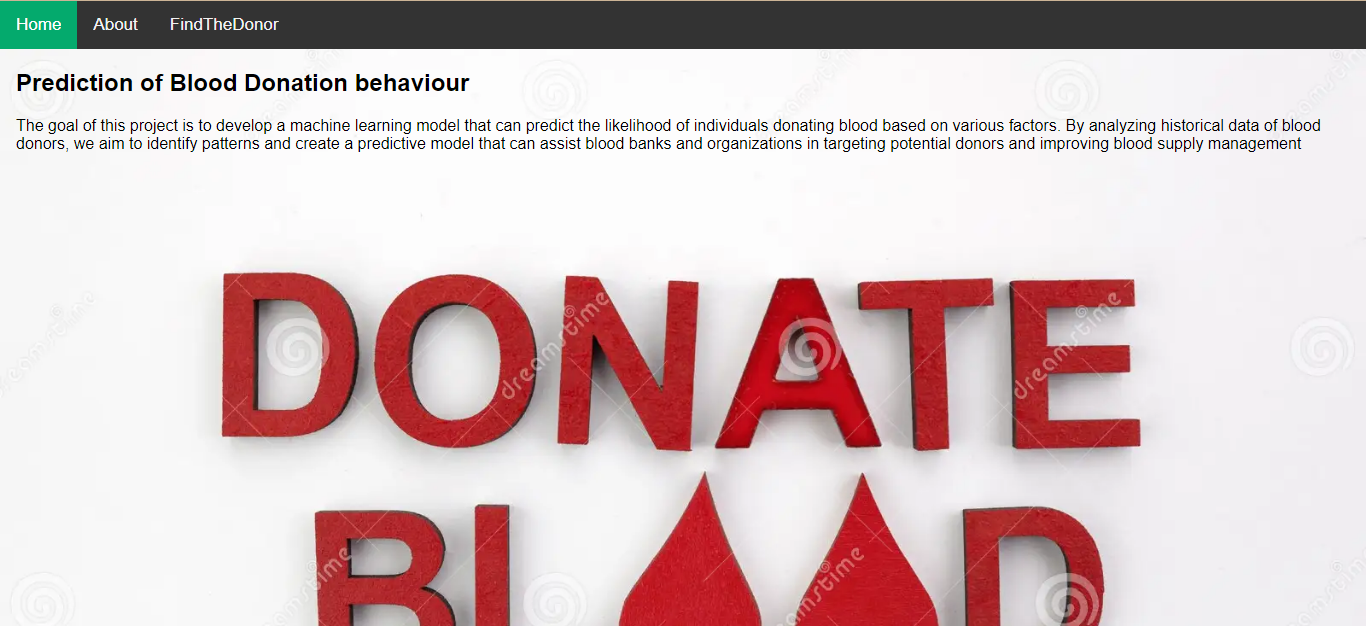
Main Function:



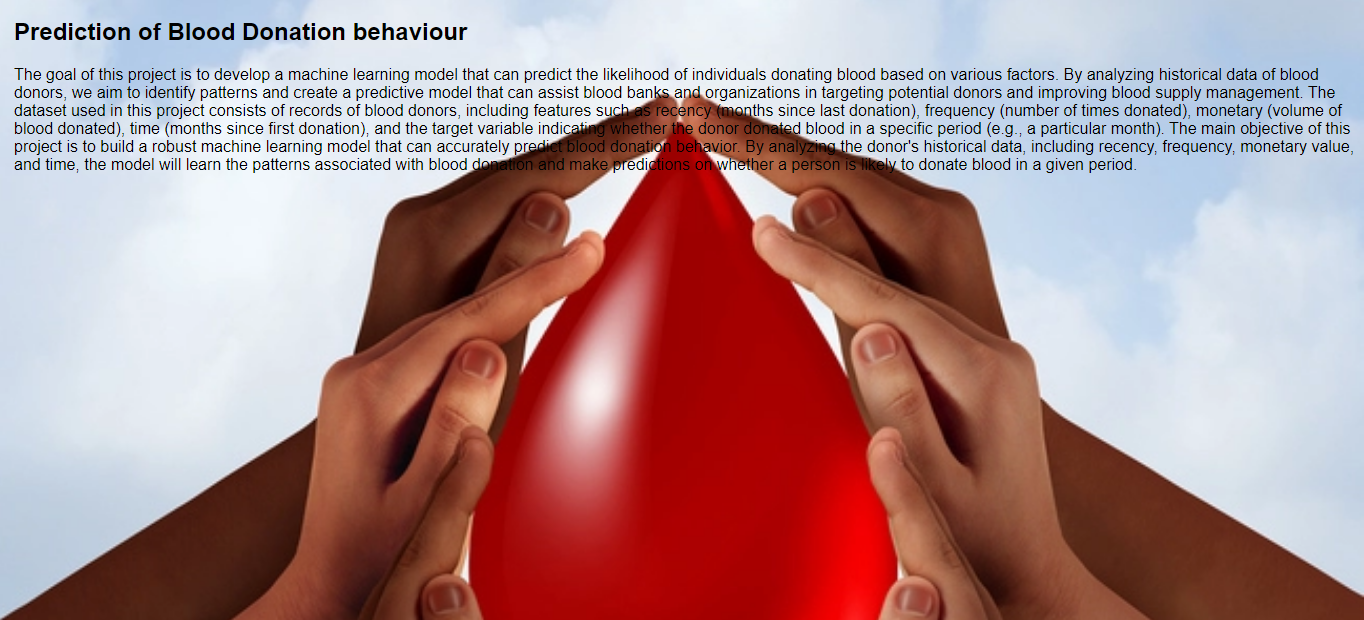
**Activity 3: Run the application**

Open Visual studio code and Import all the project folders.

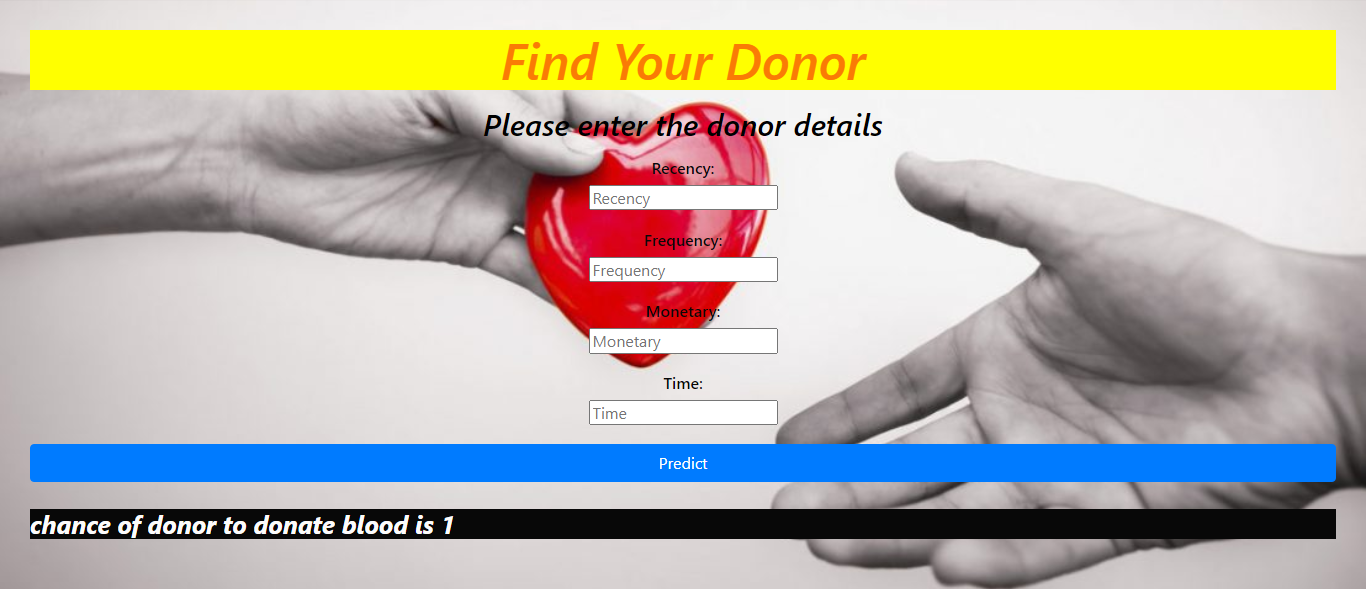
When you run the app.py file and click on the server url in terminal, you will redirected to home page. The home page will looks like:



If you click on About option, you will redirected to about.html page and it looks like:



Then click on FindTheDonor option from top left corner, you will redirected to findthedonor.html page. Enter the factors and click on predict button, the output looks like:



**Conclusion:**

This machine learning project aims to develop a predictive model that can assist blood banks and organizations in identifying potential blood donors. By accurately predicting blood donation behavior, the model can support targeted campaigns, optimize resource allocation, and contribute to more effective blood supply management.

The above description outlines a general approach to a blood donation ML project. The specific implementation and details may vary based on the available dataset, project scope, and requirements.