**CHRONIC KIDNEY DISEASE PREDICTION**

**Chronic Kidney Disease**, also called chronic kidney failure, describes the gradual loss of kidney function. Your kidneys filter wastes and excess fluids from your blood, which are then excreted in your urine. When chronic kidney disease reaches an advanced stage, dangerous levels of fluid, electrolytes and wastes can accumulate in your body.

So my project aim is to classify whether the patient is having Chronic Kidney Disease or not. The data set has 26 columns and 400 records. This data set is used to train the Machine Learning model. Thus the first step is   
  
**1. Preprocessing of data set:**

The ID column from the data set is removed and further the dataset's missing values are replaced by NaN. The categorical values or nominal values are converted into binary values for training. Then the NaNs are replaced with its respective columnar means. The resultant data set is converted to csv and stored. Using the resultant data set the correlation between the attributes is found to reduce or remove the attributes to get better accuracy. The highly correlated attributes are retained by dropping the other attributes that have low correlation. Also, further in this data set, the binary values are converted into its nominal values for uploading it in the Watson Auto AI model for prediction.

**2. IBM Cloud Account:**

I have created an IBM Cloud account and created a Cloud Object Storage service to store all my data set and model. Then I created Watson Studio instance and Machine Learning Service.

**3. Model Building:**

I have created a project in Watson Studio named Chronic Kidney Disease Prediction and in that I have created a Auto AI Experiment. Once created I imported the data set final\_modified\_ckd\_for\_prediction.csv into the Auto AI experiment and provided Class as prediction column and Ran the model. After training the data set, it showed RandomForestClassifier to have accuracy of 0.992. I then saved it as model and deployed it.

**4. Application Building:**

I again created a Node Red App instance in cloud to deploy my model in an application. Once created, I downloaded the node-red-dashboard from manage palette in Node Red service. Then I created the flow using form, functions, HTTP requests, text and debug. The form contains all the variable labels, names and their types. The PreToken function will have all the global variables, api key, headers and payload. The HTTP request has the URL token through which the UI is displayed. The Pre Prediction function will get all the values from the former HTTP request and sends it to the latter HTTP request function which has the scoring end point URL where the model was trained and the output gets displayed in the Debug messages. The Output function gets the output from Debug messages and sends it to the Prediction text and gets displayed.