# ALGORITHM

In all models we used **Decision Tree** algorithm and **Grid Search** for tuning the parameters like “max\_depth” and “min\_split” for getting better results.

# MODEL 1

In MODEL1 **Inputs** which will be used for training are

**-Order Created Time**

**-Location**

**-Total Orders to Assign**

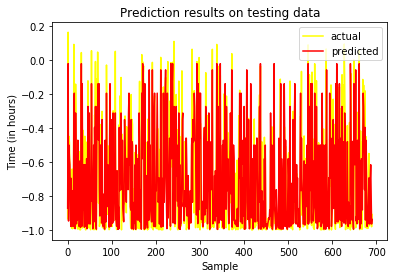
**-Quantity**

**Ouput** value should be

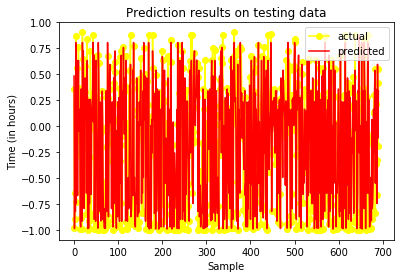
**-Order Delivery Time**

For the better accuracy of prediction, we convert time into **sine** and **cosine** and results in numbers and graph are below.

**Results of sine graph**



**Results of cosine graph**



MSE error is **0.05125897183684418**

R\_square is **0.7332288989702019**

Here is **Feature Importance** (It shows that, which inputs put more impact on algorithm, larger values means larger impact)

**'Qty': 0.0137901**

**'Location Code': 0.08292505**

**'Total Orders to Assign': 0.00845169**

**'hour\_sin': 0.11641946**

**'hour\_cos': 0.7784137**

These results are way better than our earlier results because of time conversion in top sine and cosine. Earlier we converted time into hours and that time **R\_square** was near to **0.23444**

First we train our model using these inputs and outputs and for **API**, you have to give **inputs** only, which I mentioned above, input will be in JSON format as example below

**JSON Format**

**{‘OrderCreatedTime’: 2018-08-12 19:05:22, ‘Location’:10234, ‘TotalPendingOrders’:4}**

By giving this input, our model will predict output( Order Delivery Time)

# MODEL 2

In MODEL2 **Inputs** which will be used for training which will be

**-Order Assign Time**

**-Location**

**-Total Orders to Assign**

**-Quantity**

**-Biker Code**

**-**0.7784137

**Ouput** value should be

**-Order Delivery Time**

First we train our model using these inputs and outputs and for **API**, you have to give **inputs** only, which I mentioned above, input will be in JSON format as example below

**JSON Format**

**{‘OrderCreatedTime’: 2018-08-12 19:05:22, ‘Location’:10234, ‘TotalPendingOrders’:4, ‘OrderAssignTime’:2018-08-12 19:15:19, ‘BikerName’:’Faizan’, ‘TotalBikerPendingOder’:2}**

By giving this input, our model will predict output( Order Delivery Time)

# MODEL 3

We are also suggesting you model3 for better prediction of **Order Delivery Time**, what will happen in model3, we will get same inputs as we are taking in model2 including **Biker Accepted Time**

**Inputs**

**-Order Created Time**

**-Location**

**-Total Pending Order (location wise)**

**-Order Assign Time**

**-Biker Name**

**-Total Biker Pending Order (biker wise)**

**- Biker Accepted Time**

**Ouput** value should be

**-Order Delivery Time**

**JSON Format for API**

**{‘OrderCreatedTime’: 2018-08-12 19:05:22, ‘Location’:10234, ‘TotalPendingOrders’:4, ‘OrderAssignTime’:2018-08-12 19:15:19, ‘BikerName’:’Faizan’, ‘TotalBikerPendingOder’:2, ‘BikerAcceptedTime’: 2018-08-12 19:25:10}**

By giving this input, our model will predict output( Order Delivery Time)

# MODEL 4

We are also suggesting you model4 for better prediction of **Order Delivery Time**, what will happen in model4, we will get same inputs as we are taking in model3 including **Biker In Time**

**Inputs**

**-Order Created Time**

**-Location**

**-Total Pending Order (location wise)**

**-Order Assign Time**

**-Biker Name**

**-Total Biker Pending Order (biker wise)**

**- Biker Accepted Time**

**- Biker In Time**

**Ouput** value should be

**-Order Delivery Time**

**JSON Format for API**

**{‘OrderCreatedTime’: 2018-08-12 19:05:22, ‘Location’:10234, ‘TotalPendingOrders’:4, ‘OrderAssignTime’:2018-08-12 19:15:19, ‘BikerName’:’Faizan’, ‘TotalBikerPendingOder’:2, ‘BikerAcceptedTime’: 2018-08-12 19:25:10, BikerInTime: 2018-08-12 19:30:58 }**

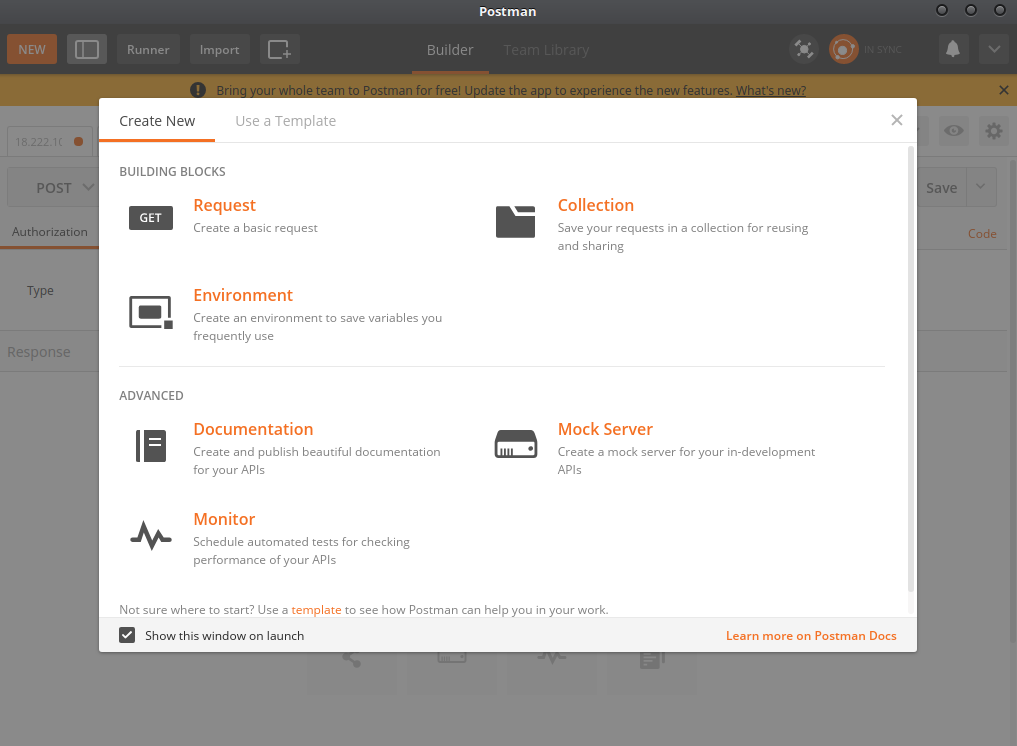
By giving this input, our model will predict output( Order Delivery Time)

# Testing the inputs using Postman

For testing this API we upload model to our server to test the results.

Now for testing these results you have to install postman first and after that follow the guide below.

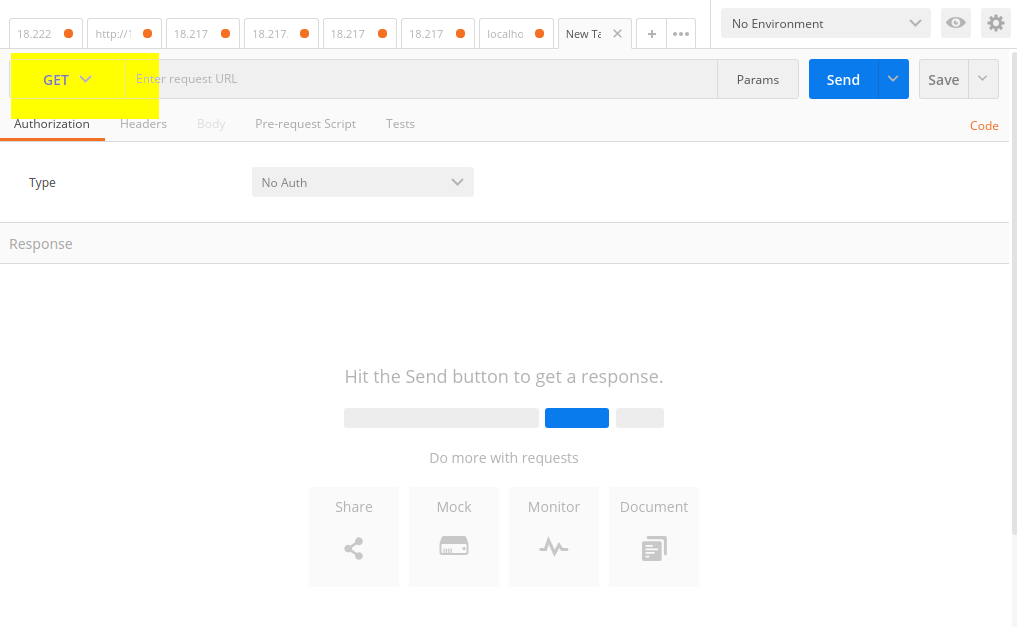
After completing the installation, open **postman,** you will see screen like below



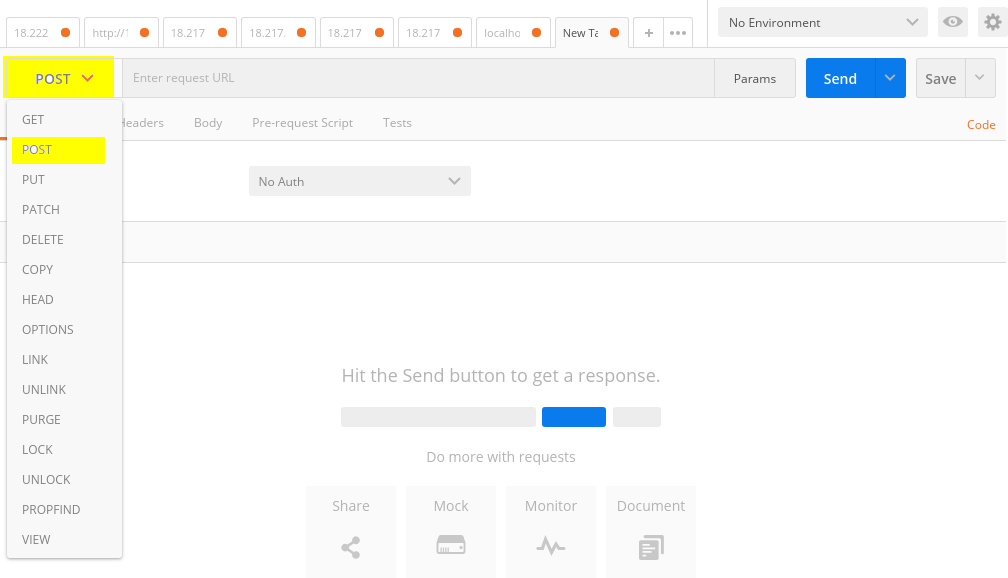
Close the above screen

After closing the above screen you will the screen, where you will write your inputs to get outputs

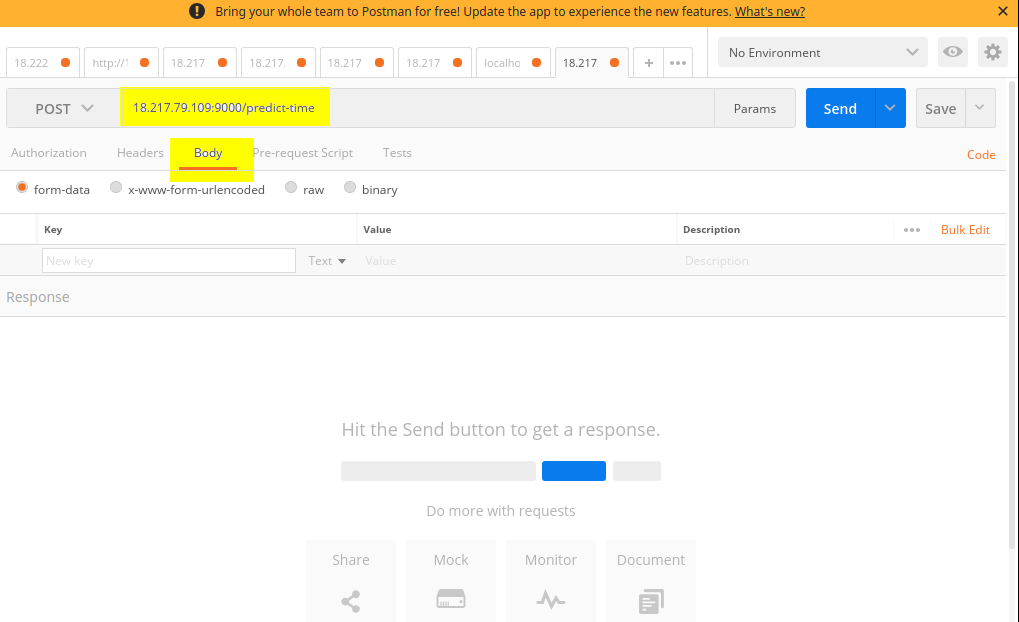
So first change method from **GET** to **POST**



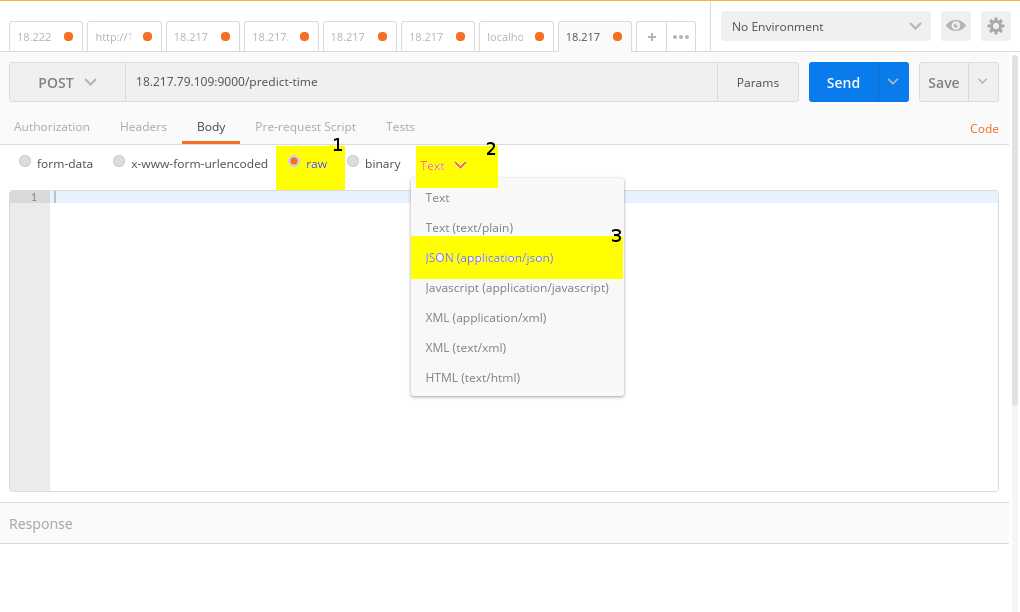
Click the place, where I highlighted and select **POST**



Now Give the server **URL (18.217.79.109:9000/predict-time)** and after that click on **Body** session

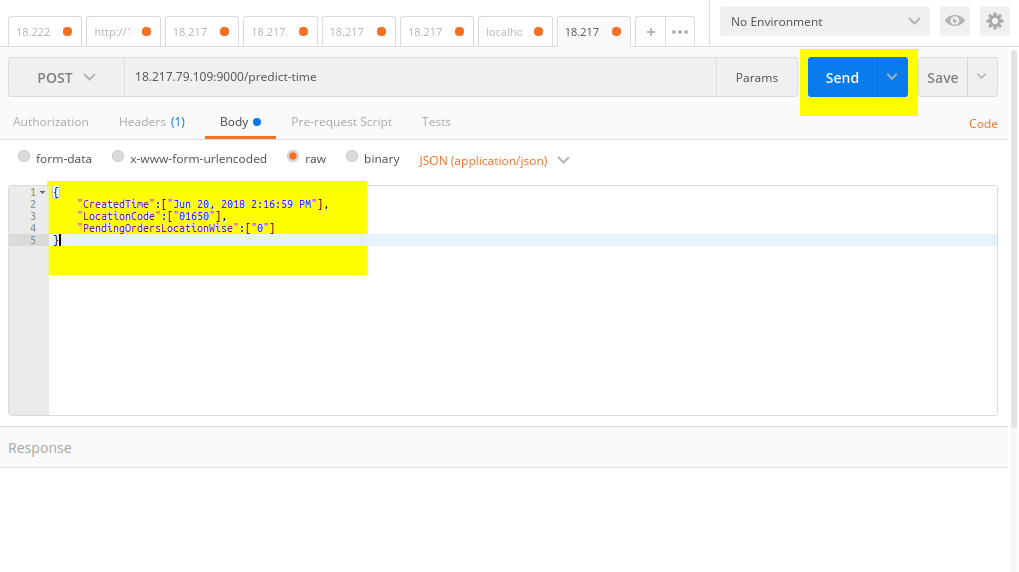


After clicking on Body session now select **raw** radio button and after that select input type in our case, our input type is **JSON**

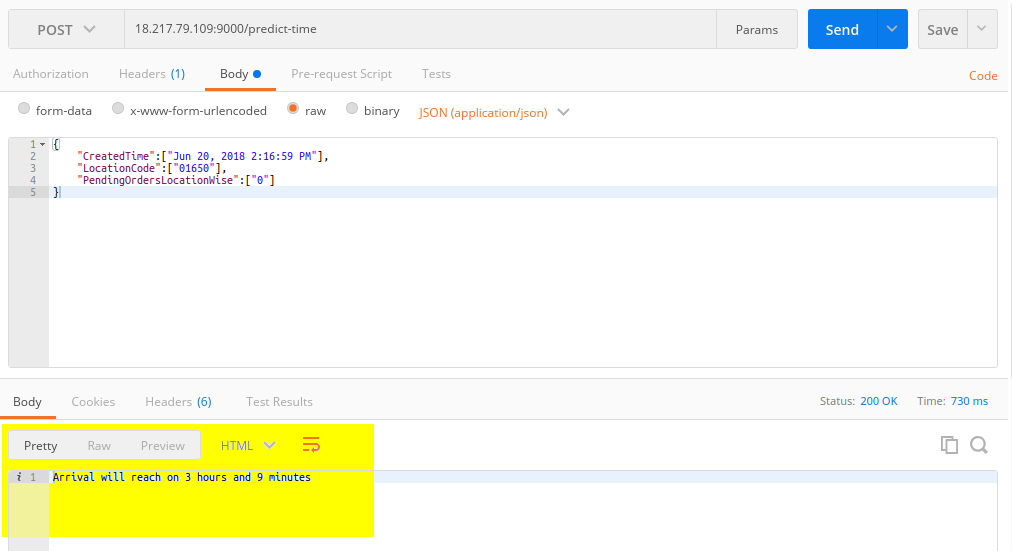


Now it time to write our input, Our inputs are vary according to our models for more information read Model1 , Model 2 and other model section above.

Let’s look at Input of model1 in screenshot below

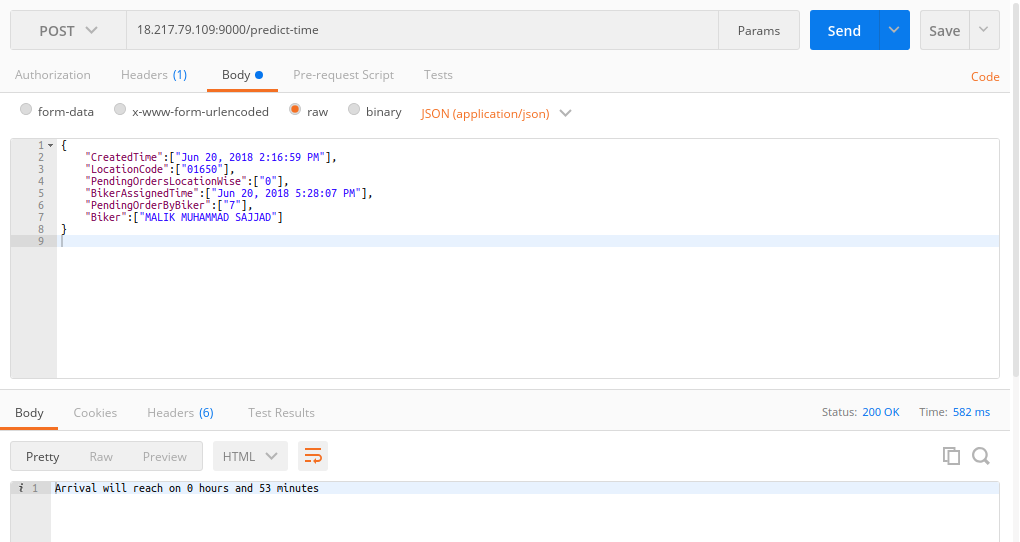


After giving inputs click on send and then you will see the output as below

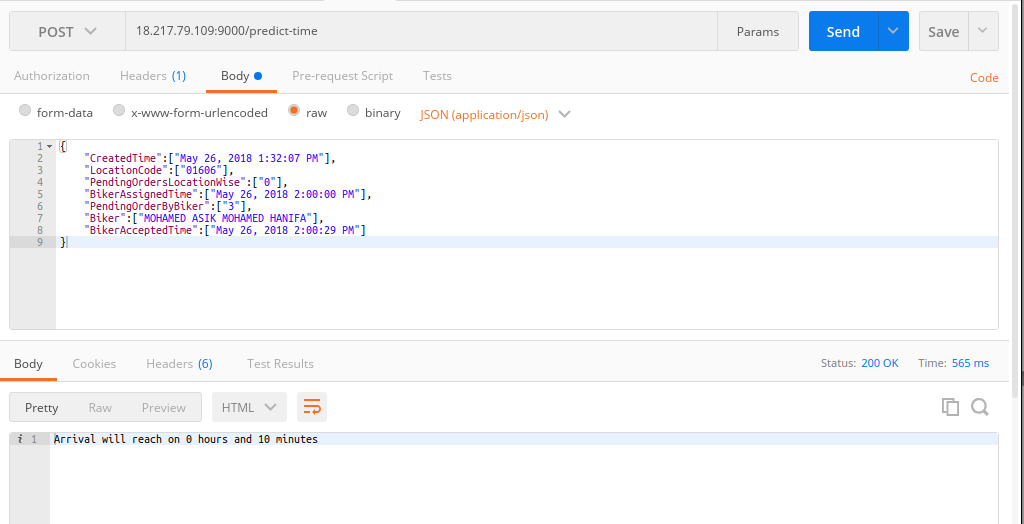


You can give other models input in similar way, for your easiness I am sharing screenshot of other inputs too.

Model2 Screenshot



Model3 Screenshot



Model4 Screenshot

