**Restaurant Take Home Case Study**

**Pre-requisite: To re-run this notebook one will only need libraries which are being imported to be installed on his/her system**

Please find below the approach to the solution against given problem.

1. Load the given data in 2 separate dataframes.
2. Perform basic analysis to check if there any missing data before merging dataframes
3. After merging the 2 dataframes into 1 based on restaurant\_id, I started with analysis which includes checking the presence of null values, checking the data type
4. Since there were no missing values, I proceeded with data type conversion. Otherwise null values needs to be handled
5. During data conversion, I thought about multiple derived features such as time\_of\_day to get better understanding that during which part of the day restaurant are receiving the max orders
6. Next step was to perform univariate analysis to understand the data better. Thus plotted different graphs to get better understanding
7. Next step was to perform bi-variate analysis to understand the relation between 2 features
8. As for the next step I performed outlier detection and handled them with mean. Although there is a possibility that the outliers might be coming due to some other reasons about which better understanding can be acquired post discussing with business people.
9. Perfomed data encoding using Label encoders for ordinal features and BaseNEncoder for others. Didn’t used one hot encoding since the number of distinct values was very high
10. Once the data was free from issue, I plotted heatmap to get better understanding of correlation
11. Next step was to build ML model, thus instead of trying multiple algorithms which could have been done if more time was available, I used XGBoostRegressor to build ML model.
12. Performed data split into 70:30 ration between train & test.
13. Also used sequential feature selector to identify the prominent features.
14. Used GridSearchCV to find the best hyper-parameters
15. Used XGboot model with those parameters to perform training and prediction
16. Used root\_mean\_squared\_log\_error to check model performance. It came at 0.46 which can be treated as baseline model and further improvements can be done.

**Other things that can be implemented if more time was given:**

* Adding order\_queue\_number to check how many orders are currently listed for a particular restaurant on a given day and with 40-45 mins interval
* Performing statistical analysis using methodologies like hypo-thesis testing, Chi-squared, Annova to indentify prominent features
* Using different ML algorithms
* Performing hyper-parameter tuning upto better extent using Optuna
* Handling categorical variables in a better way