# Data Scientist
# Research Assignment

1. The attached dataset has already been cleaned because the features are anonymized and it would be difficult for you to clean the dataset without understanding the features. Describe the types of things you would have looked at when cleaning the dataset if you knew what the features meant and how they related to each other, i.e. discuss the kinds of considerations you would make before cleaning a new dataset. What would you look for and how would you handle it?

## # Data Cleaning Steps:

Though the data is cleaned, but below are the steps which I would have followed to clean the data and identify issues

1. Check for nulls in the data. Would have written a PySpark script which will give us, what columns contain null values and how much proportion of nulls they have
2. Depending on the proportion, would have either deleted the column or replaced null with the mean value (in case of continuous and discrete value) or replaced with 0 in case of binary
3. Identifying and replacing the null value should proceed with caution since it will be dependent on the type of column
4. The next step would be to identify if we have any duplicates in our data, again a pyspark script which helps in identifying duplicate rows as well as removes it
5. In case of categorical/text data, converting the data into numeric, so that it can be used for analysis (using one hot encoder)
6. Renaming of columns, each column should have a meaningful names so that it becomes easy to understrand
7. Standardizing data → eg: if there are two columns merged into one then we have to separate such columns
8. Converting data types into their respective format eg: string to int or vice versa

2. Using the attached dataset, choose an algorithm (or set of algorithms) to predict the **label** column. Detail your process for selecting the algorithms and your final f-score. Please share all your associated code and your pickled model along with the environment state so we can check your work, we strongly suggest using a virtual environment or docker container to work on this assignment.

## # Final Analysis

1. The F1 score of the model is 0.86 and the AUC is 0.88

2. F1 Score is the weighted average of Precision and Recall. In case of uneven class distribution like in our dataset, F1 score helps in evaluating the model

3. After tuning the parameters using train-validation split, the overall model F1 score came out to be 0.84. Thus we can say that the initial model parameters are the best and the score is valid

4. The confusion matrix results shows that, our model predicted total 64 correct positives ( keeping in mind that this percentage is very high). It has fewer False Positives but sufficiently high False Negatives

5. F1-score could have been better if Linear SVM was used, but it ran continuously and eventually had to kill it. But I have written the code which will evaluate the performance

6. Multilayer Perceptron Classifier also like SVM ran for couple of hours and had to kill it eventually. Again, I have attached the code used </b>

7. If my AWS server was more powerful, I would have used Python instead of Pandas and would have leveraged even more powerful algorithms with better feature selection like Anova, RFE and models like XGBoost and even ANN (Keras)

Algorithms used: **Logistic Regression**
Algorithms to be used if I had more time and bigger server: **LinearSVM, MultiLayer Perceptron Classifier**

**Environment: Apache Pyspark**

3. Typically after presenting a model to stakeholders they will want to know which features were most predictive of the target column. Detail which features were the most predictive and your process of determining them. Note that feature importance does not detail the directionality of the feature, discuss ideas on what you could do to determine directionality of the top features.

## # Feature Selection Procedure:

Out of the 501 features, an important task was to identify the important features, below are the steps I took:

1. The dataset has both binary and continuous variables. Separate these columns into a different dataframe

2. The result was, there were 335 binary columns and 166 continuous variables

3. Using Chi square selector on the binary column dataframe, identified the top 50 features in that dataframe and stored them

4. Created a logistic regression model on those column and got rid of those variables whose coefficients were zero

5. Majority of the features had a negative coefficients and it means that, they impact the output variable positively when they decrease or in our case, they are zero (masked as zero in case of categorical)

6. For continuous/discrete variables, I created a logistic regression model and got their coefficients

7. Out of 166 columns, 24 had zero coefficients and thus were ruled out from analysis

8. Many variables had very low coefficients (in negative exponential terms) and thus I decided to rule them out

9. Finally out of 166, I identified 39 columns which were impactful for the output probability

10. 27 were those which impacted the output probability inversely. Their decrease was increasing the probability

11. Since it's PySpark, I could not use Anova or RFE which would have further helped in identifying important features or would have gotten rid of more variables

## Important Features are:

1. Features/variables **"f445, f458, f301, f309, f265, f253, f325, f183, f451, f275"** are the top variables which impact/increase the output probability (of 1) when they are 0

2. The other categorical/binary variables which increase the probability of the output to be 1 when they are themselves 1 are: **"f478, f228, f173, f470, f208, f113, f163"**

3. **f12, f68, f18, f65, f88, f403, f73, f63, f67, f35, f158, f313"** are the top 12 continuous/discrete variables by which probability that the event identified by the dependent variable happens decreases as the value of the independent variable increases

4. **"f366, f6, f489, f115, f122, f330, f272, f369, f488, f491, f483, f258"** are top 12 continuous/discrete variables which impact output probability positively when they increase