**Data Warehouse and Mining project**

**Introduction**

Modern police work often relies on analytical judgement and early advising of intelligence information can prove to be very beneficial in providing clues for various criminal offenses. Public security agencies are responsible for extracting and compiling the useful information from the massive internet resources and corelating the data to criminal offences, thus allowing them to make accurate predictions of future criminal activities. Bicycle theft is a reoccurring issue taking place all around Toronto from sidewalks to private property and this aim of this project is to analyze the relationship between bicycle theft reports and open data obtained from the Internet. From there, the data will be explored, described, and modeled, allowing us to further explore the relationship between intelligence acquisition and modern police security work and promote the transition to intelligence-led active policing.

**Overview**

The data has been collected from the Toronto Police website. For our project, we used the python code and different kinds of bundles. Firstly, Panda is imported for pursuing from a CSV document for controlling further use. In addition, Numpy is used to change over information into an organization appropriate to encourage models. Also, Seaborn and matplotlib are used for perceptions.

Data has been loaded to calculate means, average, and correlate the data among elements. Scatterplot has been created for making different kinds of graphs like histogram, box plot, heatmaps, line graphs. For the creation of data modelling, a dummy dataframe has been created for normalization and standardization of data. Among Linear and Logistic Regression, we used Logistic Regression for calculation from sci-kit learn. In this project, decision tree has been used to build the predictive model. Confusion matrices have been developed to score and evaluate the model. For deploying in the API, it has been used the Flask framework. However, Our API has also been tested by using Postman.