AI-DRIVEN EXPLORATION AND PREDICTION OF COMPANY REGISTRATION TRENDS WITH REGISTRAR OF COMPANIES (ROC)

PHASE :3

DEVOLPMENT PHASE PART 1

PROCESS AND DATA:

1. ***Data Collection**: Gather historical stock prices, trading volumes, and relevant financial data. Sources could include financial databases, APIs, or web scraping tools.*
2. ***Data Pre-processing**: Clean the data, handle missing values, and perform feature engineering. This step might involve normalization, scaling, or transforming the data to make it suitable for the chosen model.*
3. ***Feature Selection:** Choose relevant features that might affect stock prices, such as historical prices, trading volumes, news sentiment, economic indicators, and company-specific information.*
4. ***Model Selection:** Select an appropriate machine learning algorithm such as linear regression, decision trees, random forests, or deep learning models like recurrent neural networks (RNNs) or long short-term memory networks (LSTMs).*
5. ***Training the Model:** Use a portion of the data to train the model, adjusting parameters and hyper parameters to optimize performance.*
6. ***Model Evaluation:** Assess the model's performance using metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), or Root Mean Squared Error (RMSE) on a validation dataset.*
7. ***Testing the Model:** Apply the trained model to a separate test dataset to evaluate its predictive power.*
8. ***Iterate and Refine:** Fine-tune the model by iterating on the pre-processing steps, feature selection, and model selection to improve predictive accuracy.*

USED SENSORS:

Predicting stock prices using sensor data is an emerging application of data science, combining the fields of finance, data analysis, and sensor technology. By leveraging sensor data, such as market sentiment, economic indicators, or even physical sensors monitoring relevant parameters, data scientists can develop more nuanced and accurate models for stock price forecasting.

To achieve this, the process typically involves collecting and analysing a diverse range of data, applying machine learning algorithms, and employing techniques like time series analysis to identify patterns and correlations. However, it's important to note that stock market prediction, despite technological advancements, remains inherently challenging due to its dynamic nature and the influence of various external factors.

Program:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
from IPython.display import HTML
from sklearn.preprocessing import LabelEncoder
```

```
#Get the data
df = pd.read_csv('/Data_Gov_Tamil_Nadu.csv')
df.info()
```

```
#check Duplicate Data
print(df.duplicated().sum())
```

```
#date and time is converted object in to datetime64
df['DATE_OF_REGISTRATION'] =
pd.to_datetime(df['DATE_OF_REGISTRATION'])
df['day_of_week'] = df['DATE_OF_REGISTRATION'].dt.day_name()
df['month'] = df['DATE_OF_REGISTRATION'].dt.month
print(df['DATE_OF_REGISTRATION'])

#checking the missing values
print(df.isnull().sum())
```

```
# fill missing values in column with their most repeated
df['COMPANY_CLASS'].fillna(df['COMPANY_CLASS'].mode()[0],
inplace=True)
df['COMPANY_CATEGORY'].fillna(df['COMPANY_CATEGORY'].mode()[0]
, inplace=True)
df['COMPANY_SUB_CATEGORY'].fillna(df['COMPANY_SUB_CATEGORY'].m
ode()[0], inplace=True)
df['DATE_OF_REGISTRATION'].fillna(df['DATE_OF_REGISTRATION'].m
ode()[0], inplace=True)
df['INDUSTRIAL_CLASS'].fillna(df['INDUSTRIAL_CLASS'].mode()[0]
, inplace=True)
df['REGISTERED_OFFICE_ADDRESS'].fillna(df['REGISTERED_OFFICE_A
DDRESS'].mode()[0], inplace=True)
df['REGISTRAR_OF_COMPANIES'].fillna(df['REGISTRAR_OF_COMPANIES
'].mode()[0], inplace=True)
df['EMAIL_ADDR'].fillna(df['EMAIL_ADDR'].mode()[0],
inplace=True)
df['LATEST_YEAR_ANNUAL_RETURN'].fillna(df['LATEST_YEAR_ANNUAL_
RETURN'].mode()[0], inplace=True)
df['LATEST_YEAR_FINANCIAL_STATEMENT'].fillna(df['LATEST_YEAR_F
INANCIAL_STATEMENT'].mode()[0], inplace=True)
```

```
# interpolate missing values using linear interpolation
df['COMPANY_CLASS'].interpolate(inplace=True)
df['COMPANY_CATEGORY'].interpolate(inplace=True)
df['COMPANY_SUB_CATEGORY'].interpolate(inplace=True)
df['INDUSTRIAL_CLASS'].interpolate(inplace=True)
df['REGISTERED_OFFICE_ADDRESS'].interpolate(inplace=True)
df['REGISTRAR_OF_COMPANIES'].interpolate(inplace=True)
df['EMAIL_ADDR'].interpolate(inplace=True)
df['LATEST_YEAR_ANNUAL_RETURN'].interpolate(inplace=True)
df['LATEST_YEAR_FINANCIAL_STATEMENT'].interpolate(inplace=True
)
```

```
# standardize the numerical data
encoder = LabelEncoder()
df['COMPANY_CLASS'] =
encoder.fit_transform(df['COMPANY_CLASS'])
df['COMPANY_CLASS'] =
encoder.fit_transform(df['COMPANY_CLASS'])
print(df['COMPANY_CLASS'].to_string())
```