

Phase 3

Development Part 1

In this part you will begin building your project by loading and pre-processing the dataset

Import necessary libraries with dataset:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
data = pd.read_csv("E:\IBM\Sales.csv")
```

Exploring data analysis:

Explore the dataset to get an understanding of its structure and characteristics.

```
print(data.head())
print(data.info())
print(data.describe())
```

Data pre-processing:

Data pre-processing is a critical step that involves handling missing values, encoding categorical variables, and scaling/normalizing numerical features.

```
data = data.dropna()
data = pd.get_dummies(data, columns=['categorical_column'])
```

Splitting the dataset:

It involves machine learning, split your dataset into training and testing sets. This is crucial for model evaluation.

```
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
data['numerical_column'] = scaler.fit_transform(data['numerical_column'].values.reshape(-1, 1))
X = data.drop('target_column', axis=1)
y = data['target_column']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Building and training the models:

Depending on the project, we might build and train machine learning models using libraries like "**Scikit-Learn or deep learning frameworks like TensorFlow or PyTorch**".

```
from sklearn.linear_model import LogisticRegression
model = LogisticRegression()
model.fit(X_train, y_train)
from sklearn.metrics import accuracy_score
y_pred = model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
```

After training the model, evaluate its performance using appropriate metrics

Output:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
data = pd.read_csv("E:\IBM\Sales.csv")
print(data.head())
print(data.info())
print(data.describe())
data = data.dropna()
data = pd.get_dummies(data, columns=['categorical_column'])
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
data['numerical_column'] = scaler.fit_transform(data['numerical_column'].values.reshape(-1, 1))
X = data.drop('target_column', axis=1)
```

phase-3 x

1	Radio	200 non-null	float64
2	Newspaper	200 non-null	float64
3	Sales	200 non-null	float64

dtypes: float64(4)
memory usage: 6.4 KB
None

	TV	Radio	Newspaper	Sales
count	200.000000	200.000000	200.000000	200.000000

Run Python Packages Python Console Problems Terminal Services

pre-built shared indexes: Reduce the indexing time and CPU load with pre-built Python packages shared indexes // Always download // Download once // Don't show again // Config... (55 minutes) 23:1 CRLF

Team Leader: SALMAN AHMED S M I

Team Member: MOHAMMED SUFIYAN S

Team Member: MOHAMMED SAAD A

Team Member: MOHAMMED SAAD G

Team Member: MOHAMED RASEEF RAYYAAN I