**Python Coding Challenge**

**Name: Vikas Reddy Gorantla**

**Date: 15-11-2024**

1.Printing rows of the Data

import pandas as pd

data = pd.read\_csv('Downloads/annual-enterprise-survey-2023-financial-year-provisional.csv')

print(data)

A screenshot of a computer

Description automatically generated

* pd.read\_csv(): This function loads data from a CSV file into a pandas DataFrame.
* print(data): Displays the entire DataFrame in the output.

2.Printing the column names of the DataFrame

data = pd.read\_csv('Downloads/annual-enterprise-survey-2023-financial-year-provisional.csv')

print(data.columns)

A screenshot of a computer

Description automatically generated

* columns gives the column names in the DataFrame.

3.Summary of Data Frame

data = pd.read\_csv('Downloads/annual-enterprise-survey-2023-financial-year-provisional.csv')

data.info()

A screenshot of a computer code

Description automatically generated

* info() provides a concise summary of the DataFrame, including the number of non-null values and the data types of each column.

4.Descriptive Statistical Measures of a DataFrame

data = pd.read\_csv('Downloads/annual-enterprise-survey-2023-financial-year-provisional.csv')

data.describe()

A screenshot of a cell phone

Description automatically generated

* describe() computes summary statistics such as mean, std, min, 25%, 50%, 75%, and max for numerical columns.

5.Missing Data Handing

data = pd.read\_csv('Downloads/annual-enterprise-survey-2023-financial-year-provisional.csv')

data.dropna()

A screenshot of a computer

Description automatically generated

* **dropna()**: By default, it removes any row that has at least one NaN value. It returns a new DataFrame with the rows containing missing data removed.

6.Sorting DataFrame values

data = pd.read\_csv('Downloads/annual-enterprise-survey-2023-financial-year-provisional.csv')

data.sort\_values(by = 'Year',ascending = True).head

A screenshot of a computer

Description automatically generated

* sort\_values() sorts the DataFrame based on one or more columns. You can control the sorting order using ascending=True for ascending or False for descending.

7.Merge Data Frames

d1 = pd.read\_csv('Downloads/annual-enterprise-survey-2023-financial-year-provisional.csv')

d2 = pd.read\_csv('Downloads/annual-enterprise-survey-2023-financial-year-provisional.csv')

d = pd.merge(d1, d2)

print(d)

A screenshot of a number of numbers

Description automatically generated

* merge() combines two DataFrames based on a common column (on='common\_column'). You can choose the type of merge ('inner', 'outer', 'left', 'right').

8.Apply Function

d2 = pd.read\_csv('Downloads/annual-enterprise-survey-2023-financial-year-provisional.csv')

def fun(x):

if x > 2010:

return "Profit Year"

else:

return "Loss Year"

data['newcolumn'] = data['Year'].apply(fun)

data

A screenshot of a computer

Description automatically generated

* apply() applies a function to each element in a Series or DataFrame.

9.What is the number of columns in the dataset?

num\_columns = data.shape[1]

print(f"The number of columns in the dataset is: {num\_columns}")



* shape[1] gives the number of columns.

10.How is the dataset indexed?

print(data.index)



* index returns the index (row labels) of the DataFrame. By default, it is a RangeIndex starting from 0.

11.What is the number of observations in the dataset?

num\_rows = data.shape[0]

print(f"The number of observations (rows) in the dataset is: {num\_rows}")



* shape[0] gives the number of rows (observations) in the DataFrame.

12.Visualizing DataFrame

data.plot(x='Value', y='Year', kind= 'hist')

A screenshot of a graph

Description automatically generated

* **Histogram (kind='hist')**: Used to visualize the distribution of a single numeric column. It groups the data into bins and shows how many data points fall into each bin.