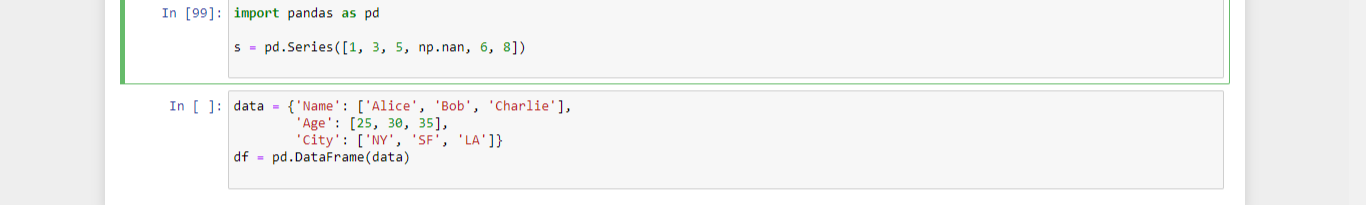
**NAME :- ANIKET SANJAYKUMAR BIYANI**

**DATA ENGINEERING BATCH -1**

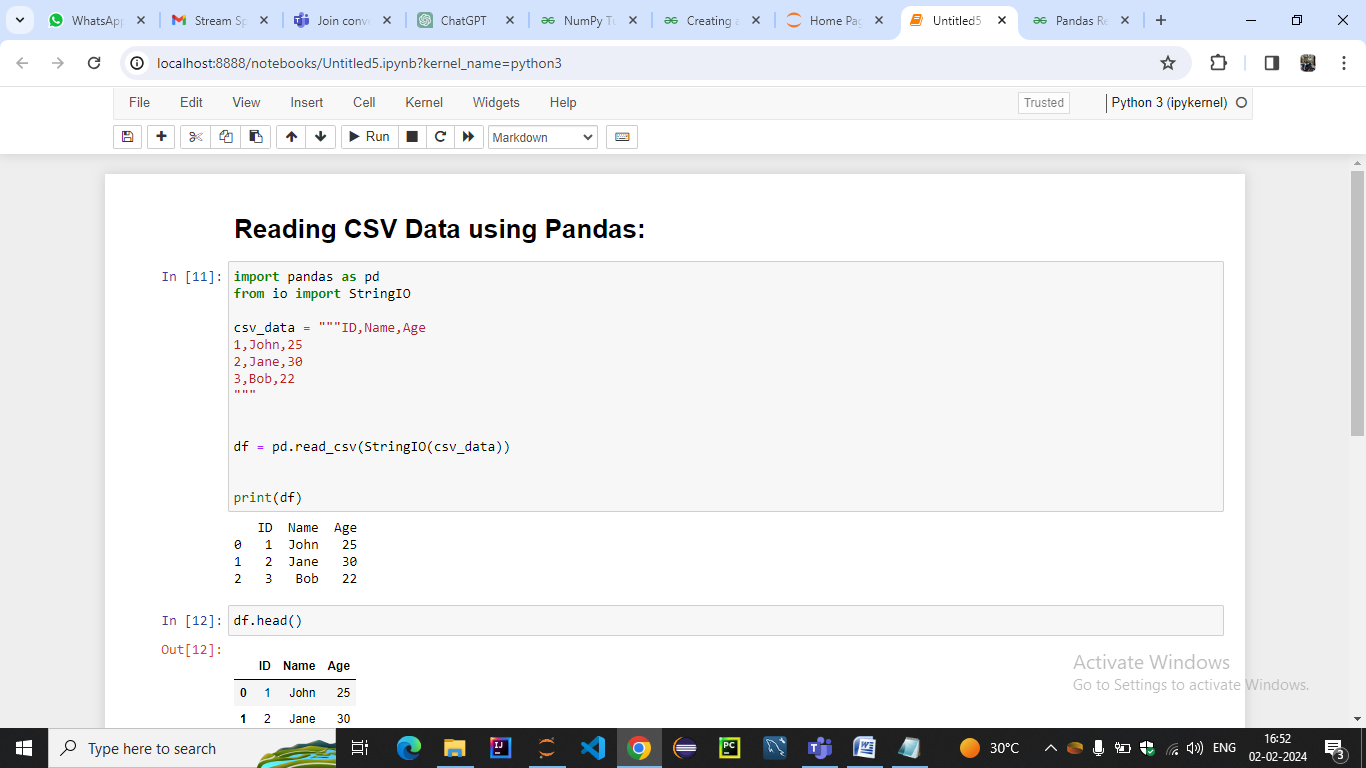
**PYTHON CODING ASSESMENT**

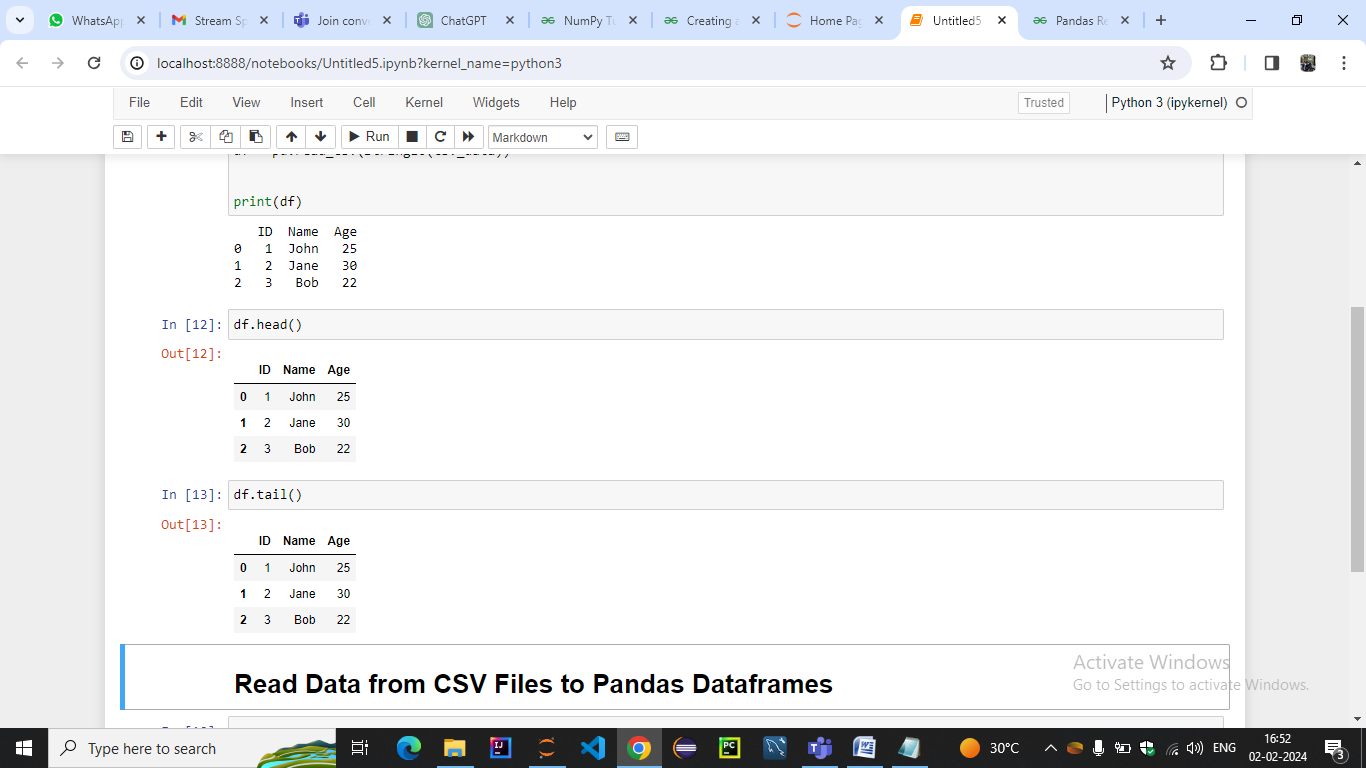
1. Explain Pandas for Data Processing & execute Reading CSV Data using Pandas & Read Data from CSV Files to Pandas Dataframes &Filter Data in Pandas Dataframe using query.

* **Pandas for Data Processing:**
* Pandas is an open-source data manipulation and analysis library for Python.
* It provides easy-to-use data structures like Series and DataFrame, making it efficient to work with structured data.
* Key features include data cleaning, reshaping, merging, and analysis.
* Pandas Data Structures:
* Series:
* A one-dimensional labeled array.
* Created using pd.Series(data, index).
* Supports various data types, including numeric, string, datetime, etc.
* DataFrame:
* A two-dimensional table with labeled rows and columns.
* Created using pd.DataFrame(data, columns, index).
* Created from dictionaries, lists of lists, NumPy arrays, or other DataFrames

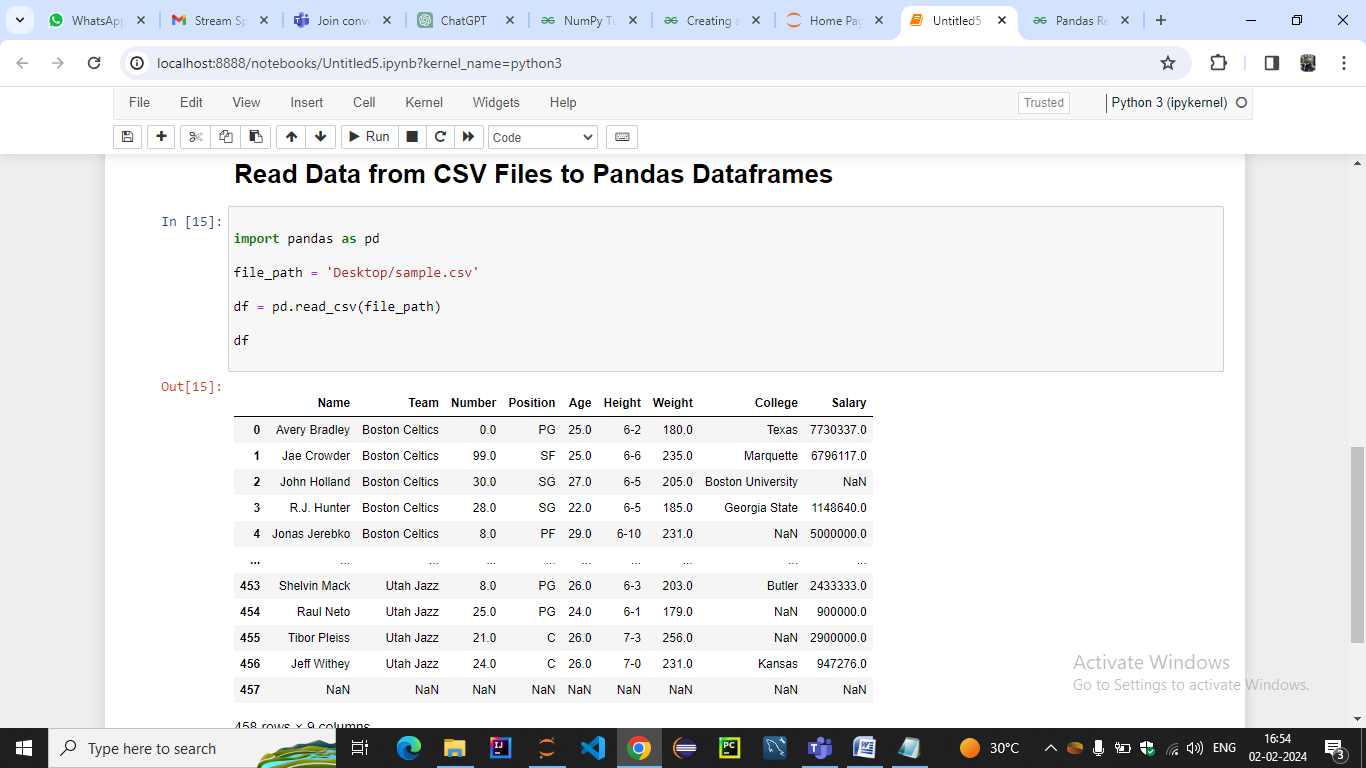


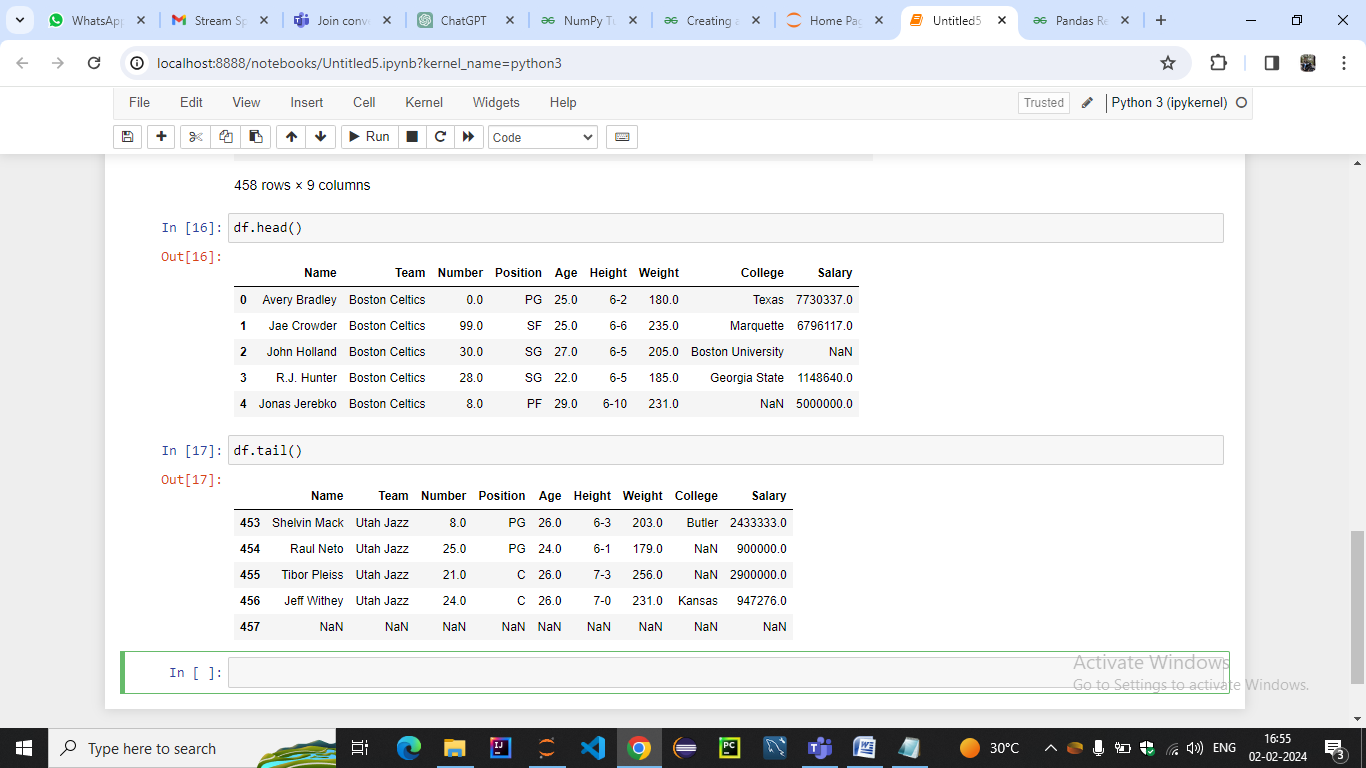
* Pandas is an essential tool in the data science and analysis toolkit, offering a flexible and efficient framework for handling and analyzing structured data.
* Its ease of use and wide range of functionalities make it a go-to library for data preprocessing, exploration, and manipulation in Python.
* Pandas is a Python library for data manipulation and analysis, providing versatile data structures like DataFrames.
* It excels in reading, filtering, and transforming data from various sources, such as CSV files. With powerful querying capabilities, it enables efficient data exploration.
* Pandas facilitates tasks like aggregation, sorting, and joining, offering a comprehensive toolkit for data processing. Additionally, it supports seamless data export to various formats, including CSV and JSON.
* Reading CSV Data using Pandas
* CSV (Comma-Separated Values) is a popular format for storing tabular data.
* Pandas provides the read\_csv() function, which simplifies the process of reading data from a CSV file and creating a DataFrame.
* The function is versatile and can handle various parameters to accommodate different CSV file structures.
* Reading CSV Data using Pandas: Implies the process of reading data from a CSV-formatted source into a Pandas DataFrame





* Read Data from CSV Files to Pandas Dataframes

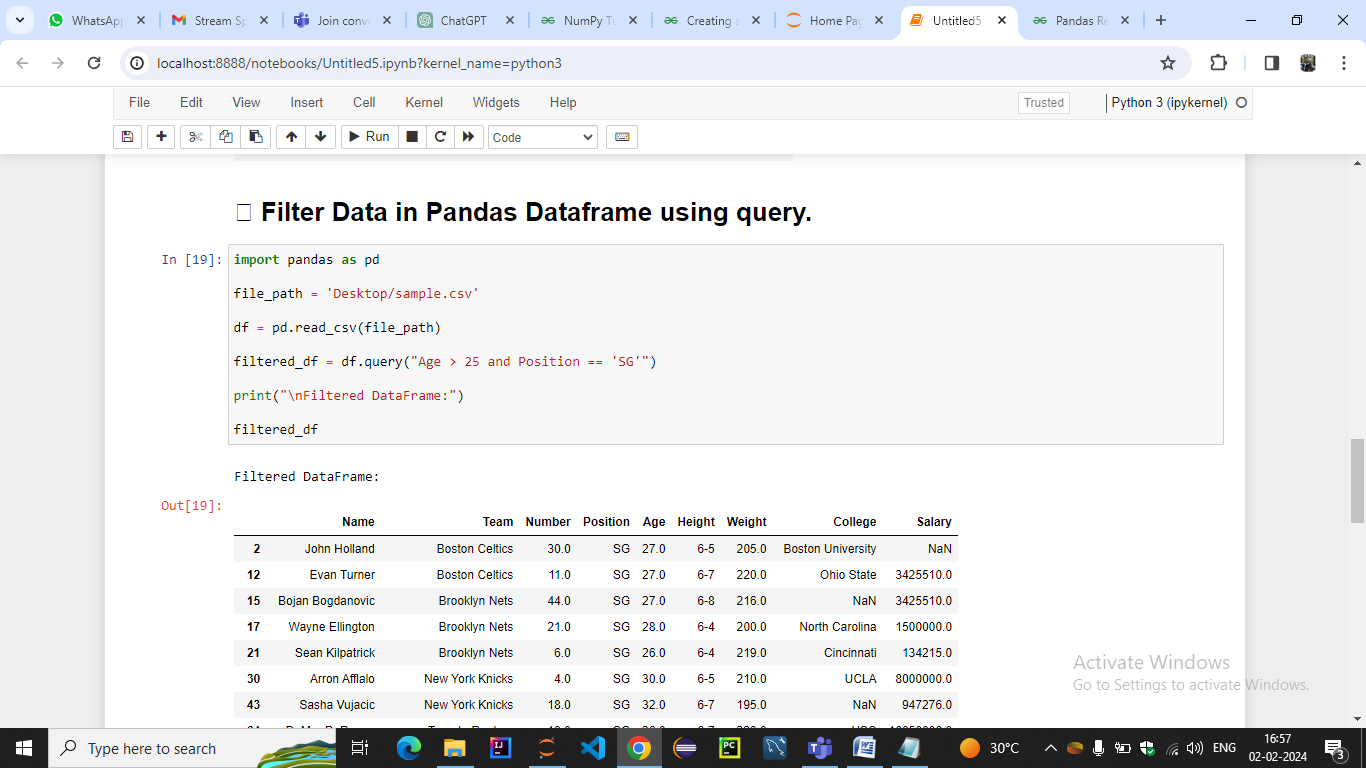


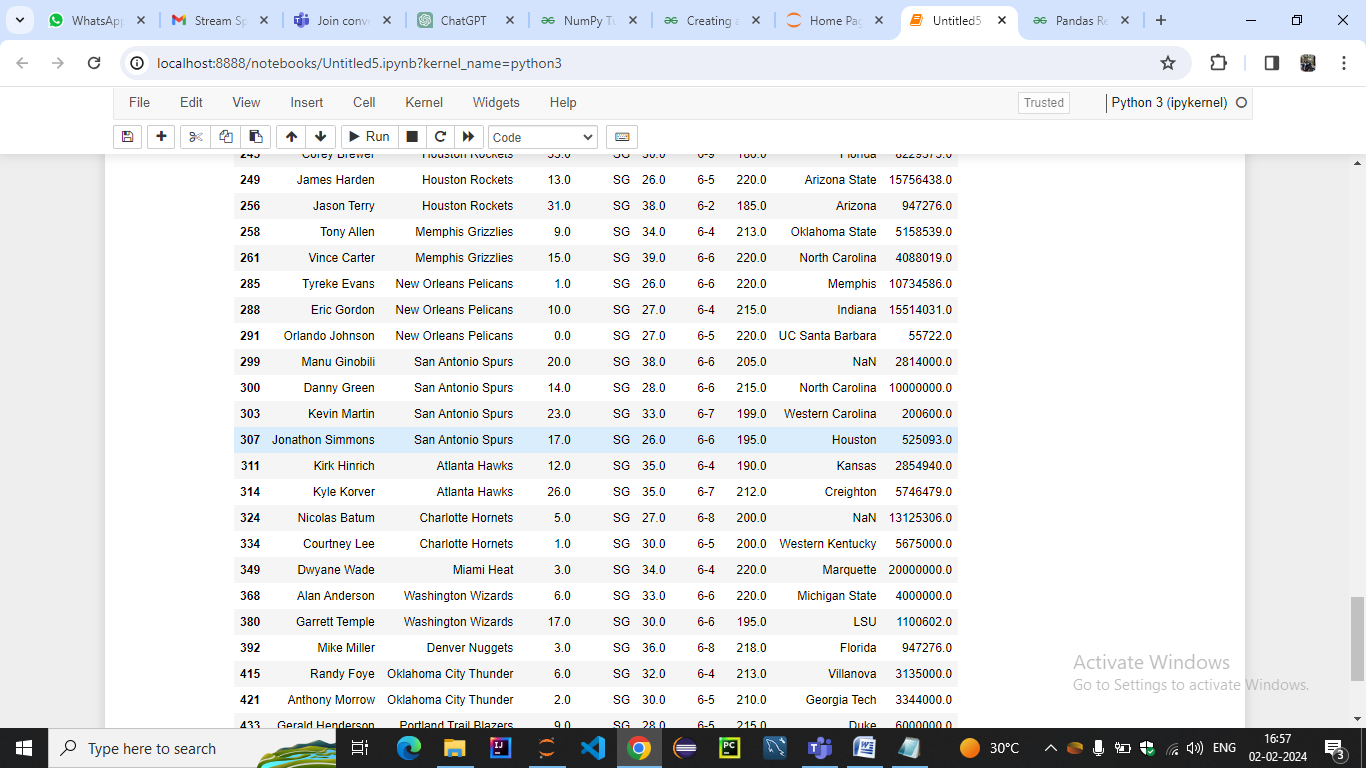


* Filter Data in Pandas Dataframe using query.
* Pandas provides the query method, which allows you to filter data in a DataFrame based on a specified query expression.
* This method provides a convenient and expressive way to select rows from a DataFrame that meet certain conditions.
* The syntax is similar to SQL, making it easy to read and write

filtered\_df = original\_df.query("condition\_1 and condition\_2")

* The query string can include various comparison and logical operators, such as == (equal), != (not equal), < (less than), > (greater than), <= (less than or equal), >= (greater than or equal), and, or, and not.Column Names:
* Column names in the query string should be directly used without quotes. If a column name contains spaces, you can use backticks to enclose it





This will result in a DataFrame containing only the rows where the age is greater than 25, and the position is 'SG'

