**Report: Exe. Mathematical Modeling and Simulations (MAL7080)**

**Submitted By: Vikash Yadav (M22AI661)**

**Title: RelationalTable - Python Class for Handling CSV Data**

**Introduction:**

The RelationalTable class is a Python class designed to handle CSV data in a relational table format. It provides methods for reading CSV files, querying data by columns and rows, and establishing relationships between tables based on column values. The class is designed to be easy to use and versatile for handling CSV data in a convenient and efficient manner.

**Purpose:**

The purpose of the RelationalTable class is to provide a convenient and efficient way to work with CSV data in a relational table format. It aims to simplify tasks such as reading CSV files, querying data by columns and rows, and establishing relationships between tables based on column values. The class is intended to be used in scenarios where CSV data needs to be processed and analyzed in a relational manner, such as in data manipulation, data analysis, and data integration tasks.

**Features:**

The RelationalTable class provides the following features:

**Reading CSV data:** The class has a read\_csv() method that can read CSV data from a file and store it in a defaultdict(list) object. It can handle various options such as specifying the delimiter, including or excluding an index column, and verbosity of output.

**Querying data:** The class has a query() method that allows querying data by columns and rows. It can retrieve values for a specific column, retrieve values for a specific row and column, and retrieve all values for a specific column.

**Establishing relationships:** The class has a relate() method that allows establishing relationships between tables based on column values. It can find matching rows in the current table and another table based on specified columns, and return the matching rows in both tables.

**Data type handling:** The class includes a \_selective\_float() function that can convert data to float if possible, otherwise keep it as is. This helps in handling numeric data in the CSV files.

**Usage:**

To use the RelationalTable class, you can create an instance of the class by providing a CSV file path or a defaultdict(list) object as the data source. Once the instance is created, you can use the provided methods to read CSV data, query data, and establish relationships between tables.

**Conclusion:**

The RelationalTable class is a useful tool for handling CSV data in a relational table format in Python. It provides convenient methods for reading CSV data, querying data by columns and rows, and establishing relationships between tables based on column values. It is a versatile class that can be used in various data processing tasks where CSV data needs to be manipulated and analyzed in a relational manner.