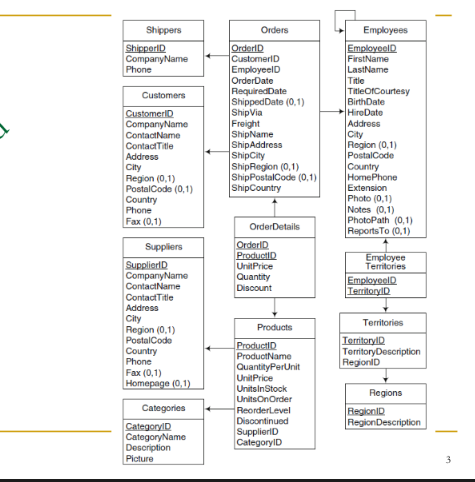
**Data warehouse and Business Intelligence**

**Motivation**

To develop an understanding of automating table creation and data population using Python scripts.

**Task 01: Database Connection and Table Creation**

### **Objective:**

Connect your SQL server to a Python file and create a database name northwind using Python. Then, create the necessary tables using Python scripts.Logical schema of northwind database is given below:  


### **Instructions**:

* Establish a connection between your SQL server and Python.
* Create the database and the necessary tables through Python (you may adjust the data types, but ensure they are handled appropriately in later tasks).

**Task 02: Data Population from CSV Files**

### Part A: Populate Tables Using CSV Files

**Objective:**  
Fill the tables by inserting data from CSV files available in the provided folder name northwind\_data.

**Instructions:**

* The CSV files are named in the format: <table\_name>.csv.
* Use the python library Pandas to populate data from csv files to the tables
* There may be difference in number of columns between csv files and corresponding table, adjust the table accordingly or put null in columns whose values are missing

### Part B: Populate Tables Without CSV Files

**Objective:**  
For tables whose CSV files are unavailable, use the Python Faker module to generate synthetic data.

**Instructions:**

* Implement the Faker module to create dummy data for missing tables.

## **Task 03: Index Creation**

### **Objective:**

Improve the performance of queries by creating indexes on specific columns.

### **Instructions**:

* Create indexes for the following keys:

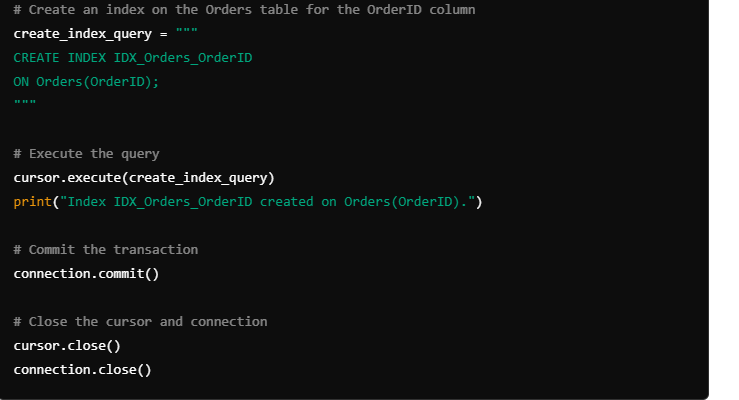
**Primary Key (PK) Indexes:**

* Orders.OrderID
* Customers.CustomerID
* Suppliers.SupplierID
* Products.ProductID
* Employees.EmployeeID

**Non-Primary Key (Non-PK) Indexes:**

* Customers.CustomerName
* Shippers.CompanyName
* Orders.OrderDate

### Python Syntax for Creating Indexes:



## **Task 04: Query Execution and Performance Measurement**

### **Objective:**

Run specific queries multiple times, observe the response time, and store the results.

### **Queries Involving PK Indexes:**

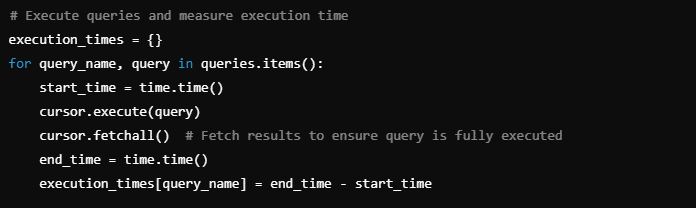
1. Retrieve all details of an order based on a specific OrderID.
2. Find all customer information using a specific CustomerID.
3. Retrieve supplier details for a given SupplierID.
4. Get product details using a specific ProductID.
5. Retrieve employee details by EmployeeID.

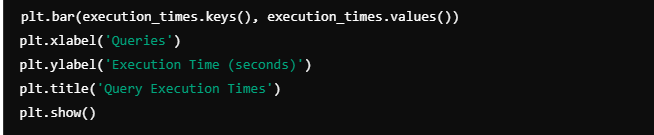
### **Queries Involving Non-PK Indexes:**

1. Retrieve all customer details for a given CustomerName.
2. Find shipper details using a specific CompanyName.
3. Retrieve all orders placed on a particular OrderDate.
4. Find all orders placed by customers with a specific CustomerName.
5. Get the list of orders shipped by a specific shipper’s CompanyName.

### **Instructions:**

* After observing the response time, create a graph to demonstrate the difference between Primary Key and Non-Primary Key query execution times.
* Use the following Python code snippets to measure execution time and to visualize it:





**Note:** You may need to multiply values by 100 or another power to magnify minor differences in execution time.