**Case Study 7:** Functional Dependency and Normalization

Consider a database for an e-commerce platform. The database has a table called "Products" with the following attributes:

ProductID (Primary Key)

ProductName

Category

Brand

Price

SupplierID

SupplierName

SupplierAddress

Write an SQL query to identify any partial dependencies and normalize the table to 3NF.

**Sol.** To identify any partial dependencies in the "Products" table and normalize it to 3NF, we can follow these steps:

Step 1: Identify the functional dependencies:

The given attributes can be functionally dependent as follows:

- ProductID -> ProductName, Category, Brand, Price

- SupplierID -> SupplierName, SupplierAddress

Step 2: Normalize the table to 1NF:

Create two tables: "Products" and "Suppliers"

Products:

- ProductID (Primary Key)

- ProductName

- Category

- Brand

- Price

- SupplierID (Foreign Key referencing Suppliers table)

Suppliers:

- SupplierID (Primary Key)

- SupplierName

- SupplierAddress

Step 3: Normalize the table to 2NF:

No partial dependencies exist, so no changes are required.

Step 4: Normalize the table to 3NF:

No transitive dependencies exist, so no changes are required.