Data Manipulation Language(DML)

In this session, you will learn:

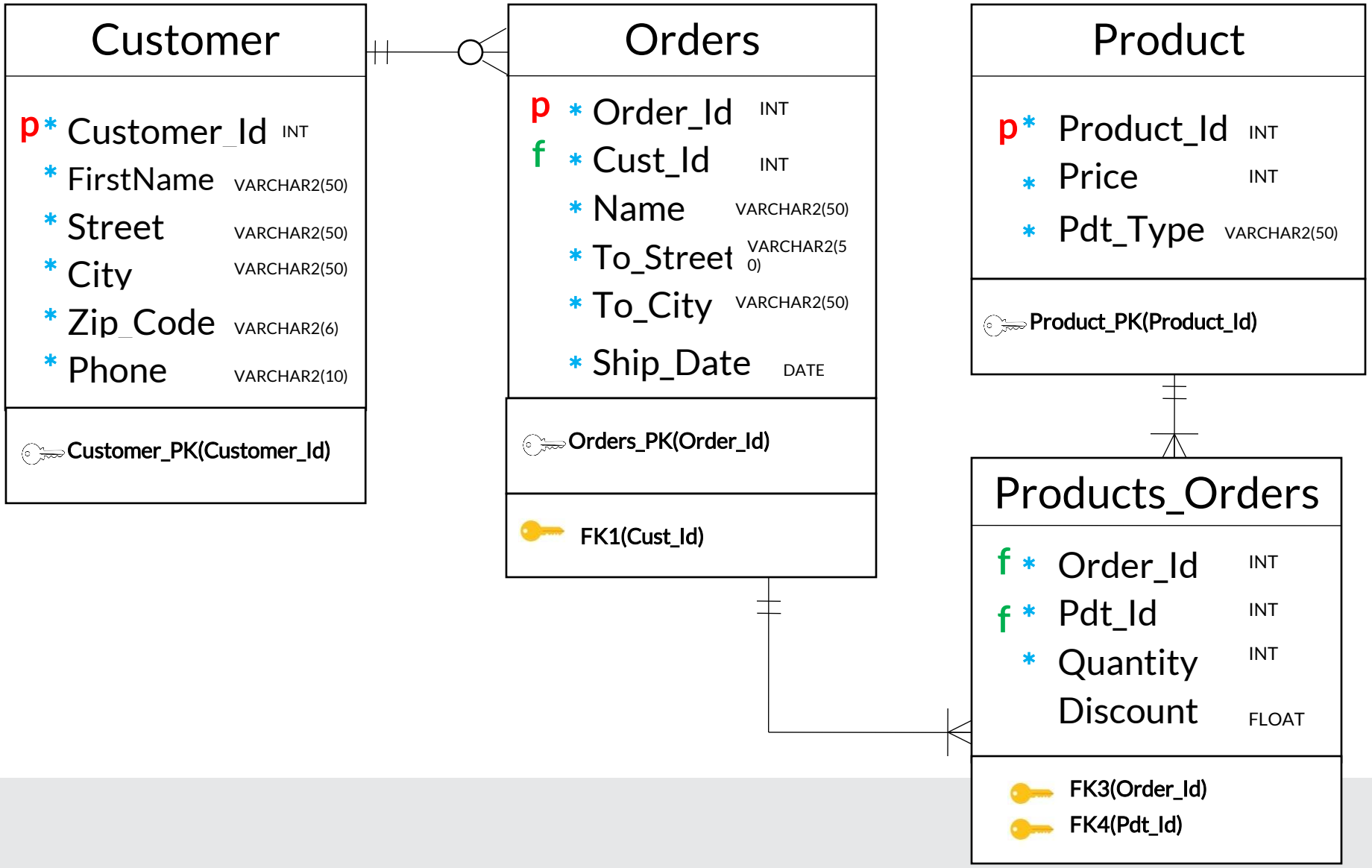
- What is Data Manipulation Language?
- DML Commands

What is Data Manipulation Language

- Data Manipulation Language is a way of telling a database exactly what you want it to do
- Used for storing and managing data in RDBMS
- DML allows a user to interact with massive amounts of data

DML Commands

Command	Description
Insert	To add a new row to a table
Update	To modify an existing row
Delete	Removes one or more rows from the table
Select	Retrieve data from a database


Customer
p * Customer_Id INT
* FirstName VARCHAR2(50)
* Street VARCHAR2(50)
* City VARCHAR2(50)
* Zip_Code VARCHAR2(6)
* Phone VARCHAR2(10)
Customer_PK(Customer_Id)
Orders
p * Order_Id INT
f * Cust_Id INT
* Name VARCHAR2(50)
* To_Street VARCHAR2(50)
* To_City VARCHAR2(50)
* Ship_Date DATE
Orders_PK(Order_Id)
FK1(Cust_Id)
Product
p * Product_Id INT
* Price INT
* Pdt_Type VARCHAR2(50)
Product_PK(Product_Id)
Products_Orders
f * Order_Id INT
f * Pdt_Id INT
* Quantity INT
Discount FLOAT
FK3(Order_Id)
FK4(Pdt_Id)

DML Command - Insert

Syntax

INSERT into table_name values(data1,data2,..)

Example

INSERT into Customer values
(100, 'ABC', 'Central', 'Chennai', '641088', '9852754812')

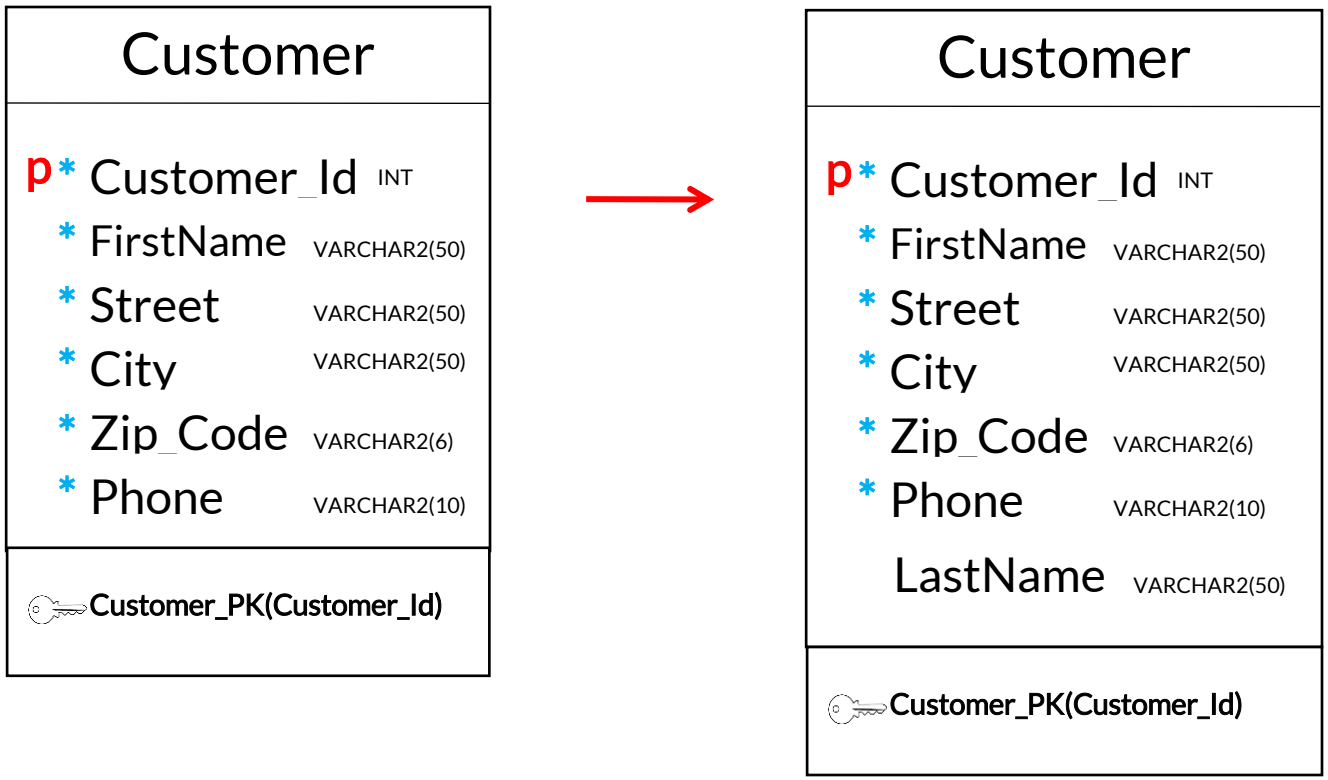
Use single quotes (' ') during insertion of varchar type attribute

Before Insert

Customer_ Id	FirstName	Street	City	Zip_Code	Phone

After insert

Customer_ Id	First Name	Street	City	Zip_Code	Phone
100	ABC	Central	Chennai	641088	9852754812


Customer
p * Customer_Id INT
* FirstName VARCHAR2(50)
* Street VARCHAR2(50)
* City VARCHAR2(50)
* Zip_Code VARCHAR2(6)
* Phone VARCHAR2(10)
Customer_PK(Customer_Id)
Customer
p * Customer_Id INT
* FirstName VARCHAR2(50)
* Street VARCHAR2(50)
* City VARCHAR2(50)
* Zip_Code VARCHAR2(6)
* Phone VARCHAR2(10)
LastName VARCHAR2(50)
Customer_PK(Customer_Id)

DML Command - Insert

Customer Table

Before Insert

Customer_Id	First Name	Steet	City	Zip_Code	Phone	LastName

Example

INSERT into Customer values
(101, 'XYZ','Central', 'Chennai', '641088', '9859954812',NULL)

Customer Table

After insert

Customer_Id	First Name	Street	City	Zip_Code	Phone	LastName
101	XYZ	Central	Chennai	641088	9859954812	null

Inserting Default Values to a column

```
CREATE TABLE Product_Orders
(
Order_Id int NOT NULL,
Pdt_Id int NOT NULL,
Quantity int NOT NULL,
Discount float default 0,
foreign key(Order_Id) references Orders
(Order_Id),
foreign key(Pdt_id) references Product
(Product_Id));
```

OR

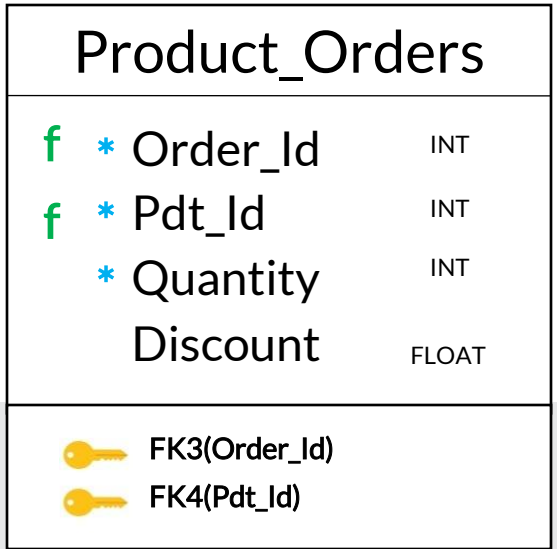
```
ALTER TABLE Product_Orders
MODIFY DISCOUNT FLOAT
DEFAULT 0;
```

Before Insert

Order_Id	Pdt_Id	Quantity	Discount

After Insert

Order_Id	Pdt_Id	Quantity	Discount
200	300	10	0



Example

```
INSERT into Product_Orders
values(200,300,10,default)
```

DML Command - Update

- Used to update existing records in a table.

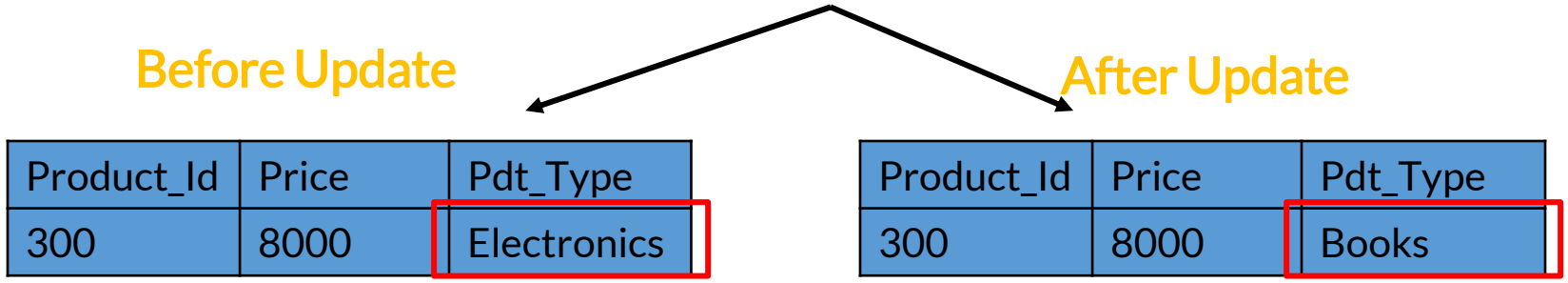
Syntax

UPDATE table_name set column_name = value where condition;

Example

UPDATE Product set Pdt_Type='Books' where Product_Id=300;

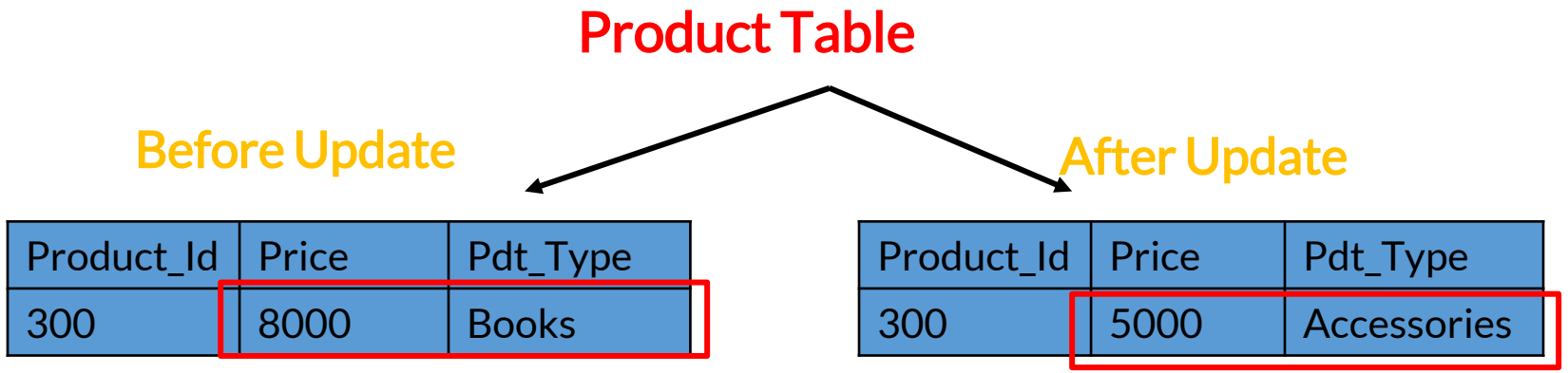
Product Table



DML Command – Update multiple columns

Example

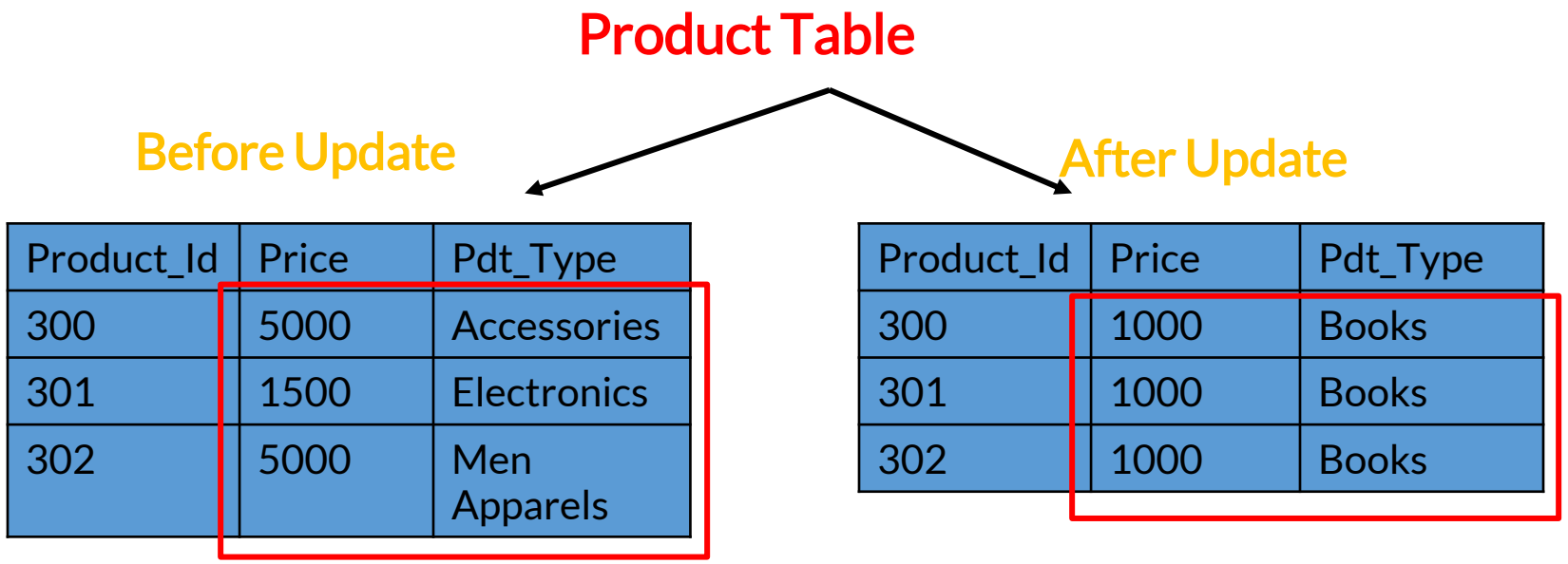
UPDATE Product set Price = 5000, Pdt_Type='Accessories' where Product_Id=300;



DML Command – Update Warning!!!

Example

UPDATE Product set Price = 1000, Pdt_Type='Books'



Product_Id	Price	Pdt_Type
300	5000	Accessories
301	1500	Electronics
302	5000	Men Apparels

Product_Id	Price	Pdt_Type
300	1000	Books
301	1000	Books
302	1000	Books

DML Command – Delete

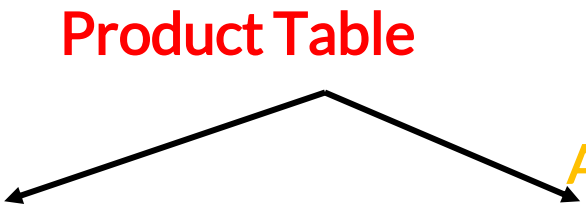
- used to delete data from a table

Syntax

delete from table_name where condition;

Example

delete from Product



Before Delete

Product_Id	Quantity	Pdt_Type
300	10	Books
301	10	Books
302	10	Books

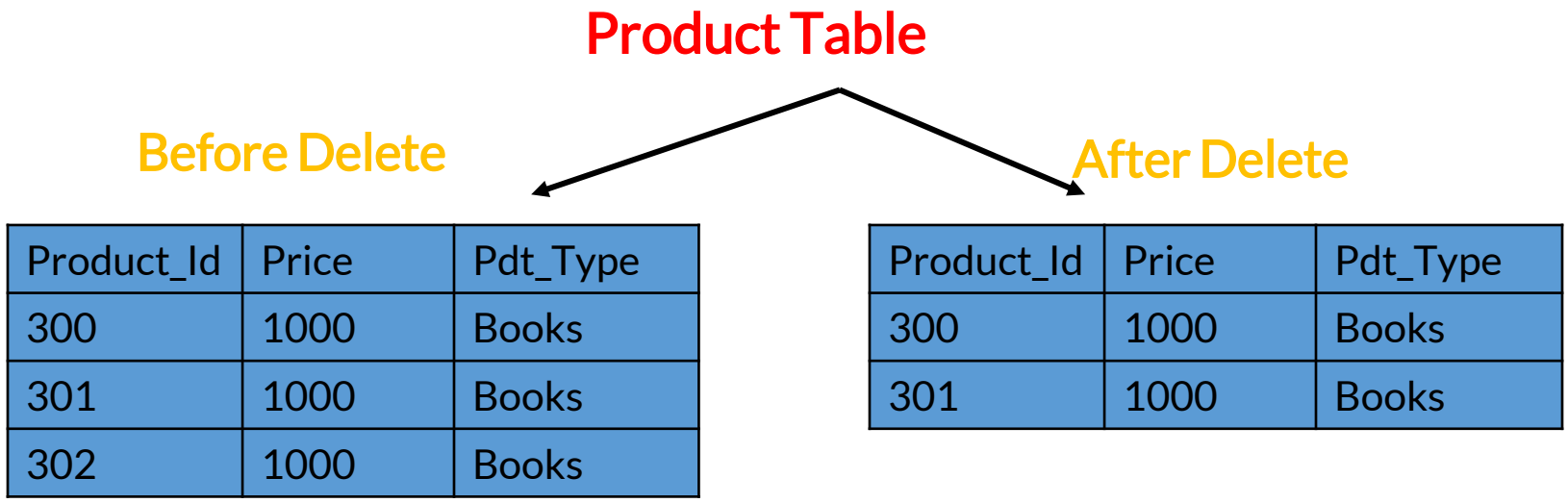
After Delete

Product_Id	Quantity	Pdt_Type

DML Command – Delete

Example

delete from Product where Product_Id = 302



Product_Id	Price	Pdt_Type
300	1000	Books
301	1000	Books
302	1000	Books

Product_Id	Price	Pdt_Type
300	1000	Books
301	1000	Books

Differences between TRUNCATE and DELETE

Speed: DELETE is slower than TRUNCATE.

Rollback: TRUNCATE cannot be rolled back like a DML operation.

Rows: TRUNCATE removes all rows from a table, while DELETE may remove rows conditionally.

THANKS