**ASSIGNMENT FRONT SHEET <No.2>**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification** | **BTEC Level 5 HND Diploma in Computing and Systems Development** | | |
| **Unit number and title** | Unit 04: Database Design & Development | | |
| **Assignment due** |  | **Assignment submitted** |  |
| **Learner’s name** |  | **Assessor name** | Dao Ngoc Anh |
| **Learner’s ID** |  | **Submission number** |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Learner declaration:**  I certify that the work submitted for this assignment is my own and research sources are fully acknowledged. | | | |
| **Learner signature** |  | **Date** |  |

**Grading grid**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P1 | P2 | P3 | P4 | P5 | M2 | M3 | M4 | M5 | D2 | D3 |
|  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| Learning Outcomes and Assessment Criteria | | |
| Pass | Merit | Distinction |
| **LO1** Use an appropriate design tool to design a relational database system for a substantial problem | | |
| **P1** Design a relational database system using appropriate design tools and techniques, containing at least four interrelated tables, with clear statements of user and system requirements. | **M1** Produce a comprehensive design for a fully functional system which includes interface and output designs, data validations and data normalisation. | **D1** Assess the effectiveness of the design in relation to user and system requirements. |

|  |  |  |
| --- | --- | --- |
| Learning Outcomes and Assessment Criteria | | |
| Pass | Merit | Distinction |
| **LO2** Develop a fully functional relational database system, based on an existing system design.  **LO3** Test the system against user and system requirements.  **LO4** Produce technical and user documentation | | |
| **P2** Develop the database system with evidence of user interface, output and data validations, and querying across multiple tables.  **P3** Implement a query language into the relational database system. | **M2** Implement a fully functional database system which includes system security and database maintenance.  **M3** Assess whether meaningful data has been extracted through the use of query tools to produce appropriate management information. | **LO2 & 3 D2** Evaluate the effectiveness of the database solution in relation to user and system requirements, and suggest improvements. |
| **P4** Test the system against user and system requirements. | **M4** Assess the effectiveness of the testing, including an explanation of the choice of test data used. |
| **P5** Produce technical and user documentation. | **M5** Produce technical and user documentation for a fully functional system, including ER Diagram and normalization statements and describing how the system works. | **D3** Assess any future improvements that may be required to ensure the continued effectiveness of the database system. |

|  |
| --- |
| **Assignment Brief** |
| ElectroShop is a company who acquire and sell electronic devices to customers throughout Vietnam. They are looking to apply their current data store system with a relational database.  The company takes orders from customers, who can order any quantity of many items that ElectroShop have in their current catalogue. The catalogue includes TV, Phone, etc.  Each of ElectroShop’s suppliers supplies many items but each item is supplied by one main supplier.  ElectroShop would like to add all the customers to the database so that they can send a mail-shot to them with any offers that are available, as well as the catalogue which is produced annually.  Salespersons at ElectroShop are paid a monthly bonus which is determined by the amount of sales they have made for each month. The company would like the bonus calculated automatically by the system.  As well as the monthly bonus figures ElectroShop would like to be able to create comparative reports from the system i.e. sales figures for each month this year compared to last year. |

|  |
| --- |
| **Submission Format** |
| You are required to make use of appropriate structure, including headings, paragraphs, subsections and illustrations as appropriate, and all work must be supported with research and referenced using the Harvard referencing system. |

|  |  |
| --- | --- |
| **Assignment title** | **Understand databases and data management systems** |
| In this assignment, you will have opportunities to provide evidence against the following criteria.  Indicate the page numbers where the evidence can be found. | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Summative feedback:** | | | |
| **Assessor’s Signature** |  | Date |  |

**Table of Contents**

P1 Design a relational database system using appropriate design tools and

techniques, containing at least four interrelated tables, with clear statements of user

and system requirements...............................................................................................................……….10

* 1. Customer order bill for electronic supply store.................................................................…………..10
  2. Normalization.................................................................................................................……….11
  3. common forms..................................................................................................................…….11
  4. database Normalization table:.............................................................................................……113
  5. The Entity Relationship Diagram:.........................................................................................…..16

**P2** Develop the database system with evidence of user interface, output and data validations,

and querying across multiple tables. ........................................................................................……………..17

2.1 Create a database named companyelectroshop and query it: .......................................................….17

2.2 create tables:.........................................................................................................................….17

2.3 create constraint.....................................................................................................................….19

2.4 Database Diagram :.................................................................................................................….22

2.5 User interface:............................................................................................................................23

**P3** Implement a query language into the relational database system............................................................28

3.1 insert function:......................................................................................................................…..28

3.2 update function:...................................................................................................................……..29

3.3 delete function:..........................................................................................................................30

3.4 select fonction:...........................................................................................................................30

3.5 statement using a combination of COUNT (), SUM (), AVG (), MAX (), MIN ()

  Combined with group by, where, having, order by..................................................................................31

3.6 advanced retrieval statements:...............................................................................………………………..32

**P4** Test the system against user and system requirements....................................................................……33

4.1. Test case....................................................................................................................................33

Query for test case .........................................................................................................................35

4.2. Test Log ....................................................................................................................................37

Query for test log......................................................................................................................... .40

**P5** Produce technical and user documentation .....................................................................................……42

5.1 User document ..........................................................................................................................42

1. Manage Suppliers.............................................................................................................................42
2. Manage Customer.............................................................................................................................43
3. Manage Employee ............................................................................................................................44
4. Manage Product................................................................................................................................45
5. Manage bill.......................................................................................................................................46

5.2 Technical documentation .............................................................................................................47

* Using Code to create database in SQL Management Server.....................................................................49
* Using Tools to create database in SQL Management Server.....................................................................52

**References**..........................................................................................................................................60

**Assignment Database Design and Development**

**P1** Design a relational database system using appropriate design tools and

techniques, containing at least four interrelated tables, with clear statements of user

and system requirements.

1.1 Customer order bill for electronic supply store.

|  |
| --- |
| **Bill ID**: 1234567  **Date**: 7/09/2019  **Customer Name**: le thai bao  **employee**: nguyen hoang tuan vu |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **product Name** | **Supplier** | **category** | **Quantity** | **Price** |
| **Apple MacBook Pro** | **Apple** | laptop | 2 | 3.600 USD |
| **Lenovo Thinkpad X1 Carbon** | **Lenovo** | laptop | 1 | 1.329 USD |
| **Dell XPS 13** | dell | laptop | 2 | 3.400 USD |
| Oppo F1S | oppo | smartphone | 3 | 800 USD |

* The objects that an order need s: customers, employees, products, suppliers, bill:
* Customers : customer ID,customer name ,customer address,customer numbers phone.
* Employees : employee ID ,employee name ,employee gender,employee numbers.
* Products : product ID,product name ,unitprice
* Suppliers : suppliers ID ,suppliers name,suppliers address,suppliers numberphone.
* Bill : bill ID,date.

1.2 Normalization.

* Standardization is a technique for organizing data in a database. It is important that the database is standardized to minimize redundancy (duplicate data) and to ensure that only relevant data is stored in each table. It also prevents any problems resulting from database modifications such as insertion, deletion and updating.
* Each database must ensure that it meets three common forms:
* First Normal Form
* Second Normal Form
* Third Normal Form

1.3 common forms.

1. First Normal Form (1NF).

First normal form relates to the duplication and over-grouping of data in tables and columns.

* There is no primary key
* The data is not in its most reduced form.
* There are two repeating groups of columns

1. Second Normal Form(2NF)

* a database must first satisfy all the conditions for 1NF.
* satisfying 2NF requires that all data in each table relates directly to the record that the primary key of the table identifies.

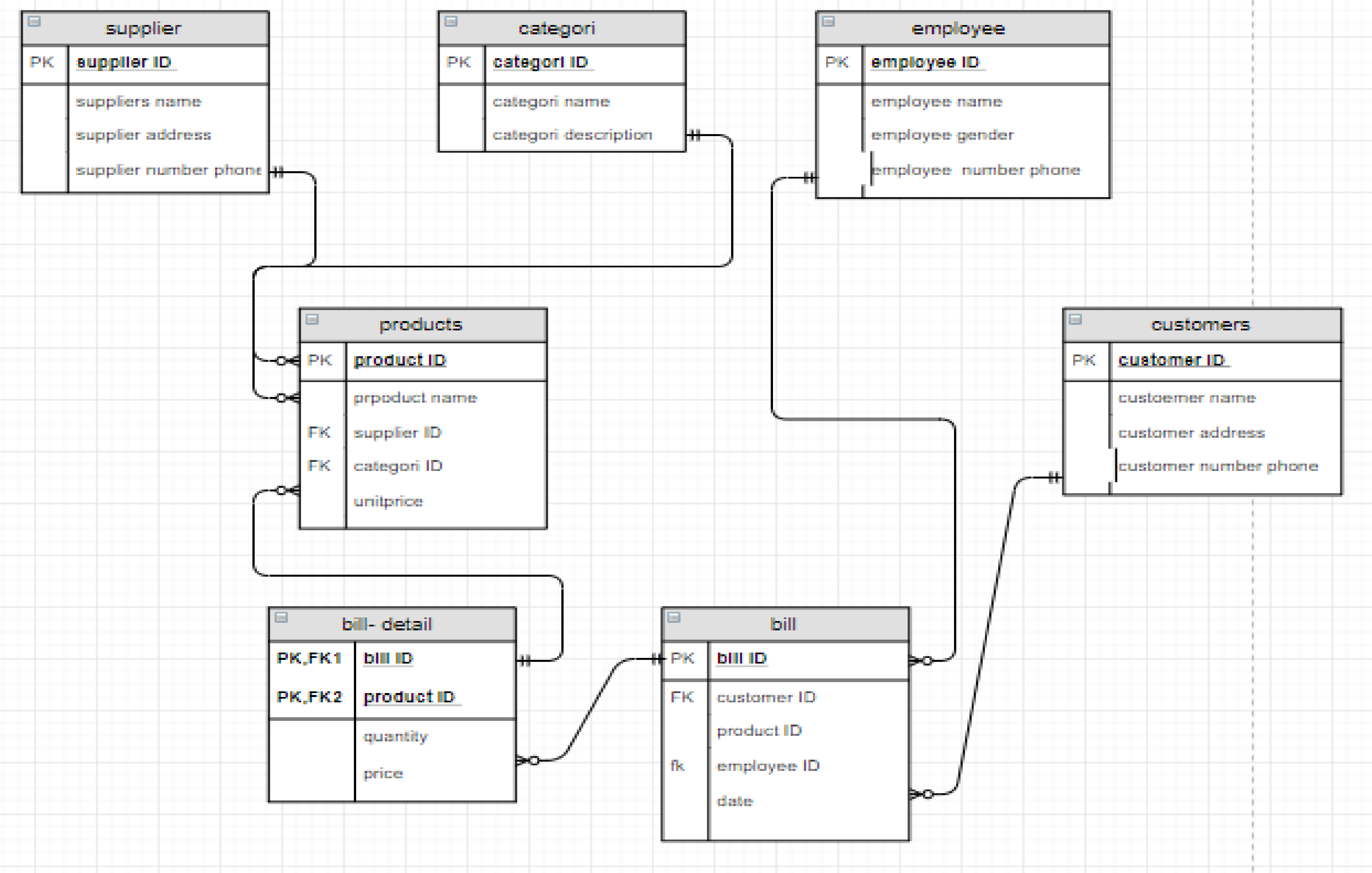
1. Third Normal Form(3NF)

* database must first satisfy all the criteria for 2NF (and therefore, also 1NF).
* each column must be non-transitively dependent on the table’s primary key. This means that all columns in a table should rely on the primary key and no other column.

1.4 database Normalization table:

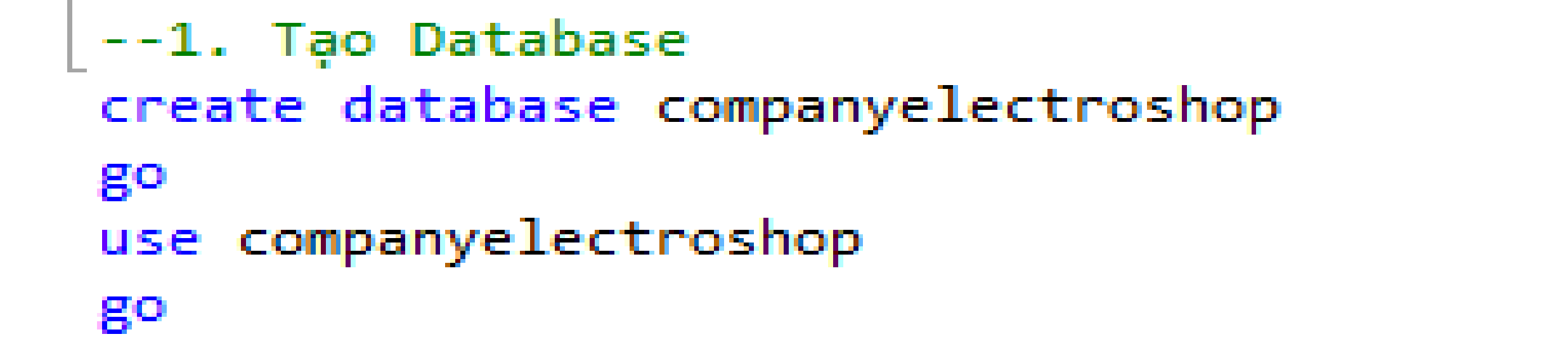
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0NF | 1NF | 2NF | 3NF | relationship |
| suppliers ID | **bill id** | **bill id** | **suppliers ID (pk)** |  |
| suppliers name | date | date | suppliers name |  |
| suppliers address | customer ID | customer ID | suppliers address | supplier |
| suppliers numberphone | customer name | customer name | suppliers numberphone |  |
| customer ID | customer address | customer address |  |  |
| customer name | customer numbers phone | customer numbers phone | **customer ID (pk)** |  |
| customer address | employee ID | employee ID | customer name | customers |
| customer numbers phone | employee name | employee name | customer address |  |
| categori ID | employee gender | employee gender | customer numbers phone(not null) |  |
| categori name | employee numbers | employee numbers |  |  |
| categori descripsion |  |  | **categori ID(pk)** |  |
| employee ID |  |  | categori name | categori |
| employee name | **bill id \*** | **bill id \*** | categori descripsion(not null) |  |
| employee gender | suppliers ID | product ID |  |  |
| employee numbers | suppliers name | categori ID | **product ID(pk)** |  |
| product ID | suppliers address | categori name | product name | product |
| product name | suppliers numberphone | categori descripsion | **suppliers ID \*** |  |
| unitprice | categori ID |  | **categori ID\*** |  |
| quantity | categori name |  | unitprice |  |
| bill id | categori descripsion |  |  |  |
| date | **product ID** | **customerid\*** | **employee ID (pk)** | employees |
|  | product name | **product ID\*** | employee name |  |
|  | unitprice | quantity | employee gender |  |
|  | quantity |  | employee numbersphone |  |
|  |  | **product ID** |  |  |
|  |  | product name | **bill ID(pk)** |  |
|  |  | unitprice | **customerid\*** |  |
|  |  | suppliers ID | **employee ID \*** | bill |
|  |  | suppliers name | date |  |
|  |  | suppliers address |  |  |
|  |  | suppliers numberphone |  |  |
|  |  | categori ID | **bill ID (FPK)** |  |
|  |  | categori name | **product ID\*(FPK)** |  |
|  |  | categori descripsion | quantity | bill and products |

1.5 The Entity Relationship Diagram:



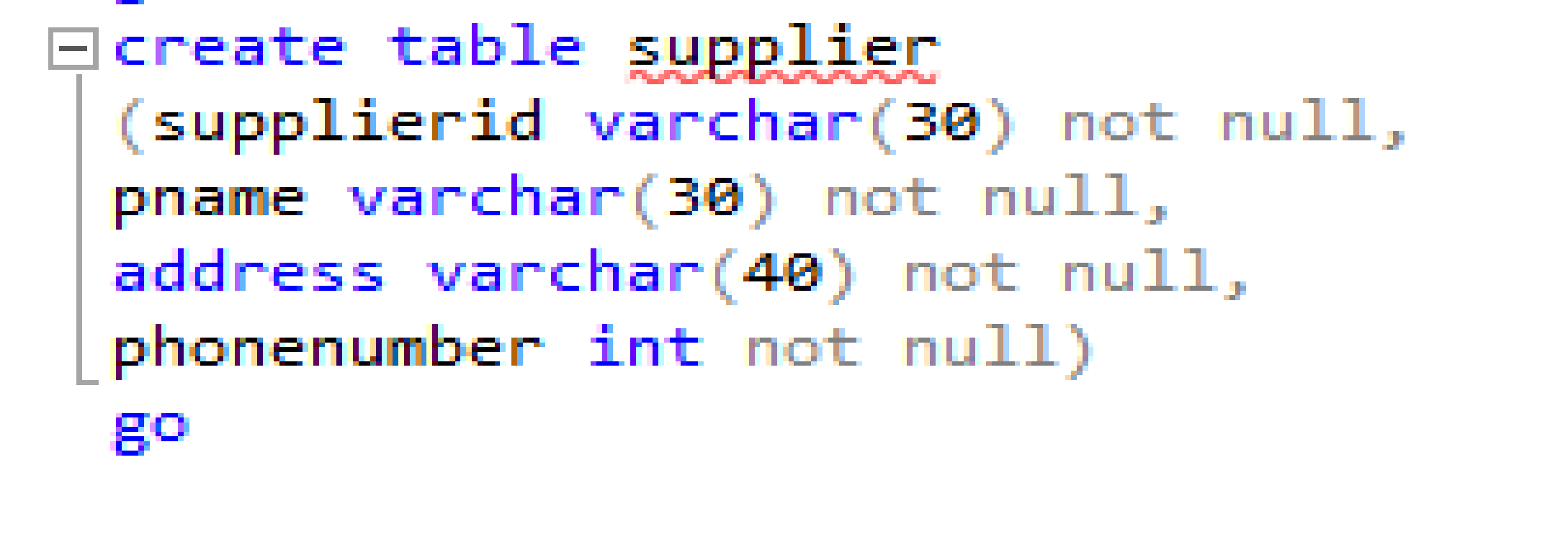
**P2** Develop the database system with evidence of user interface, output and data validations, and querying across multiple tables.

2.1 Create a database named companyelectroshop and query it:

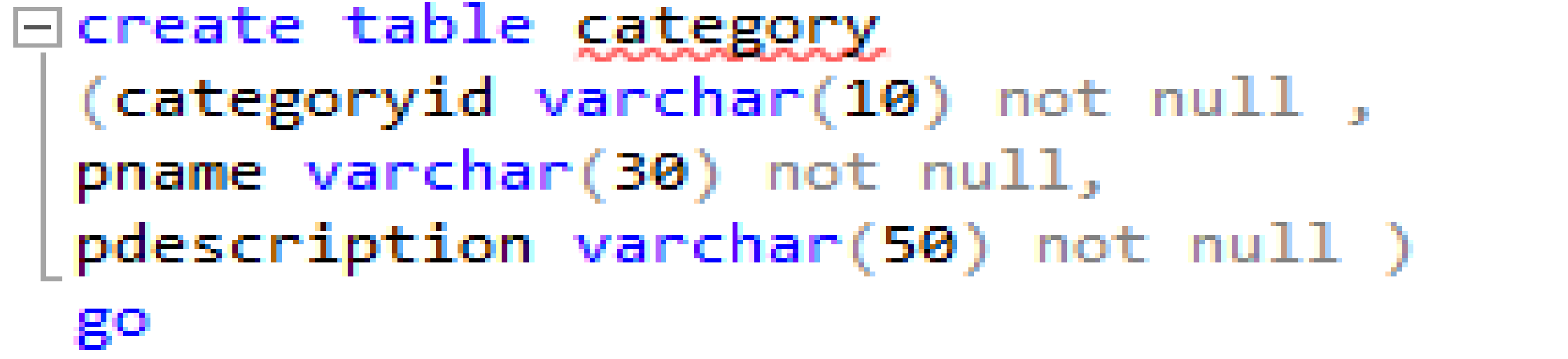


2.2 create tables:

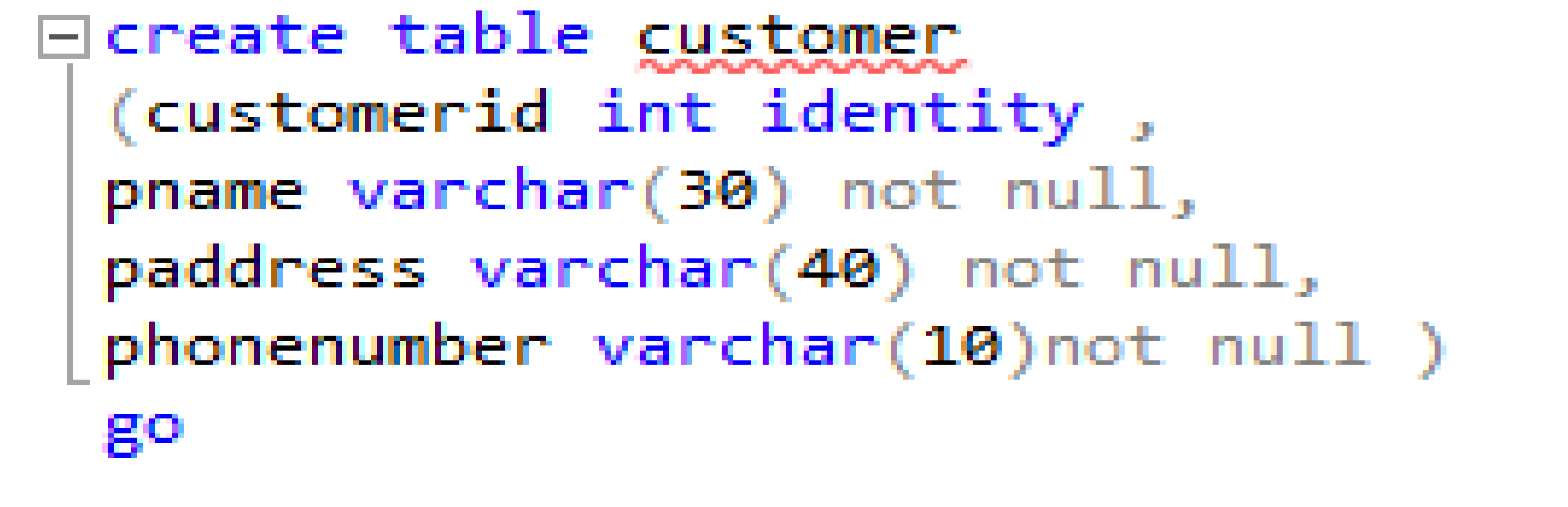
1. Create supplier table:



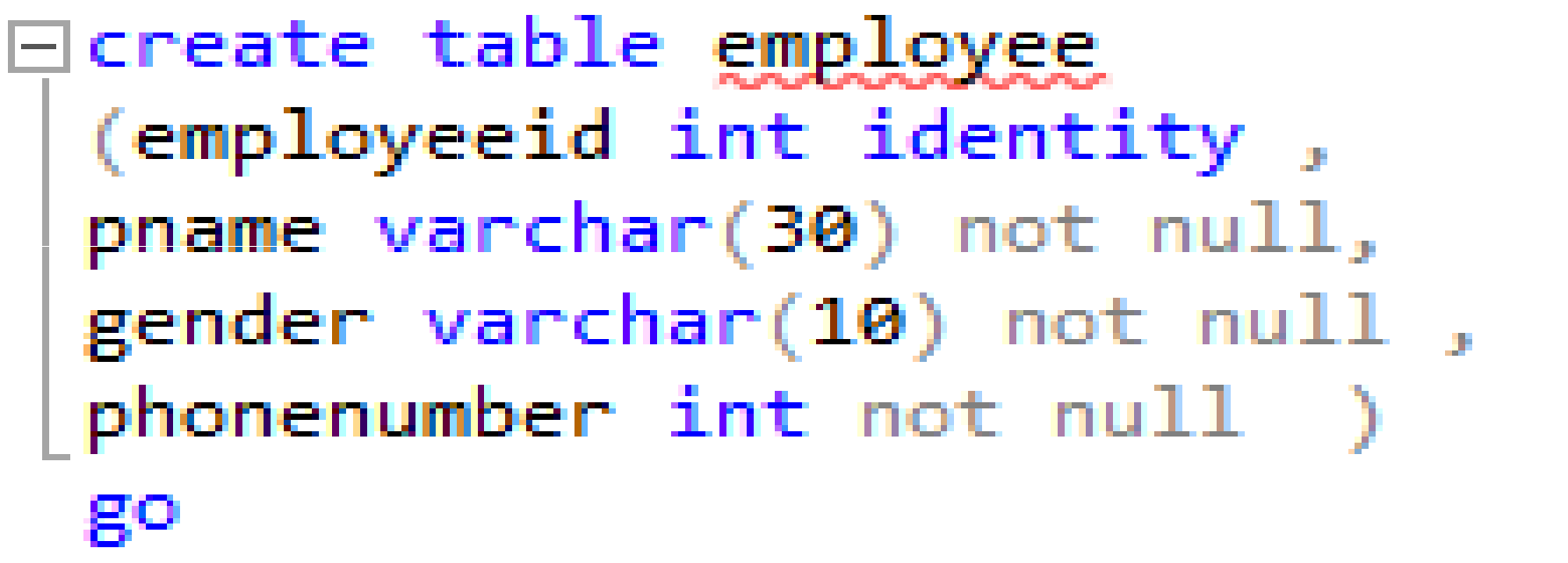
1. Create categories table:



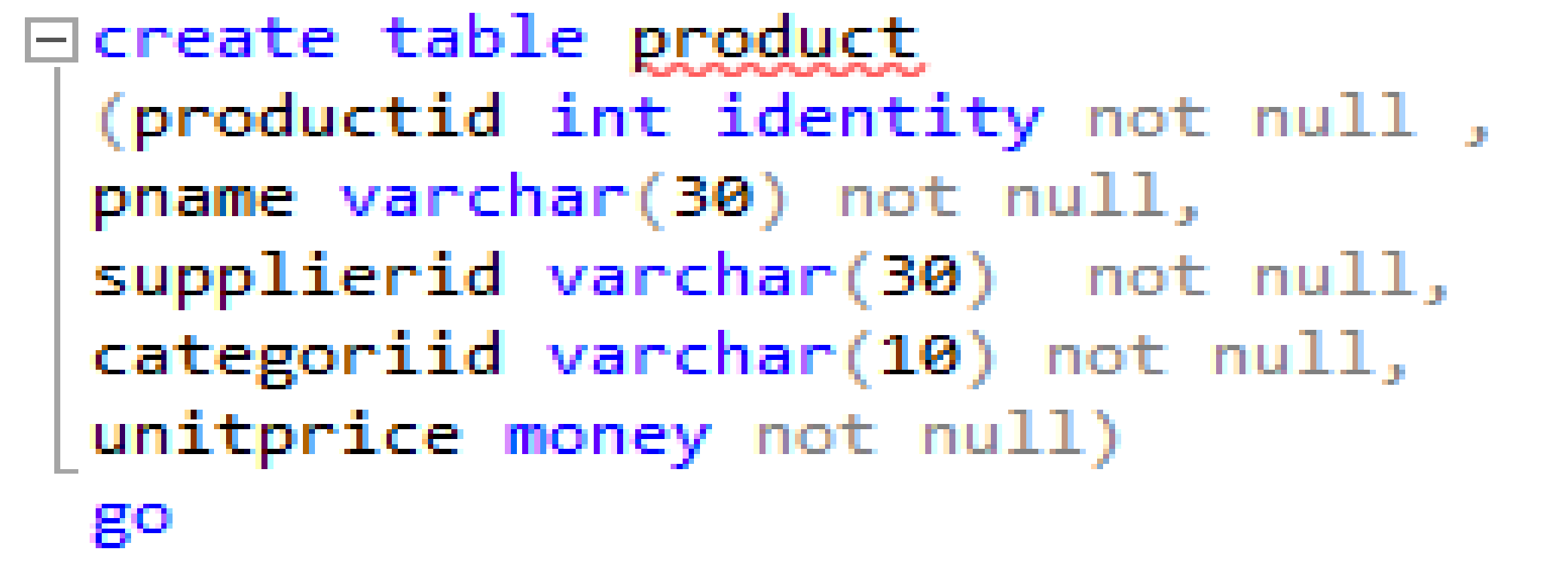
1. Create customer table:



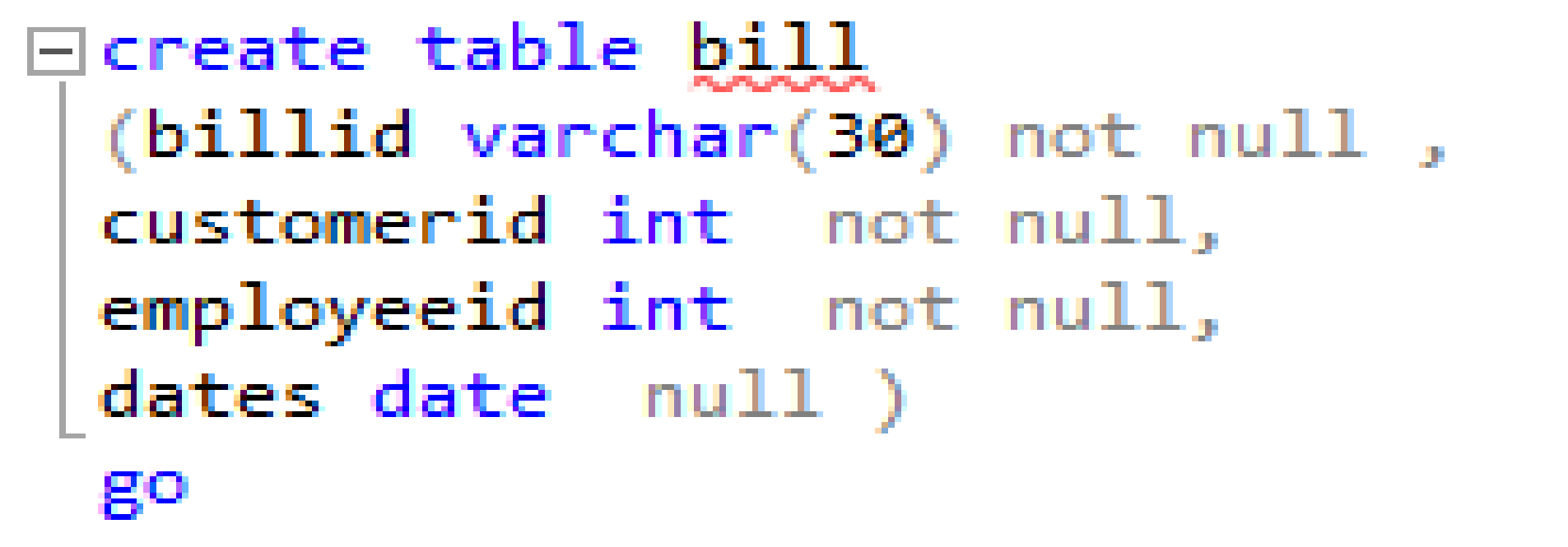
1. Create employee table:



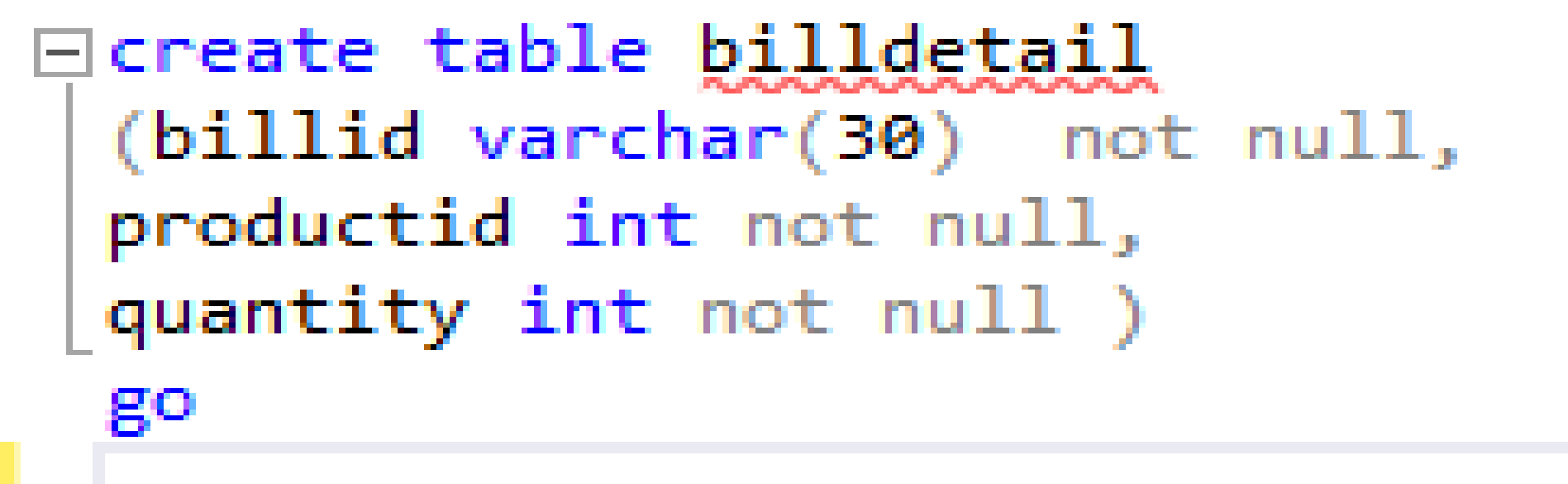
1. Create product table:



1. Create bill table:

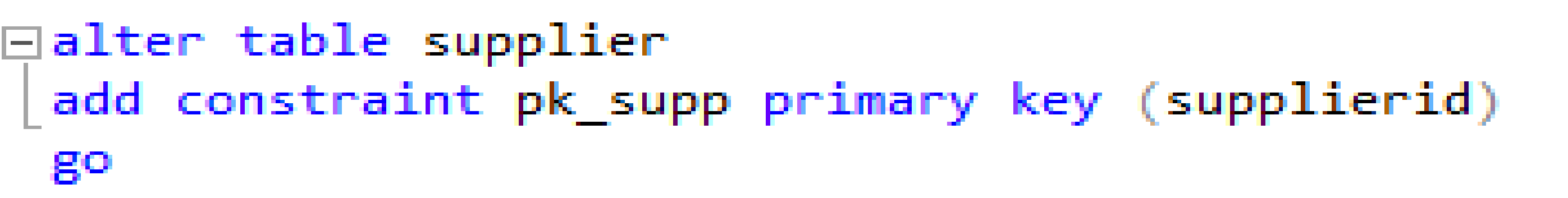


1. Create bill-detail table:

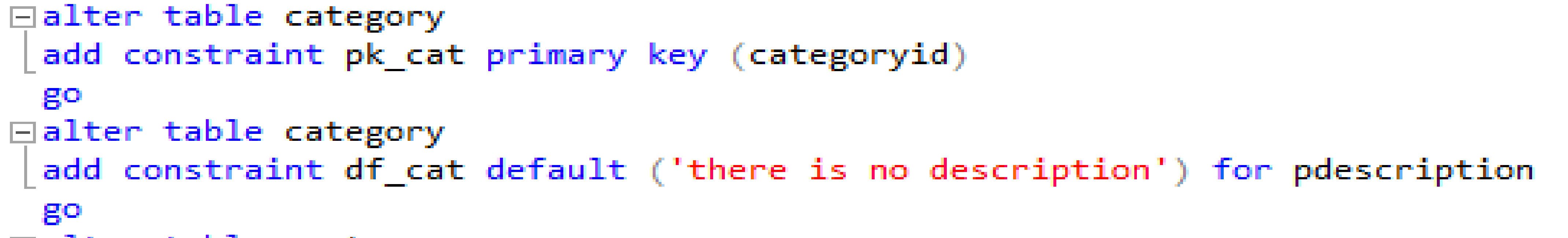


2.3 create constraint :

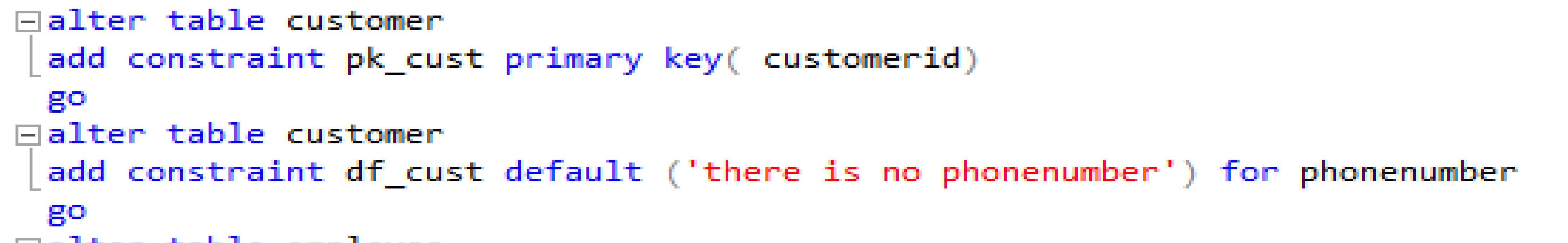
* Create constraint for supplier:



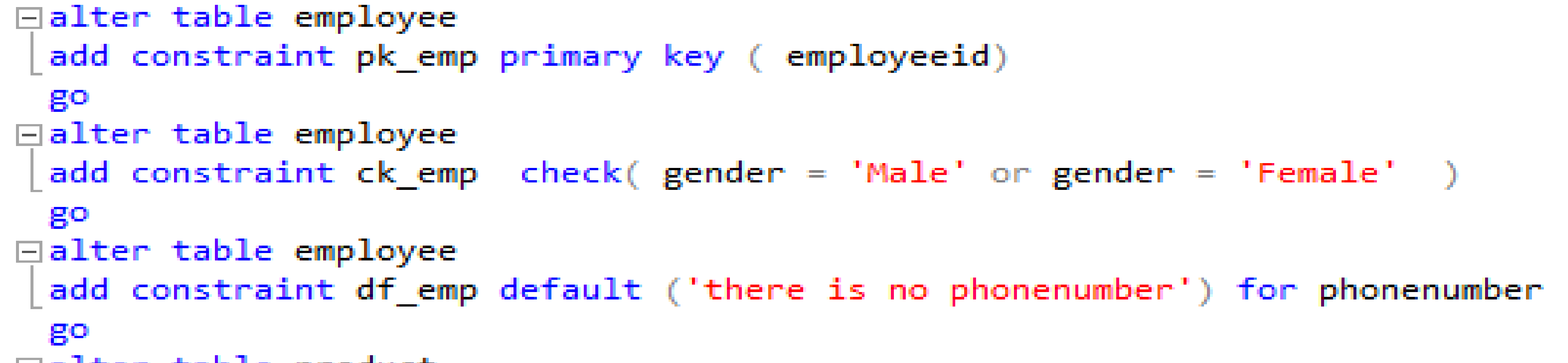
* Create constraint for category:



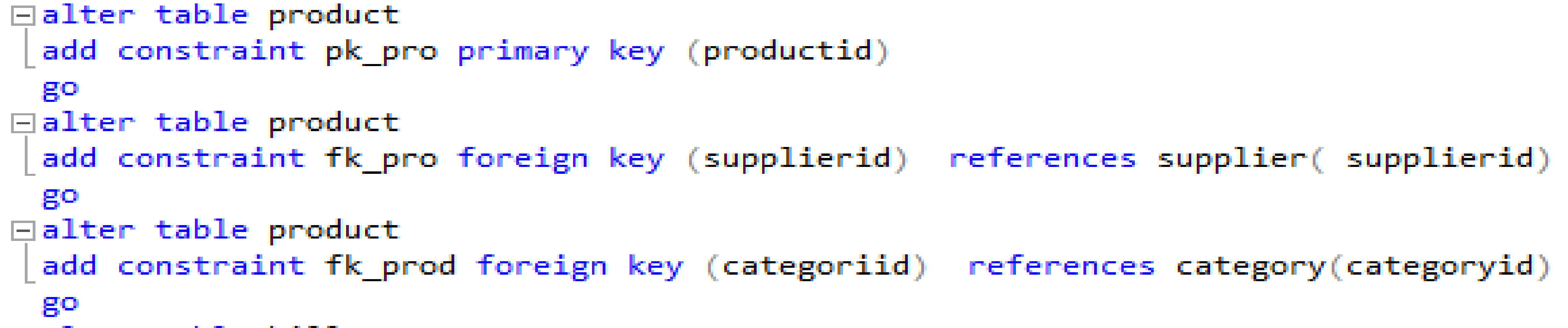
* Create constraint for customer:



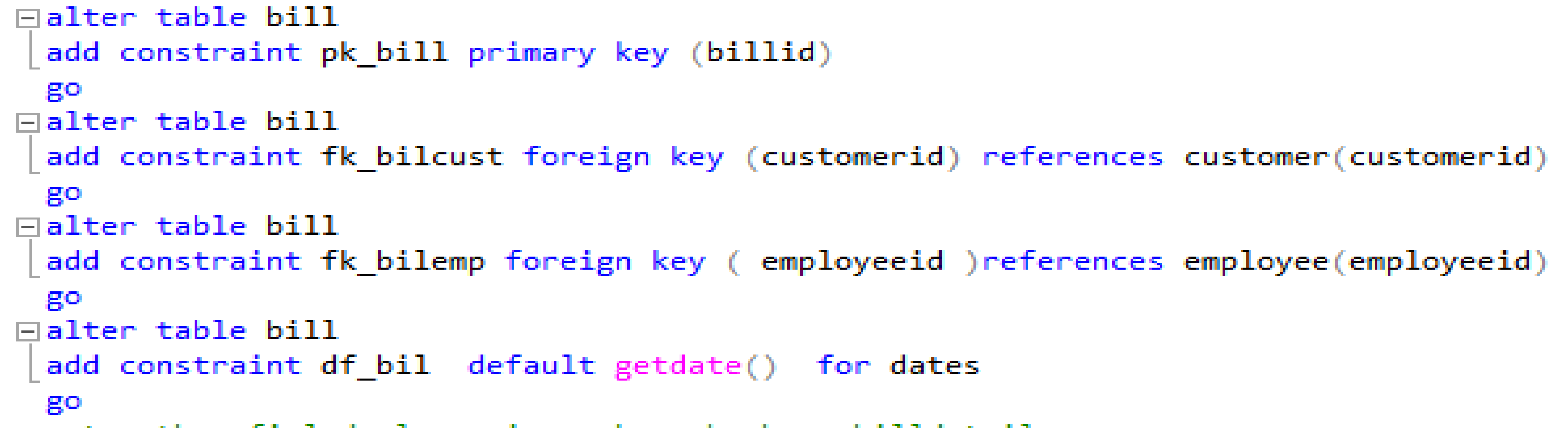
* Create constraint for employee:



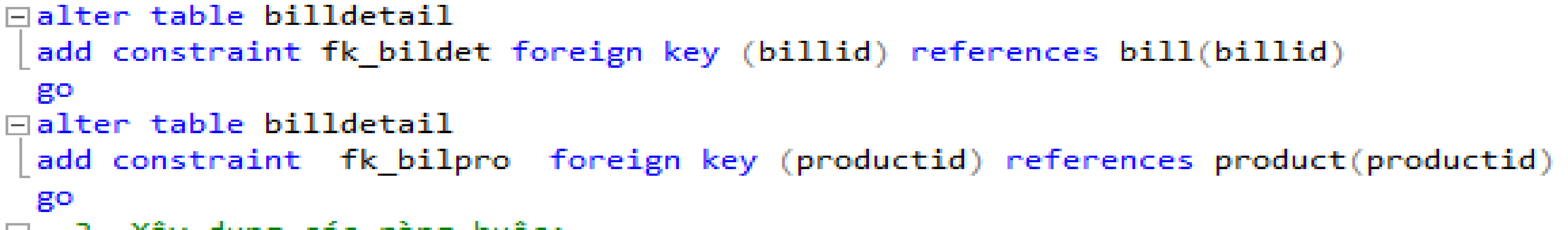
* Create constraint for product:



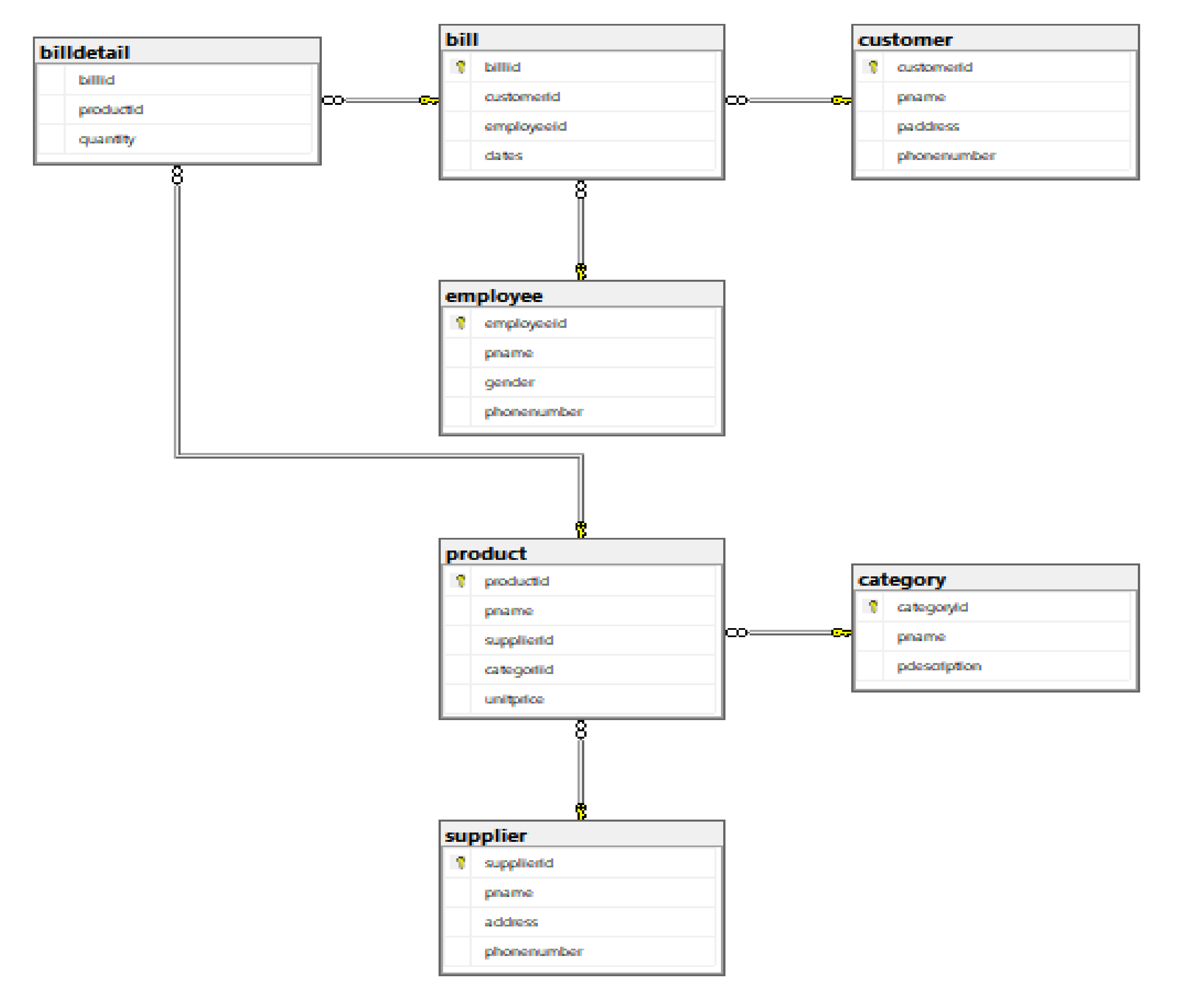
* Create constraint for bill:



* Create constraint for bill-detail:



2.4 Database Diagram :



Database Diagram

1. There are 4 one to many relationship in Database Diagram :

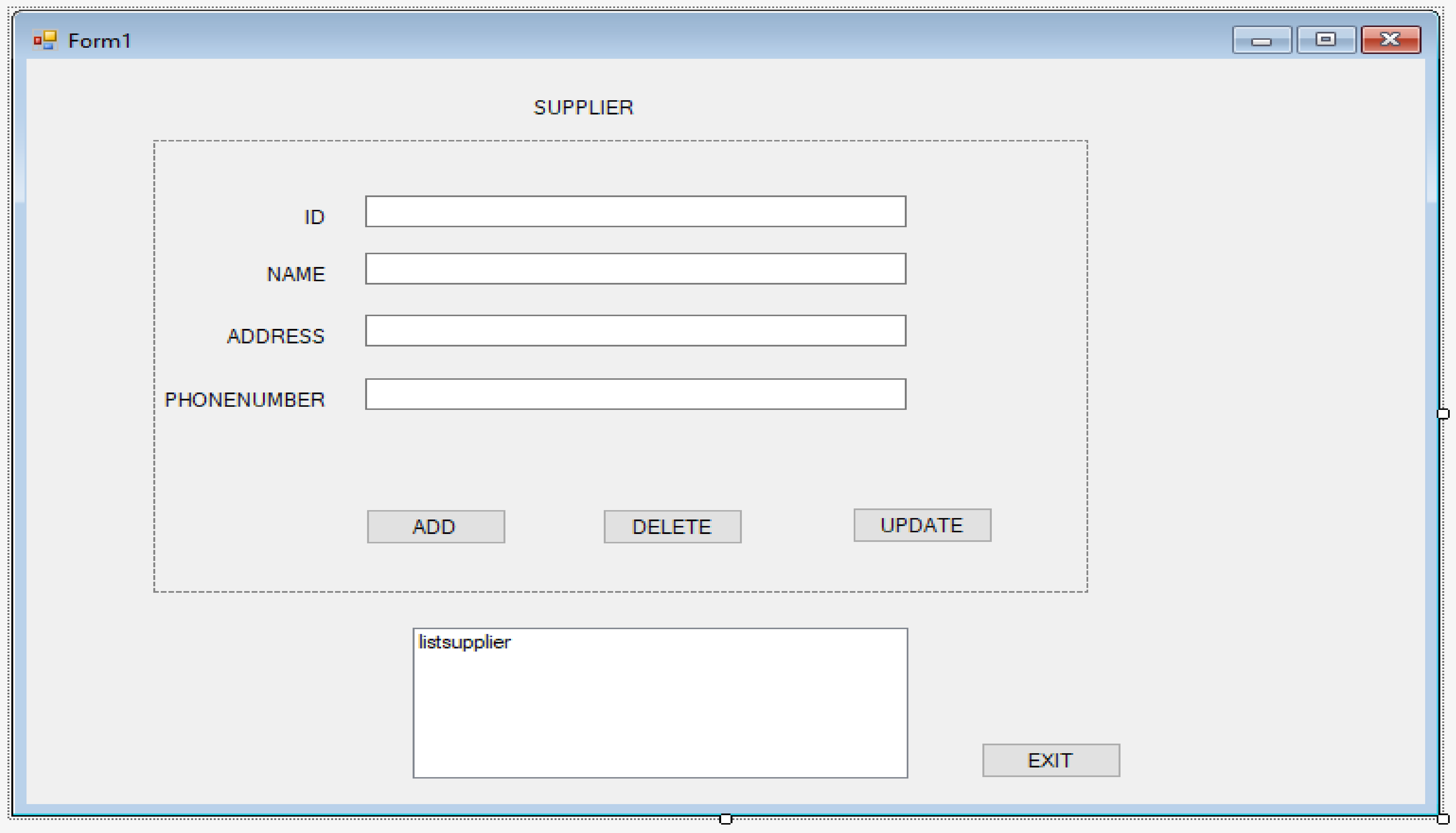
* Category vs product : A product category can have multiple products but a product can only belong to one product category.
* Supplier vs product:A supplier may provide multiple products but only one product can be supplied by a supplier.
* Bill vs customer:A customer may have multiple bills but one bill belongs to only one customer.
* Bill vs employee:An employee may sell multiple bills but a bill may only be sold by one employee.

1. There are 1 many to many relationship in Database Diagram :

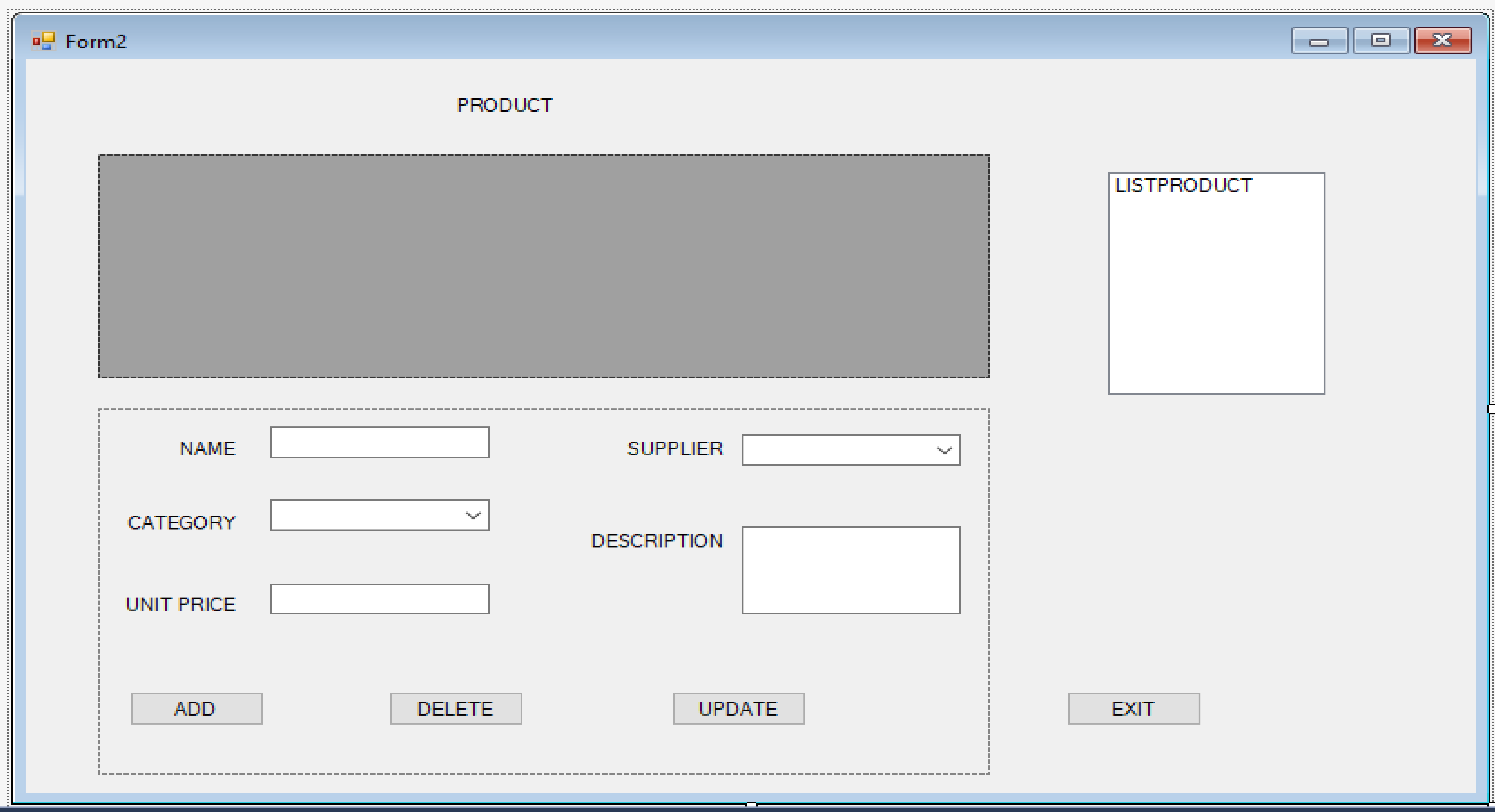
* Product vs bill .

2.5 User interface:

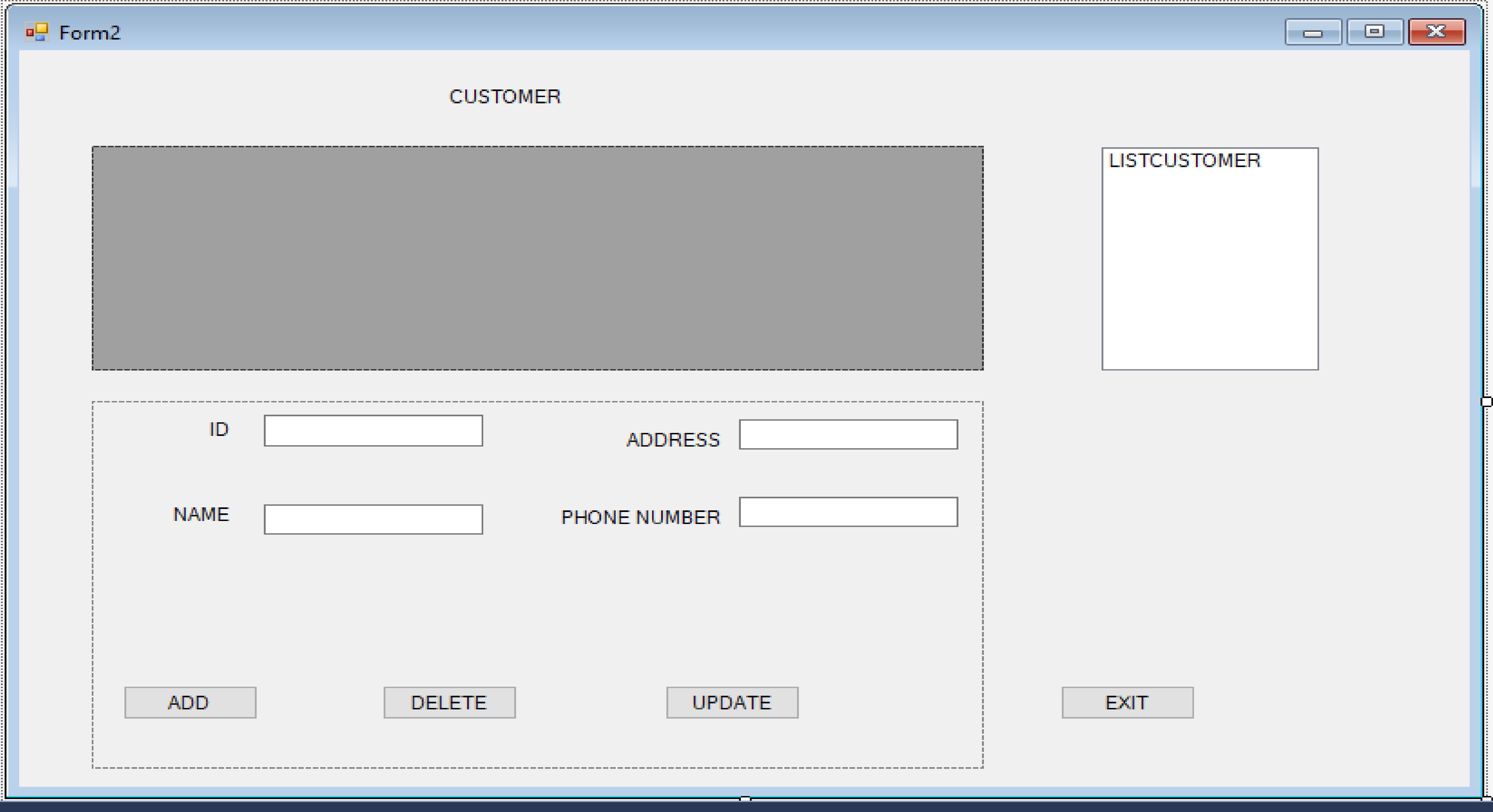
1. User interface of supplier table



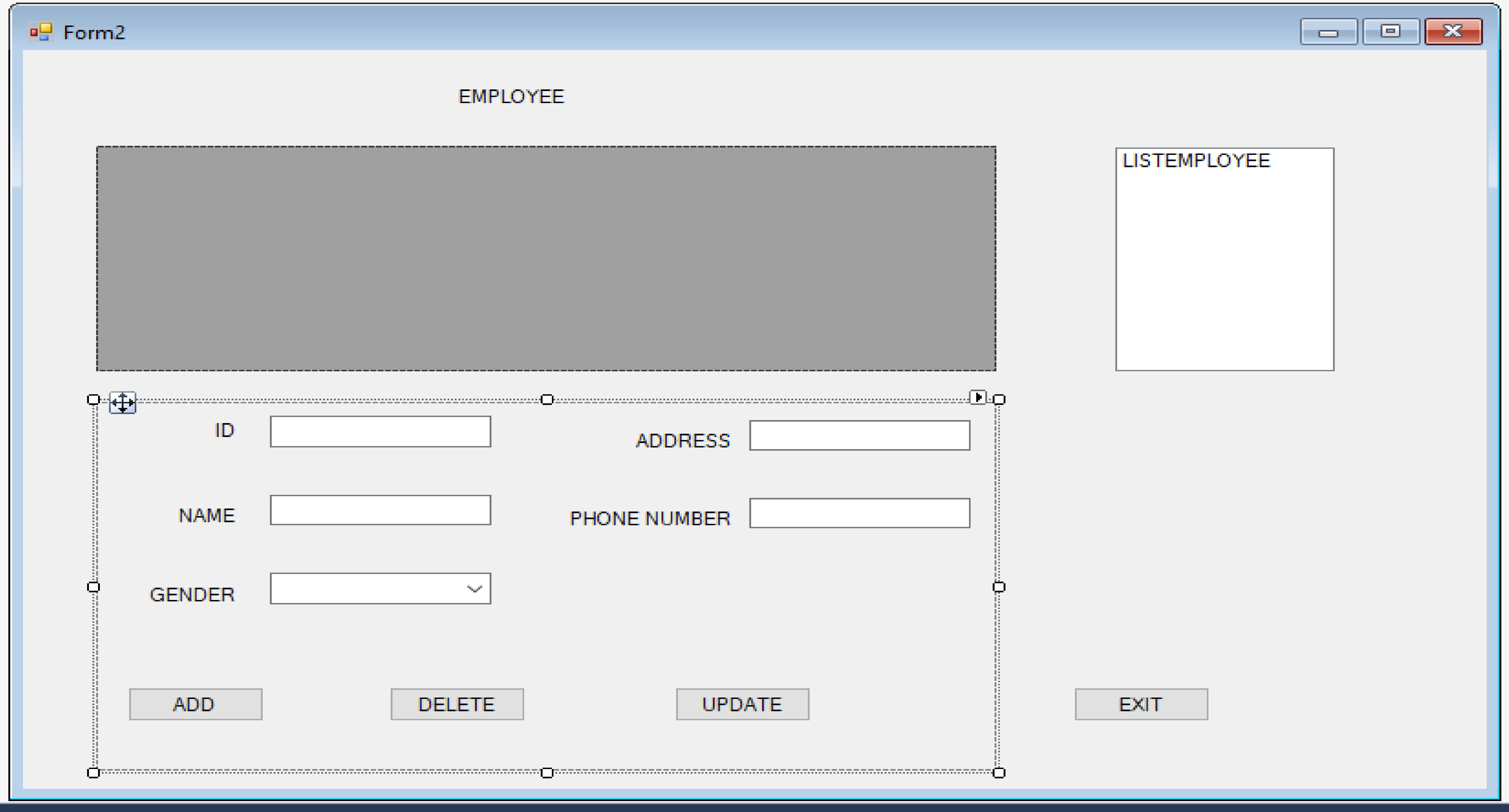
1. User interface of product table



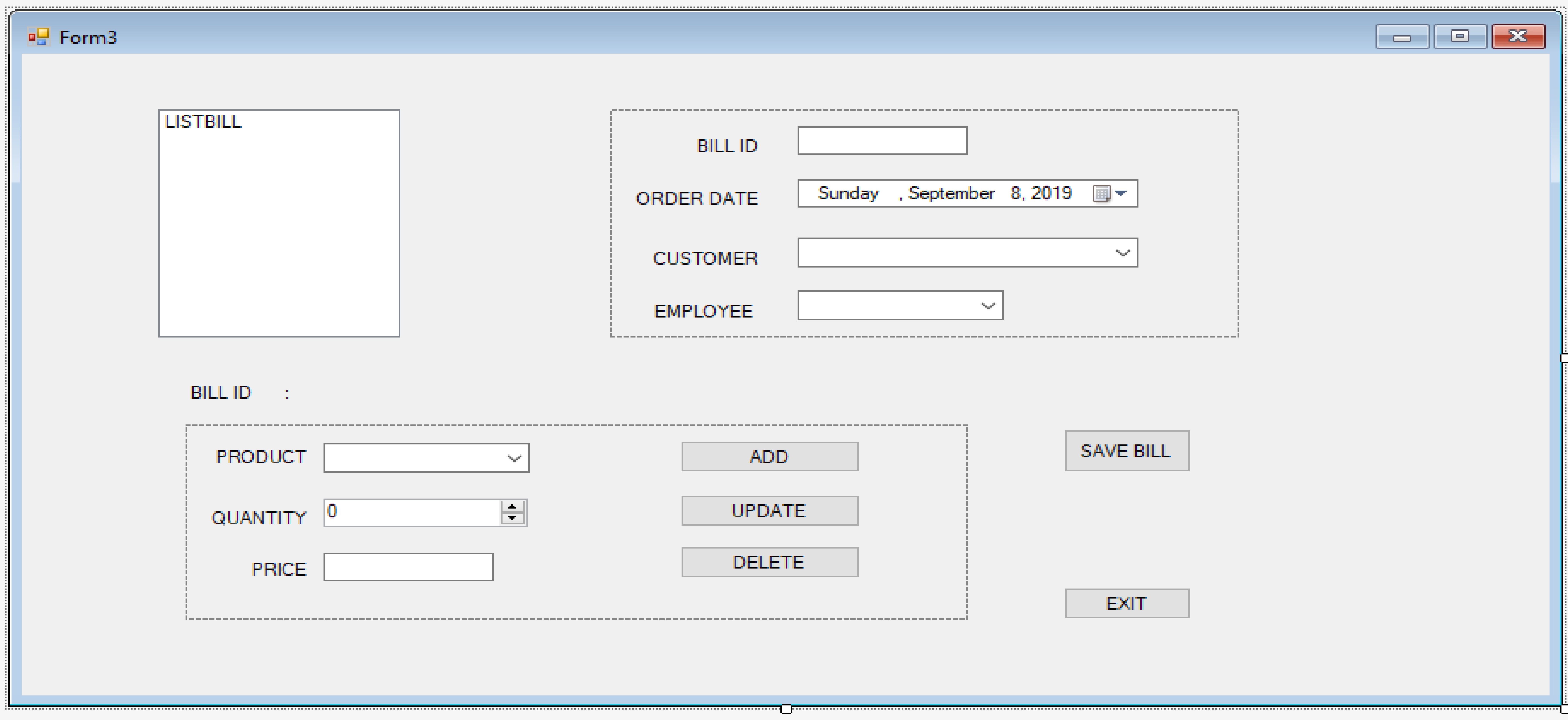
1. User interface of customer table



1. User interface of employee table

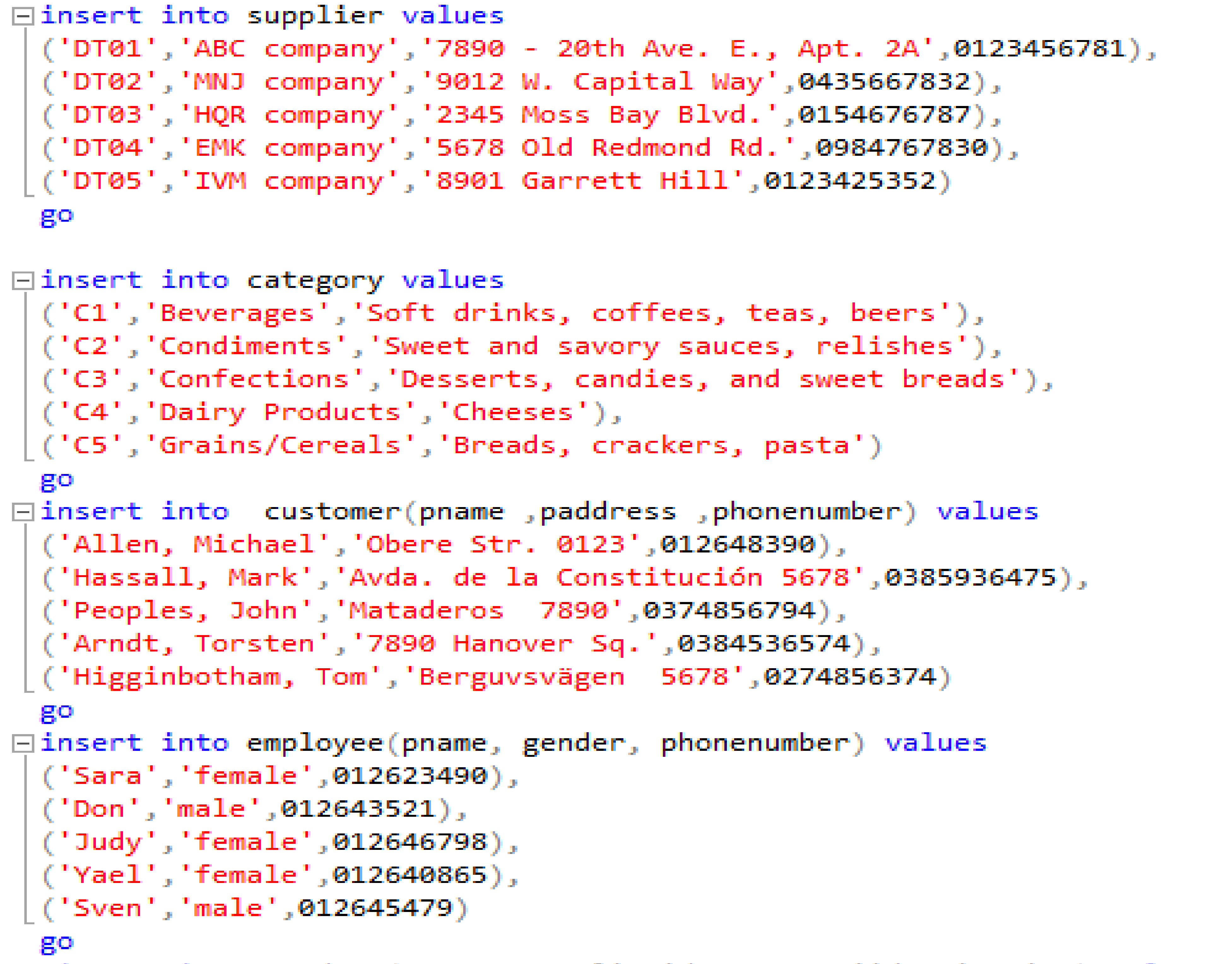


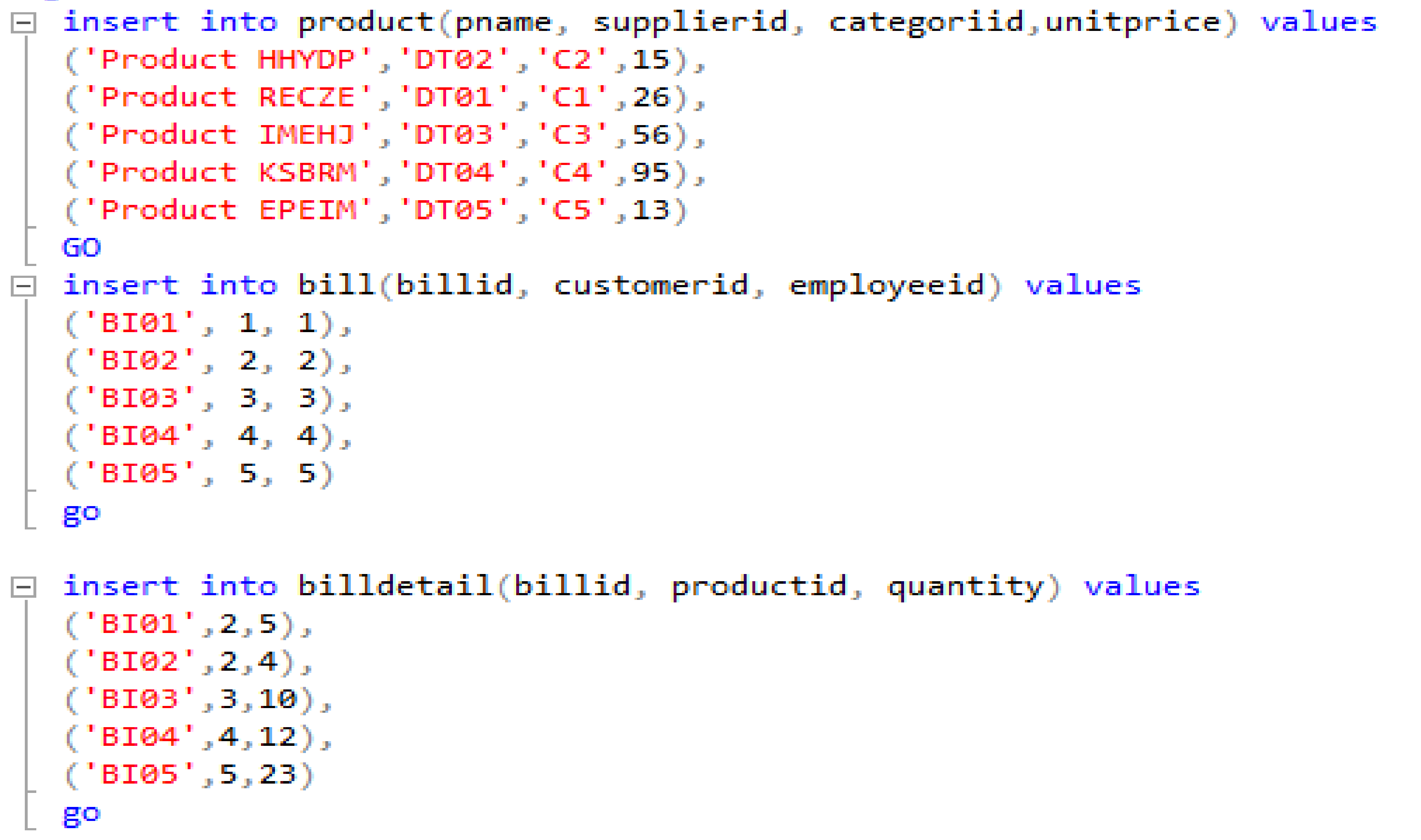
1. User interface of bill table



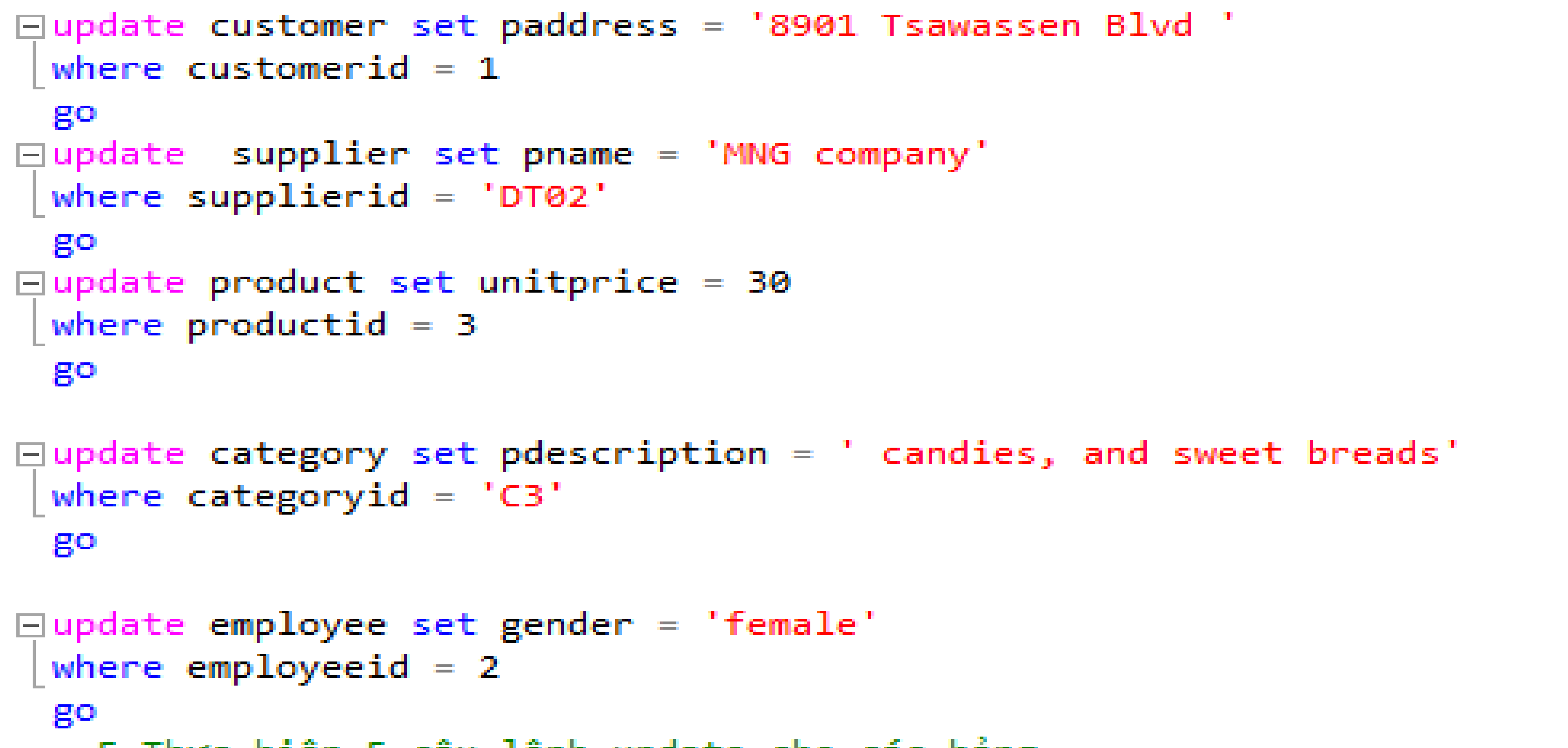
**P3** Implement a query language into the relational database system.

3.1 insert function:

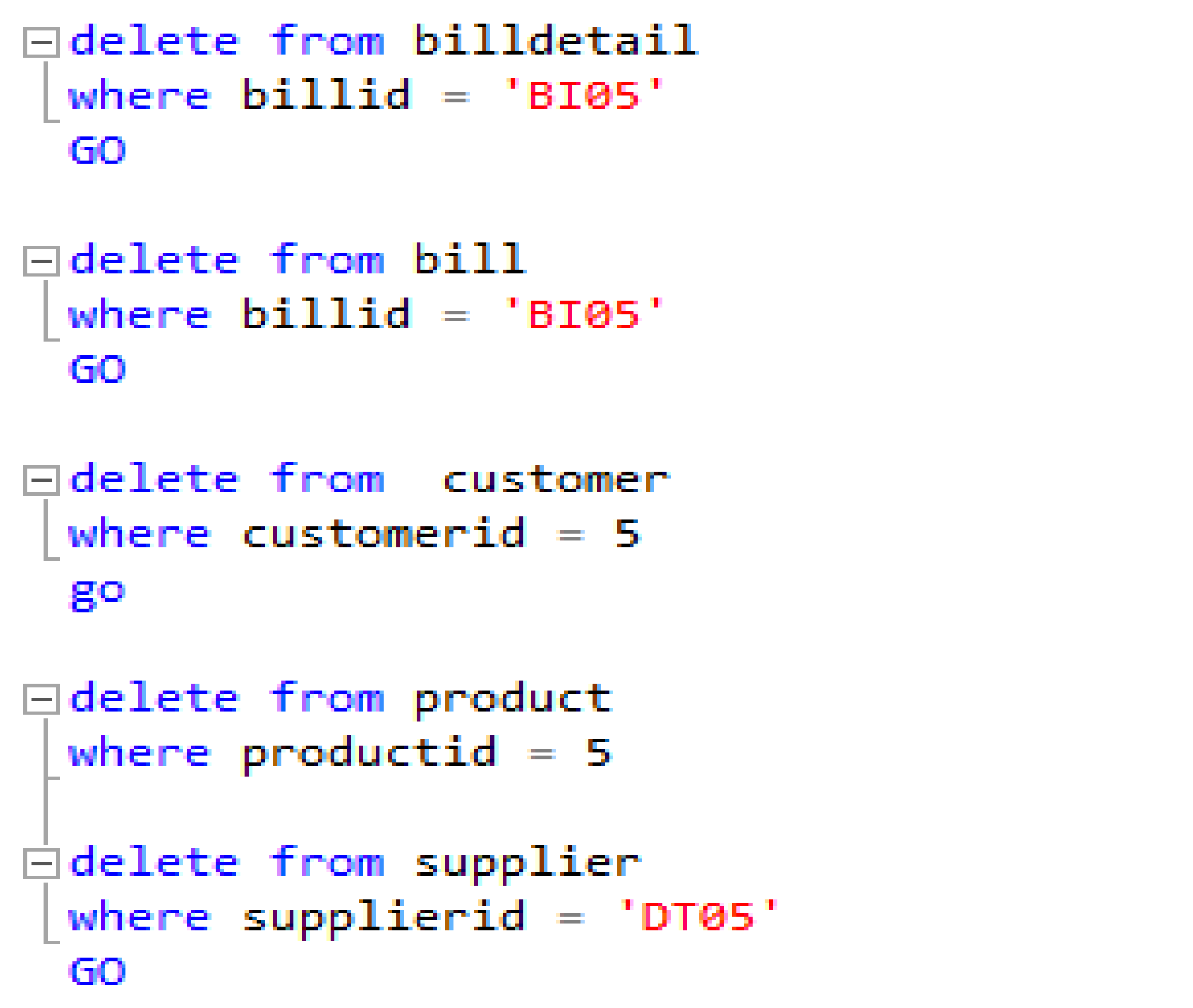




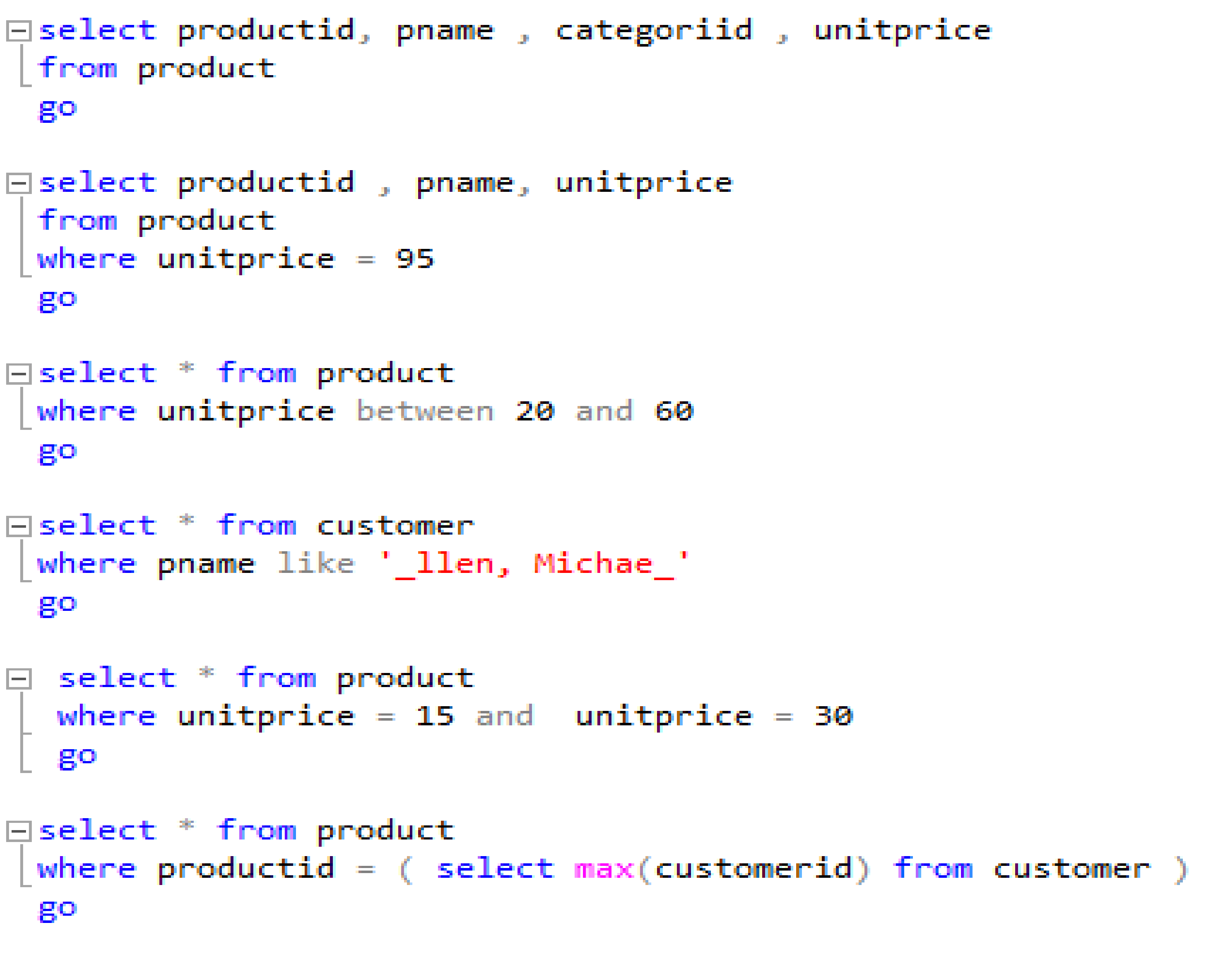
3.2 update function:



3.3 delete function:

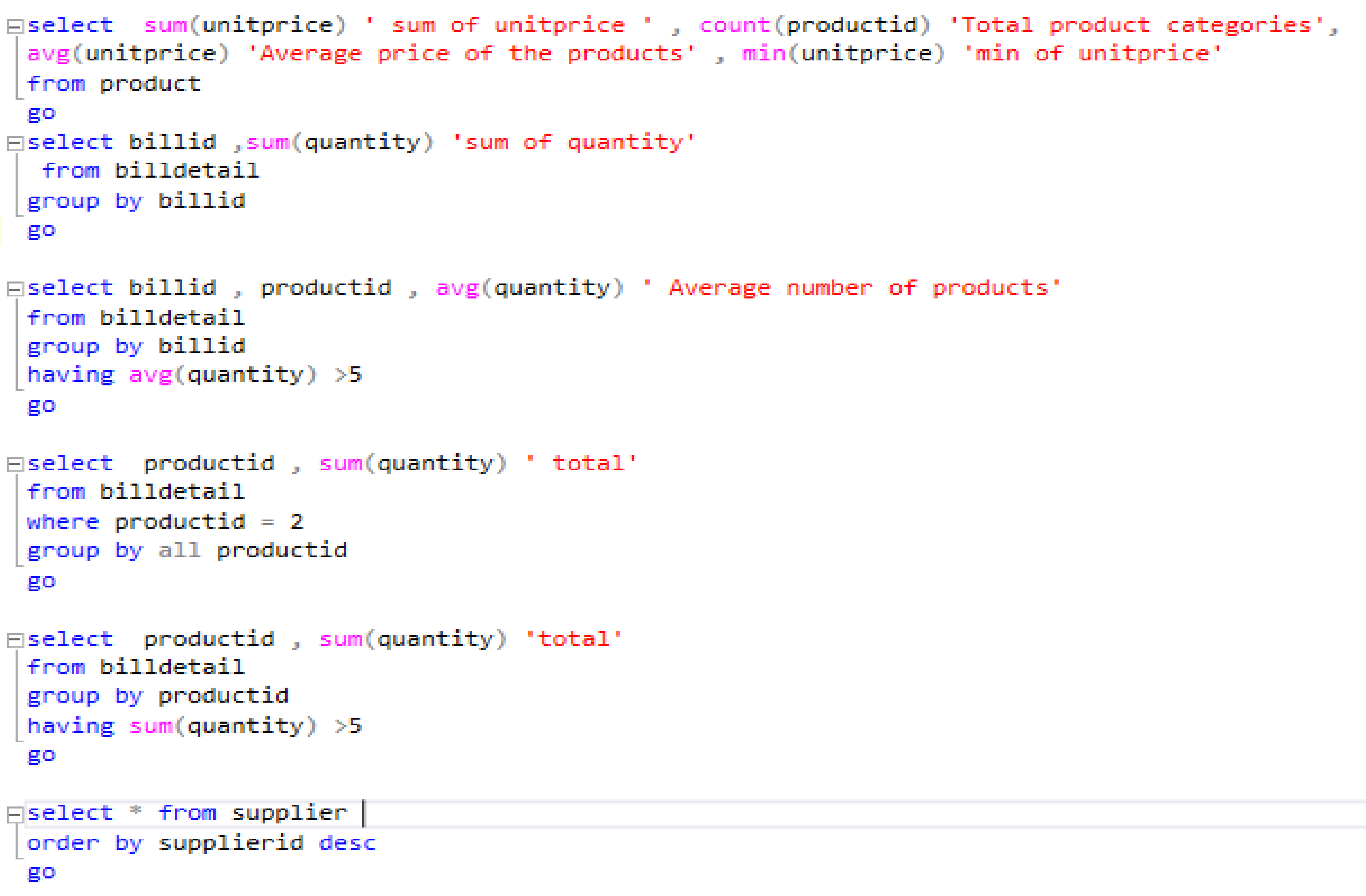


3.4 select fonction:

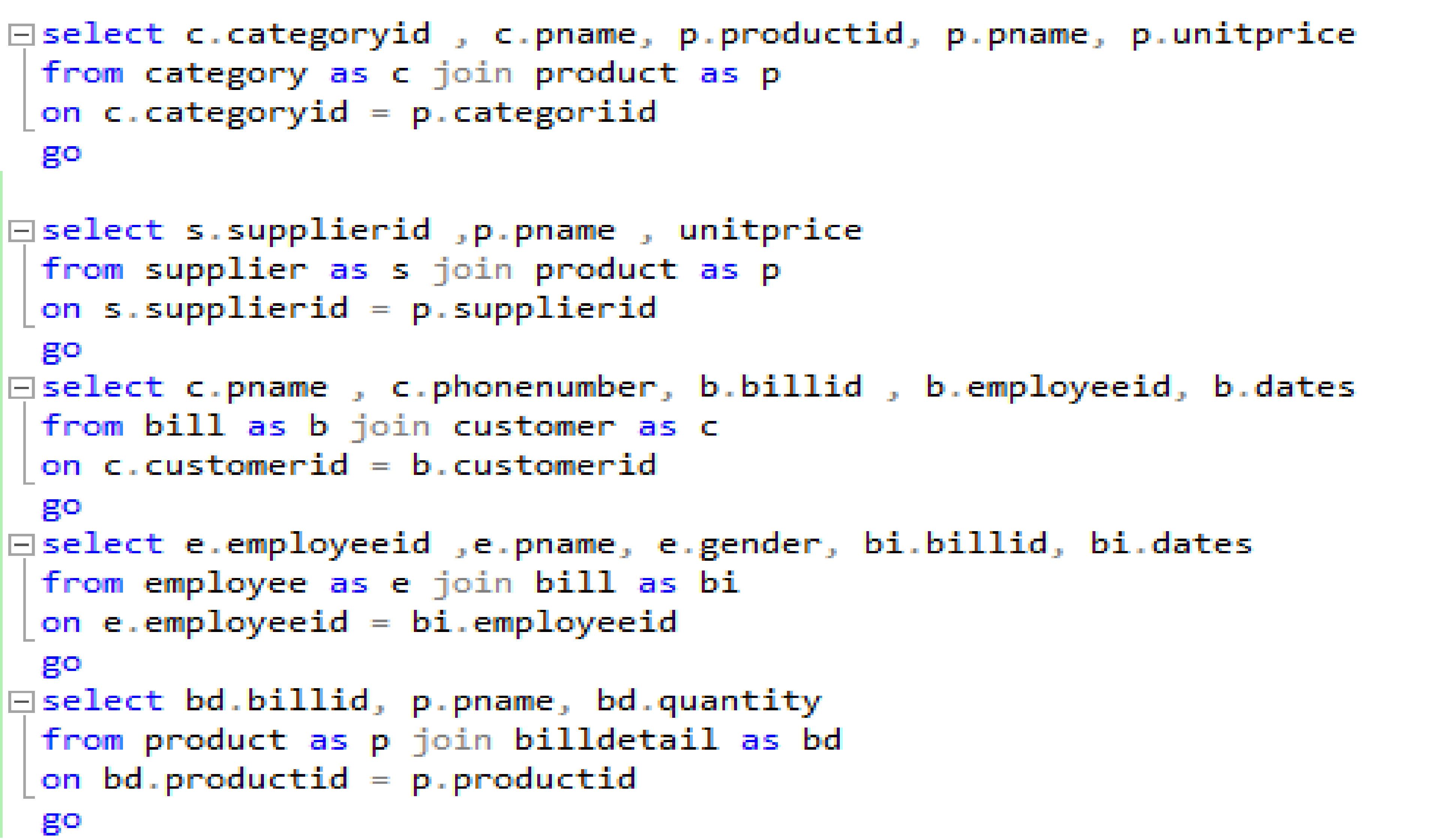


3.5 statement using a combination of COUNT (), SUM (), AVG (), MAX (), MIN ()

  Combined with group by, where, having, order by



3.6 advanced retrieval statements:



P4. Test the system against user and system requirements.

4.1 Test case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test** | **What is being tested** | **How** | **Test data used** | **Expected Results** |
| 1 | Input data into category table | Enter data from data capture sheet into the form on the data entry screen | category ID = C1  category Name = Beverages  description = Soft drinks, coffees, teas, beers | All information are insert into the category table successfully. |
| 2 | Input data into bill table | Enter data from data capture sheet into the form on the data entry screen | bill ID = BI01  Customer ID = ‘1’  Employee ID = ‘1’ | All information are insert into the bill table successfully, two foreign key Customer ID and Employee ID are insert successful too |
| 3 | Update employee gender at the row employee ID = 2 | Use the update syntax to insert the new value. | Gender = ‘female’  employee ID = ‘2’ | Employee phone number updated successfully |
| 4 | Update the unitprice at the row product ID= 3 | Use the update syntax to insert the new value | unitPrice = 30  Product ID = ‘3’ | The price of 3 has been increased from 56 to 84 |
| 5 | Delete the row which contain bill ID = BI05 | Use the delete syntax to delete the row which is chose. | bill ID = ‘DT05’ | The row which contain information of BILL ID = DT02 is deleted. |
| 6 | Delete the row which contain supplier ID = DT05 | Use the delete syntax to delete the row which is chose. | supplier ID = ‘DT05’ | The row which contain information of supplier ID = DT05 is deleted. |
| 7 | Find a specific information in table. | Use the select syntax to find the information in table | Product ID, product name, category ID , unitprice | All the ID ,Name and unitprice of the product table will be appeared. |
| 8 | Find a specific information of unitprice = 95 in product table | Use the select syntax to find the information in table | Product ID , product name, unitprice = 95 | All the information of product which has the unitprice is 95 will be appeared |
| 9 | Find the information the foreign key of the table | Use the select syntax to find the information in table. | Category ID =’C2’ | The information of category inside the product table which has ID is C2 will be appeared. |
| 10 | Find supplier have the same supplier ID = DT02, and Product ID, Product Name, unitprice will be showed | Use the select, join syntax to find the information in table. | product name, unitprice  Supplier ID | Theprouduct name, supplier  ID, unitprice which  have the same  supplier ID = DT02 will  be appeared. |

**Queries for test case**:

Q1

Insert into category Values

( Values 1 , Values 2, Values 3,…)

Go

Q2

Insert into [table name] (column 1, column 2, column 3, …)

Values ( Values 1 , Values 2, Values 3,…)

Go

Q3

update [table name]

set column1 = value1, column2 = value2, ...

where condition

Go

Q4

update [table name]

set column1 = value1, column2 = value2, ...

where condition

Go

Q5

Delete [Table name]

where condition

Go

Q6

Delete [Table name]

where condition

go

Q7

select \* from [Table Name]

where column = conditions

go

Q8

select \* from [Table Name]

where column = conditions

go

Q9

select \* from [Table Name]

where column = conditions

go

Q10

select column1, column2,column3,…

from table\_name join table\_name

on table1.column\_name = table2.column\_name

Where table1.column\_name=( select column\_name from table\_name where column= condition)

go

4.2 Test logs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test** | **What is**  **being tested** | **How** | **Test data used** | **Expected Results** | **Actual results** | **Action taken** |
| 1 | Input data into category table | Enter data from data capture sheet into the form on the data entry screen | category ID = C1  category Name = Beverages  description = Soft drinks, coffees, teas, beers | All information are insert into the category table successfully. | ok | none |
| 2 | Input data into bill table | Enter data from data capture sheet into the form on the data entry screen | bill ID = BI01  Customer ID = ‘1’  Employee ID = ‘1’ | All information are insert into the bill table successfully, two foreign key Customer ID and Employee ID are insert successful too | ok | none |
| 3 | Update employee gender at the row employee ID = 2 | Use the update syntax to insert the new value. | Gender = ‘female’  employee ID = ‘2’ | Employee phone number updated successfully | ok | none |
| 4 | Update the unitprice at the row product ID= 3 | Use the update syntax to insert the new value | unitPrice = 30  Product ID = ‘3’ | The price of 3 has been increased from 56 to 84 | ok | none |
| 5 | Delete the row which contain bill ID = BI05 | Use the delete syntax to delete the row which is chose. | bill ID = ‘DT05’ | The row which contain information of BILL ID = DT02 is deleted. | ok | none |
| 6 | Delete the row which contain supplier ID = DT05 | Use the delete syntax to delete the row which is chose. | supplier ID = ‘DT05’ | The row which contain information of supplier ID = DT05 is deleted. | ok | none |
| 7 | Find a specific information in table. | Use the select syntax to find the information in table | Product ID, product name, category ID , unitprice | All the ID ,Name and unitprice of the product table will be appeared. | ok | none |
| 8 | Find a specific information of unitprice = 95 in product table | Use the select syntax to find the information in table | Product ID , product name, unitprice = 95 | All the information of product which has the unitprice is 95 will be appeared | ok | none |
| 9 | Find the information the foreign key of the table | Use the select syntax to find the information in table. | Category ID =’C2’ | The information of category inside the product table which has ID is C2 will be appeared. | ok | none |
| 10 | Find supplier have the same supplier ID = DT02, and Product ID, Product Name, unitprice will be showed | Use the select, join syntax to find the information in table. | product name, unitprice  Supplier ID | Theprouduct name, supplier  ID, unitprice which  have the same  supplier ID = DT02 will  be appeared. | ok | none |

**Queries for test log:**

**Q1**

insert into category values

('C1','Beverages','Soft drinks, coffees, teas, beers')

Go

**Q2**

insert into bill(billid, customerid, employeeid) values

('BI01', 1, 1)

Go

**Q3**

update employee set gender = 'female'

where employeeid = 2

Go

**Q4**

update product set unitprice = 30

where productid = 3

Go

**Q5**

delete from bill

where billid = 'BI05'

GO

**Q6**

delete from supplier

where supplierid = 'DT05'

GO

**Q7**

select productid, pname , categoriid , unitprice

from product

Go

**Q8**

select productid , pname, unitprice

from product

where unitprice = 95

Go

**Q9**

select \* from product

where categoriid = 'C2'

Go

**Q10**

select s.supplierid ,p.pname , unitprice

from supplier as s join product as p

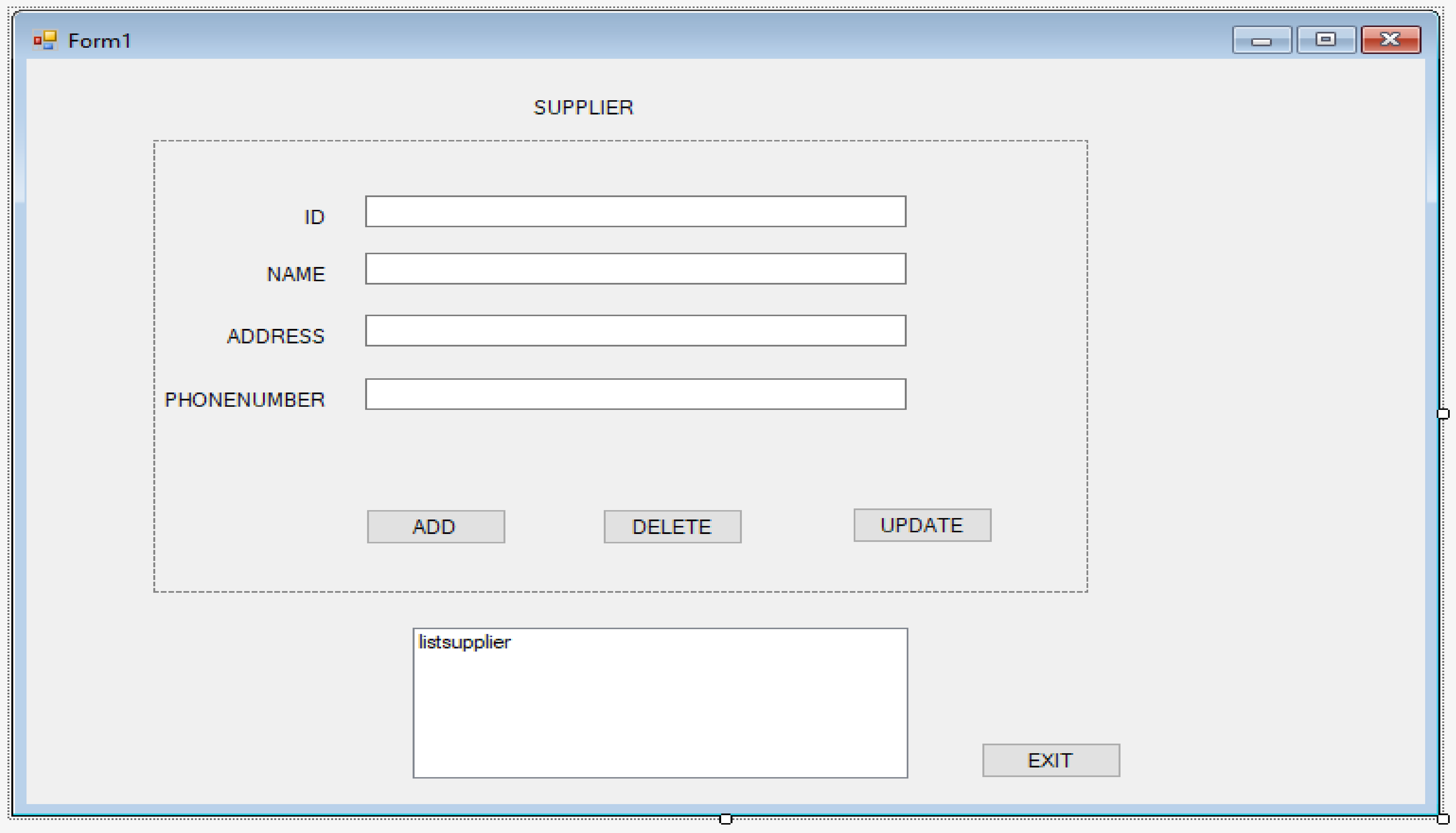
on s.supplierid = p.supplierid

where s.supplierid =( select supplierid from supplier where supplierid ='DT02')

Go

**P5. Produce technical and user documentation**

5.1 User document:

1. Manage Suppliers 
2. **Add Supplier**

Step 1: Input information.

Step 2: Click Add button.The supplierr's name will then appear on the supplierr list.

1. **Update supplier**

Step 1: Click name of supplier which you want to update.

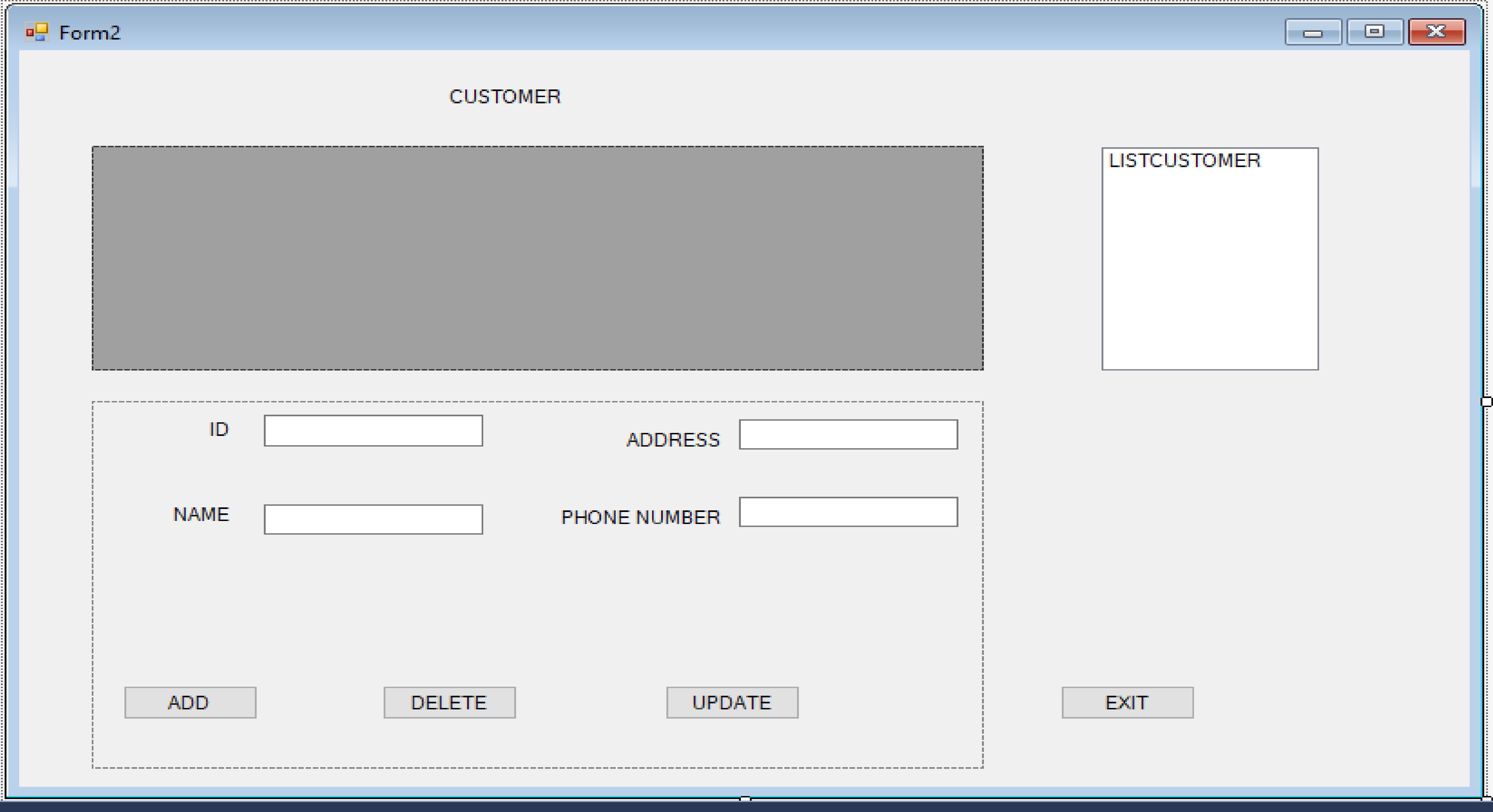
Step 2: Choose what information you want to change then click update.

1. **Delete supplier**

Step 1: Choose name of the supplier which you want to delete.

Step 2: Click delete button, all information of that supplier will be deleted.

1. Manage Customer



1. **Add new Customer**

Step 1: Input information.

Step 2: Click Add button.The customer's name will then appear in the customer list

1. **Update Customer**

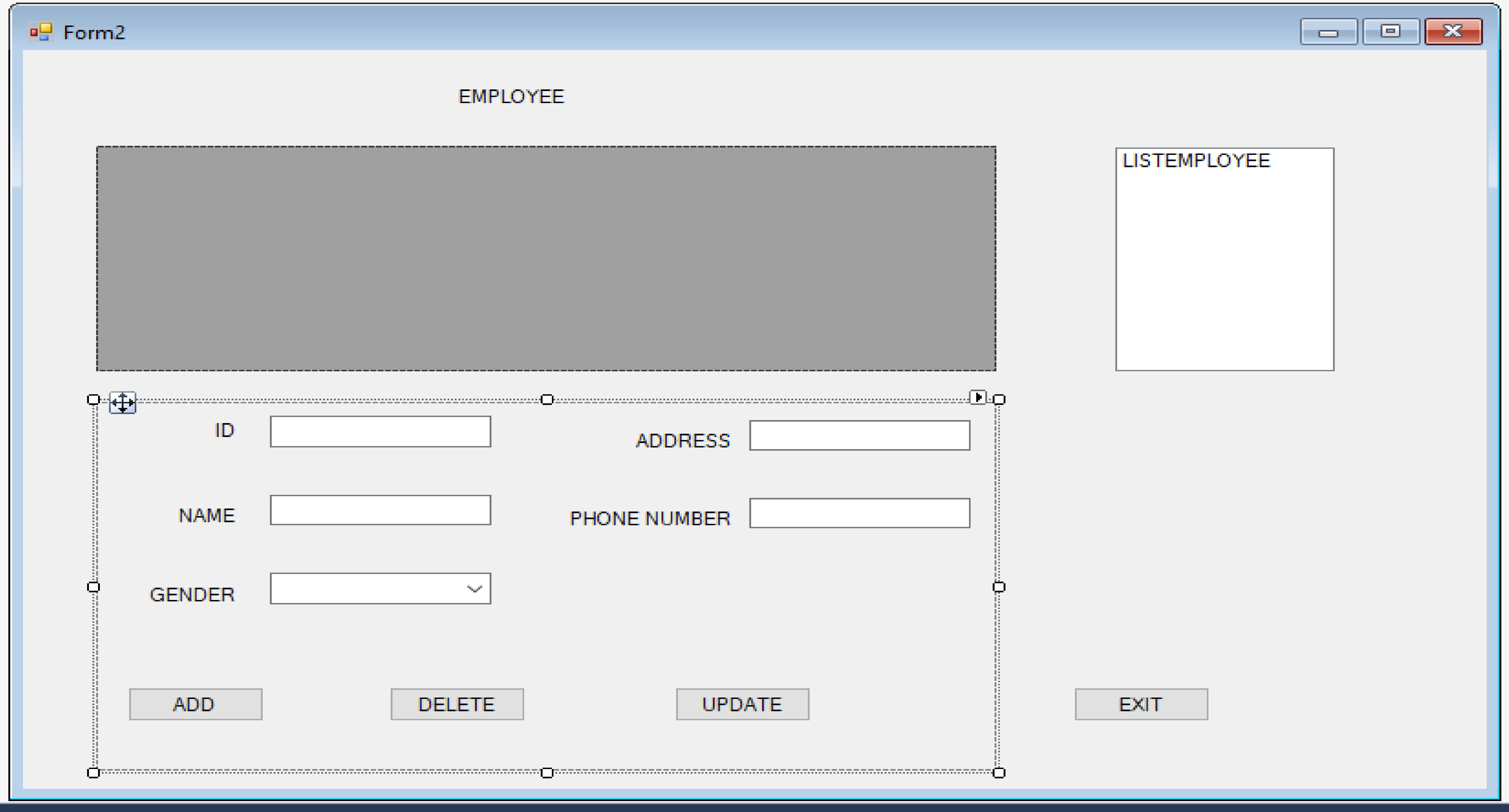
Step 1: Click name of supplier which you want to update.

Step 2: Choose what information you want to change then click update.

1. **Delete Customer**

Step 1: Choose name of the supplier which you want to delete.

Step 2: Click delete button, all information of that supplier will be deleted.

1. Manage Employee 
2. **Add Employee**

Step 1: Input information.

Step 2: Click Add button.The employee's name will then appear in the employee list

1. **Update Employee**

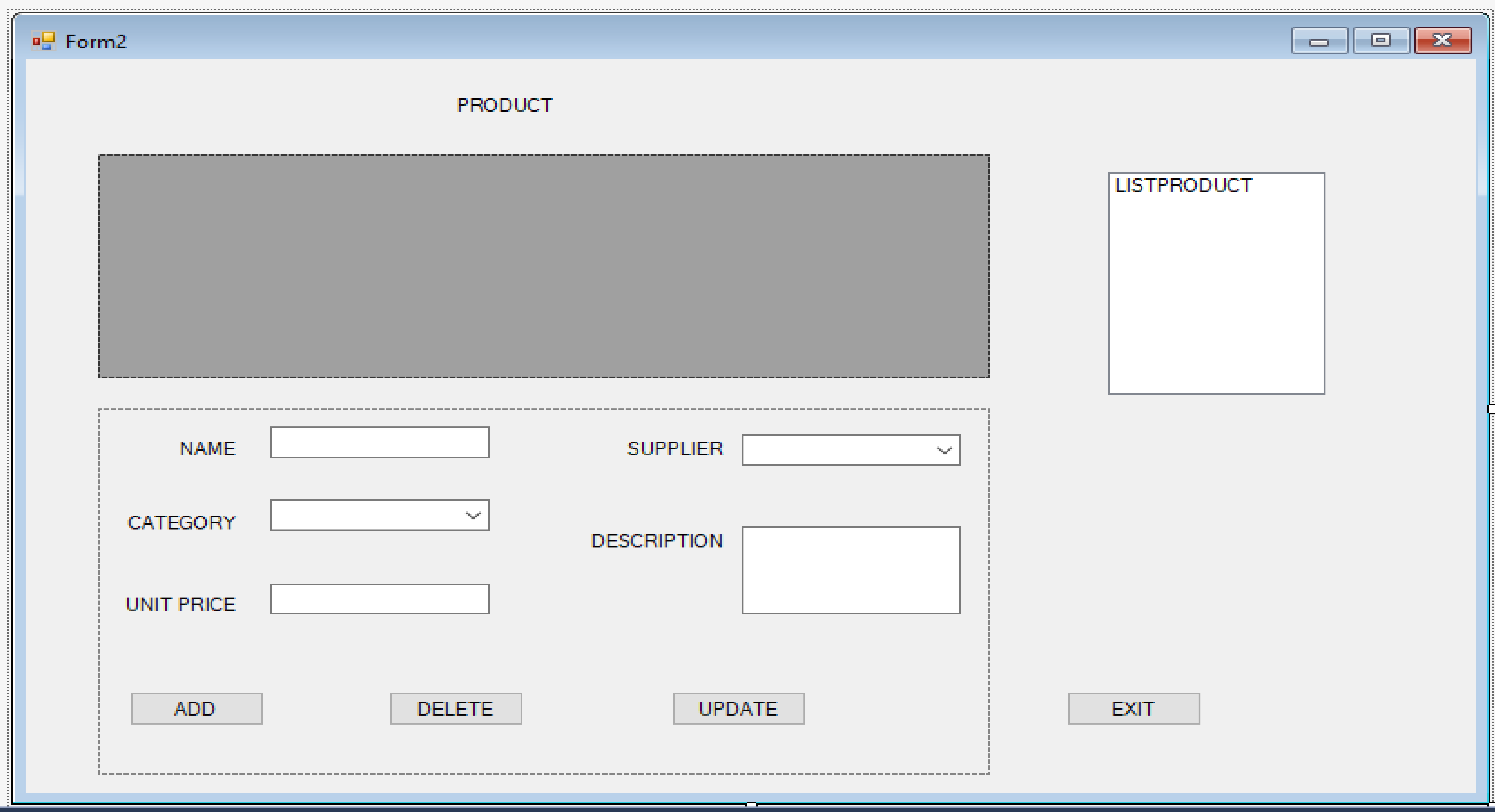
Step 1: Click name of supplier which you want to update.

Step 2: Choose what information you want to change then click update.

1. **Delete Employee**

Step 1: Choose name of the supplier which you want to delete.

Step 2: Click delete button, all information of that supplier will be deleted.

1. Manage Product 
2. **Add Product**

Step 1: Input information.

Step 2: Click Add button.The productr's name will then appear in the product list

1. **Update Product**

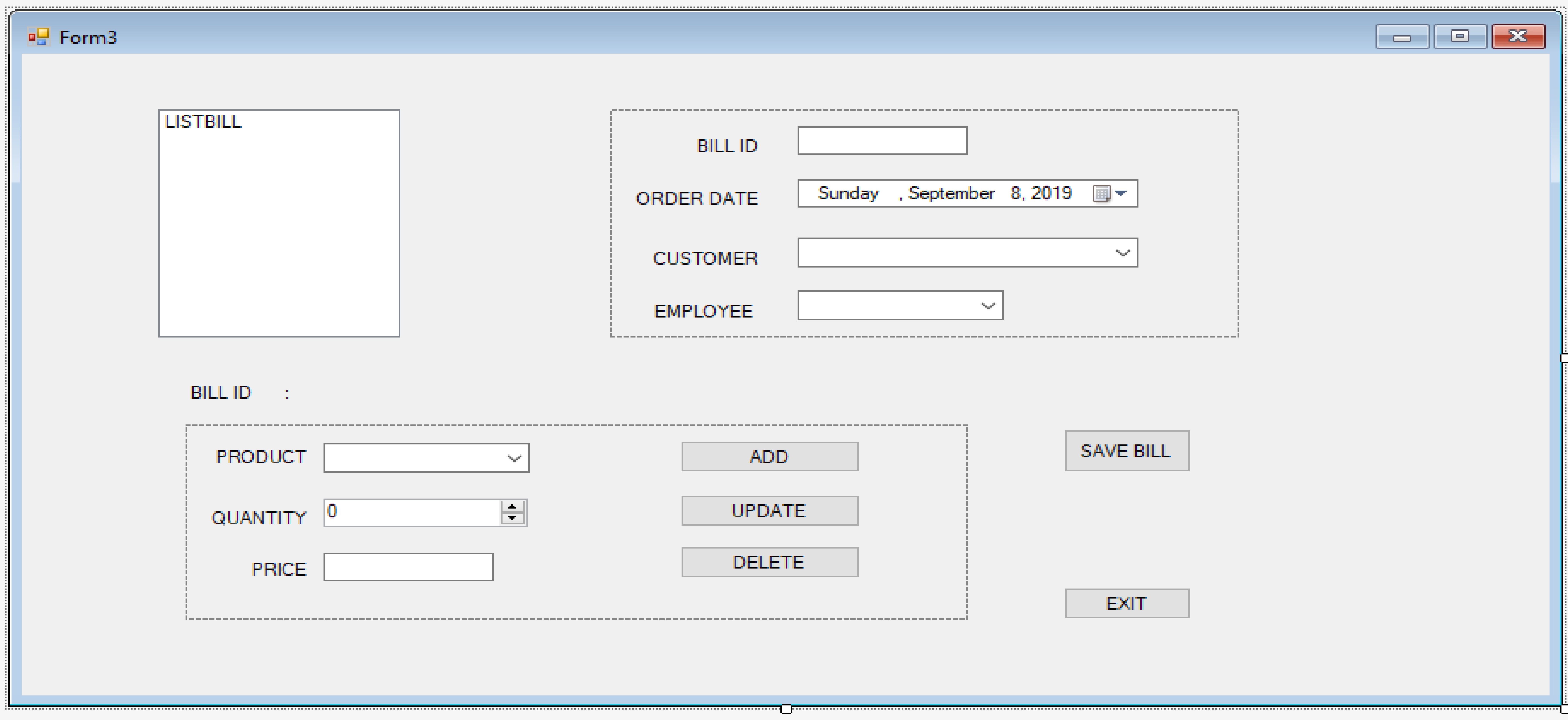
Step 1: Click name of supplier which you want to update.

Step 2: Choose what information you want to change then click update.

1. **Delete Product**

Step 1: Choose name of the supplier which you want to delete.

Step 2: Click delete button, all information of that supplier will be deleted.

1. Manage bill: 
2. **Add bill**

Step 1: Input information.

Step 2: Click Add button.

1. **Update bill**

Step 1: Click name of supplier which you want to update.

Step 2: Choose what information you want to change then click update.

1. **Delete bill**

Step 1: Choose name of the supplier which you want to delete.

Step 2: Click delete button, all information of that supplier will be deleted.

1. **Save bill**

Click save bill when finished input information if you want to save the bill .

**5.2 Technical documentation**

**A) Hardware requirement**:

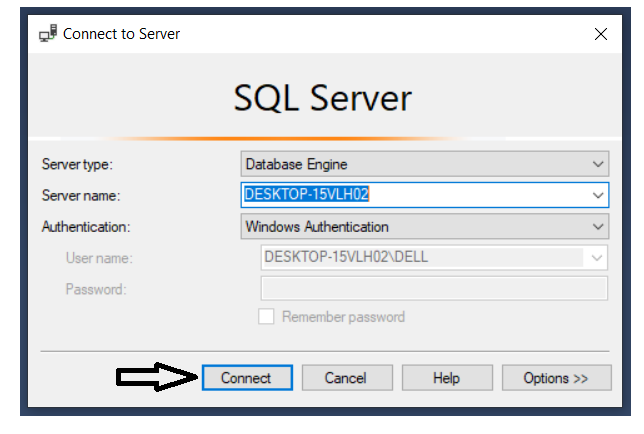
* Minimum of 6 GB of available hard-disk space.
* Super-VGA (800x600) or higher resolution monitor.
* CPU core i5, 2.0GHz , RAM 4GB, HDD/SSD 256GB…
* Clocks peed of 2 GHz or more. 1.4 GHz minimum.
* 64 bit x64-compatible AMD or Intel CPU only.

**Software requirement:**

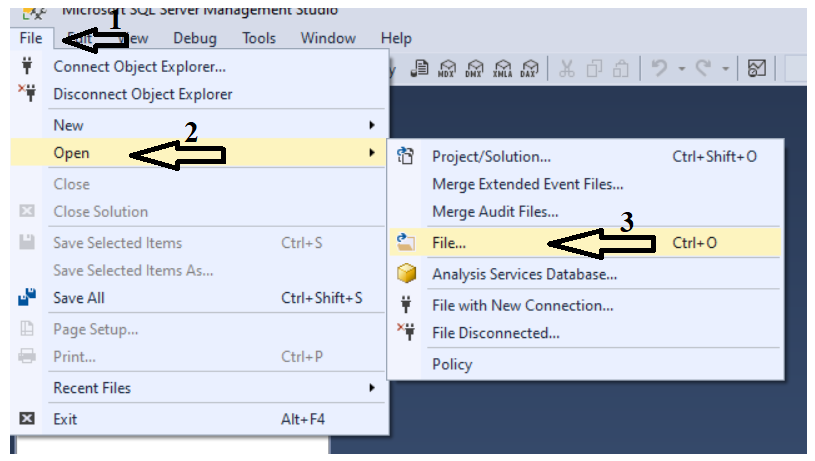
* Visual studio 2017 for design UI or/ HTML, CSS for design UI.
* SQL server core engine is 64-bit only and does not support the 32-bit editions of windows 8 or window 10.
* SQL server need hard disk space of 8030 MB.
* Microsoft SQL Server 2017

1. **Instruction for technical users:**

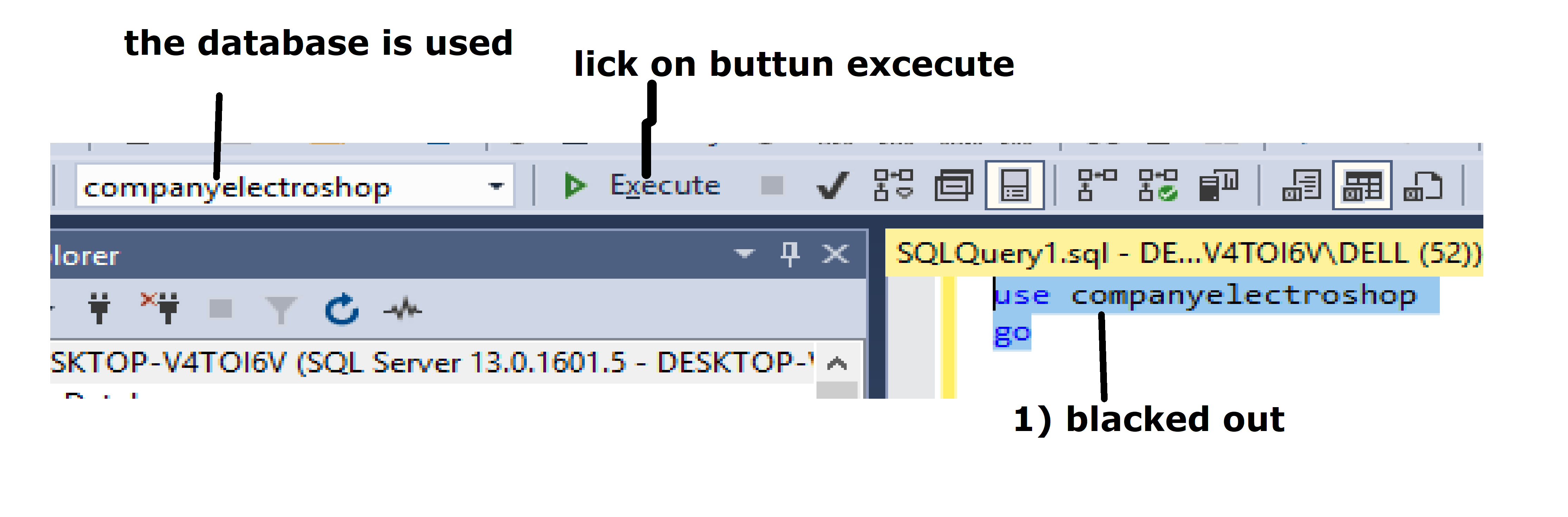
* When you access to SQL management server, this table will appear and you have to choose server name where is your database is stored. When you have chose, click connect button.



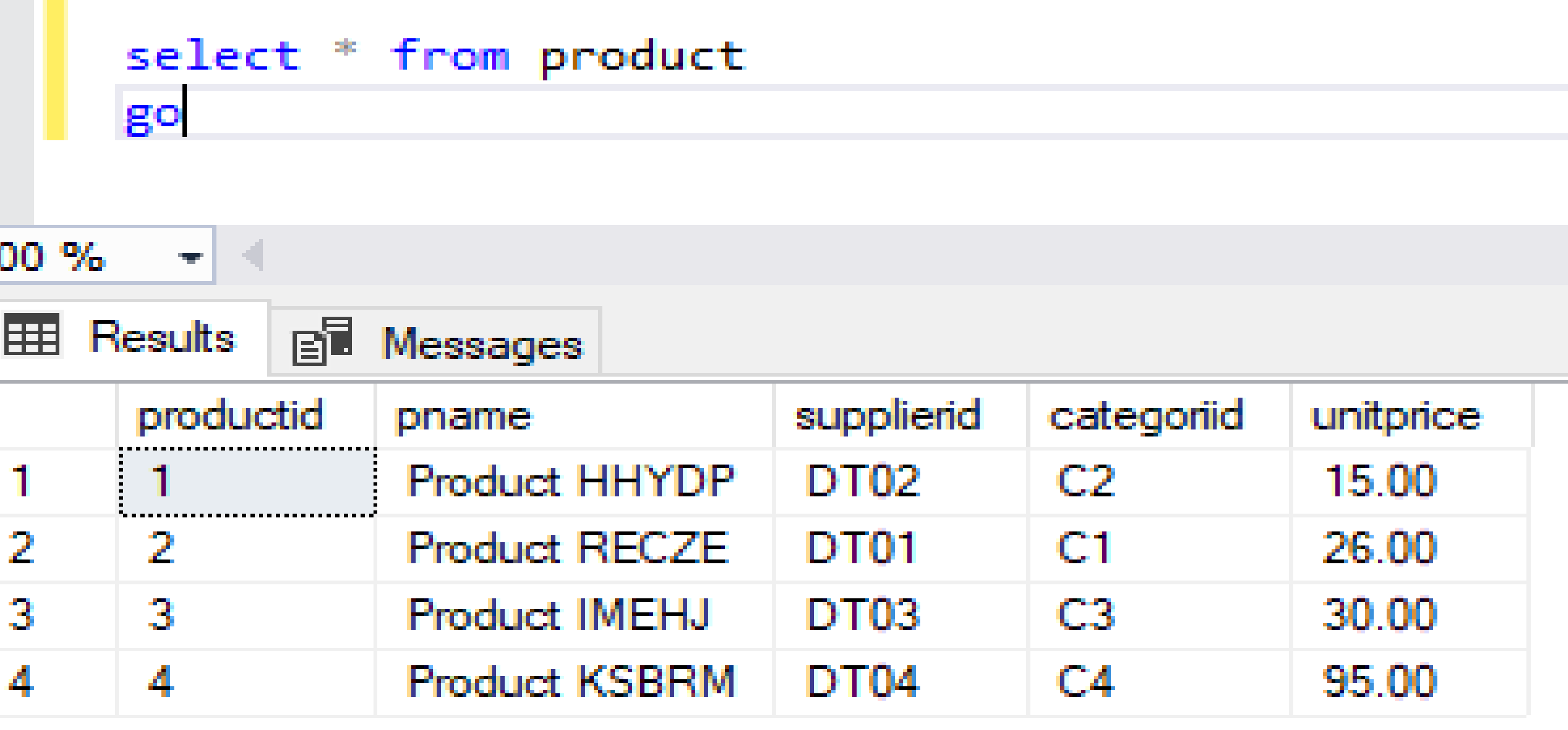
* Next you need to open an available file in your computer. Follow these step and you will see this table appear: File -> Open -> File, or you can use Ctrl + O. Then you can choose the database you want.



* The next step you select these rows contains your database name and select execute. You will then see the text owner change to your database name.



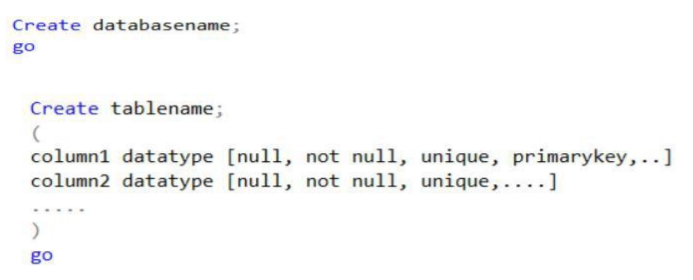
* To run a query in SQL management server, you just do the same things as above steps. Example: I want to see what I inserted into customer table. I will use select syntax



* **Using Code to create database in SQL Management Server.**

some basic query in SQL Management Server: create database, create table, insert, update, delete, select.

* Create Database and create Table

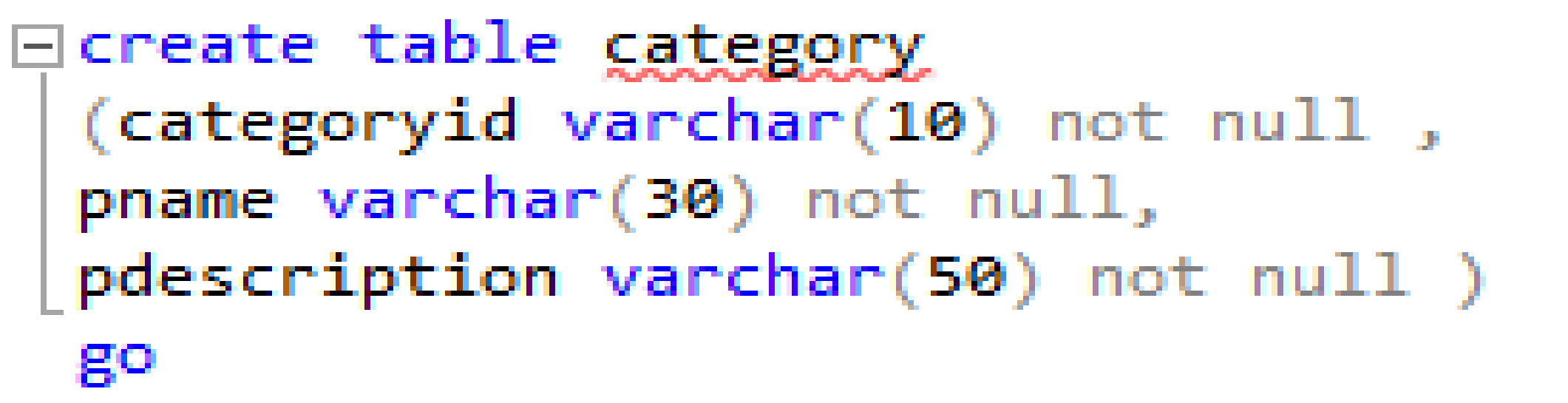


**- Example:**

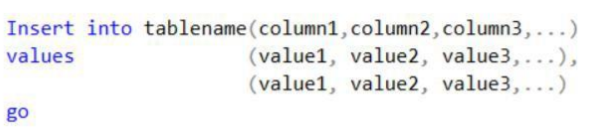
**Create Database**



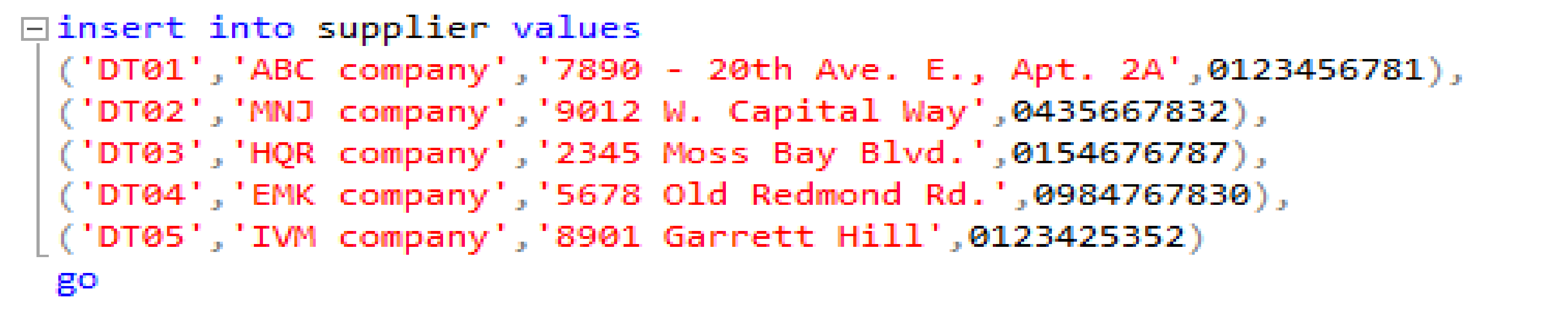
**Create Table**



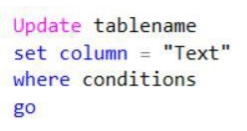
* **Insert data into table**



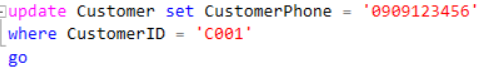
**- Example:**



* Update syntax



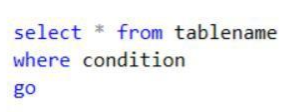
**- Example:**



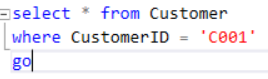
* **Select data of whole table**



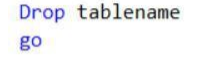
* **Select 1 or 2 or many data from table**



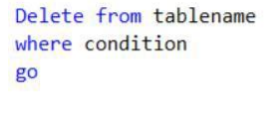
Example:



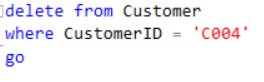
* **Delete Whole Table**



* **Delete 1 or 2 or many data from table**



**- Example:**

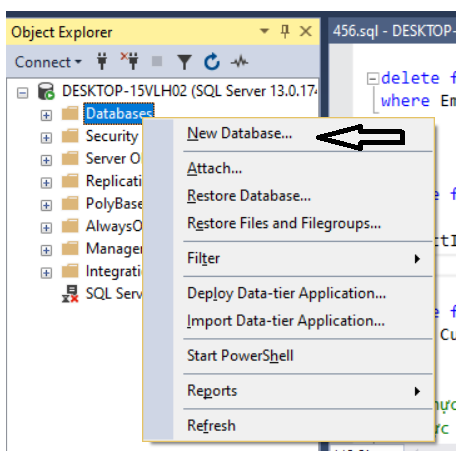


* **Using Tools to create database in SQL Management Server**

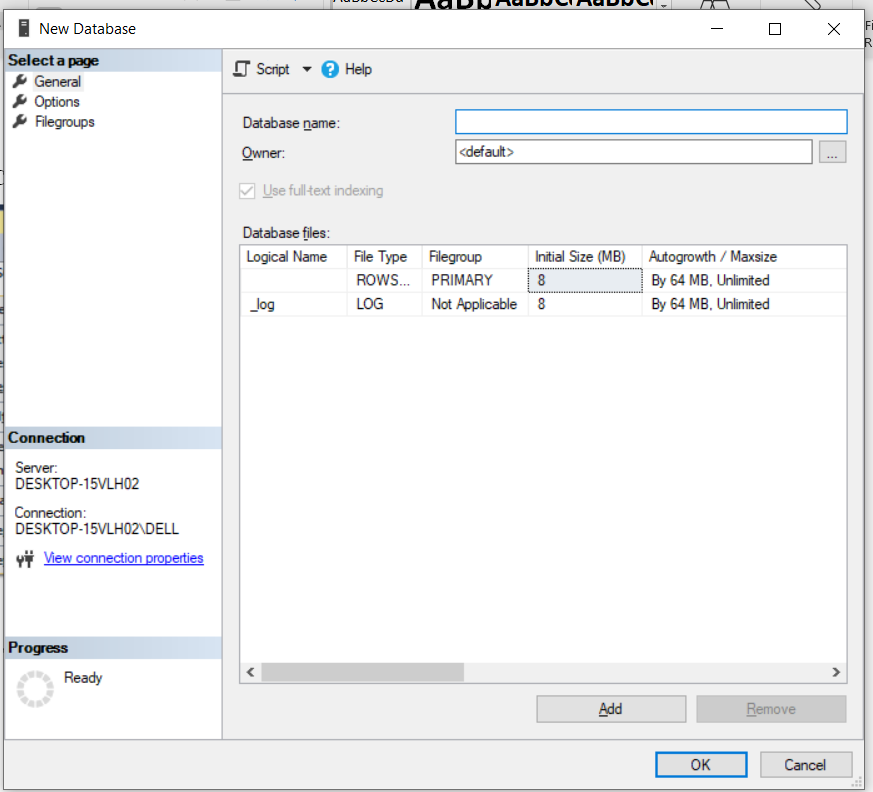
If you don’t want to create your database with coding, you can create it with tools in SQL Management Server. How to do that? Follow all steps below to do it.

* **Create database**

First right-click on databases then you choose new database.

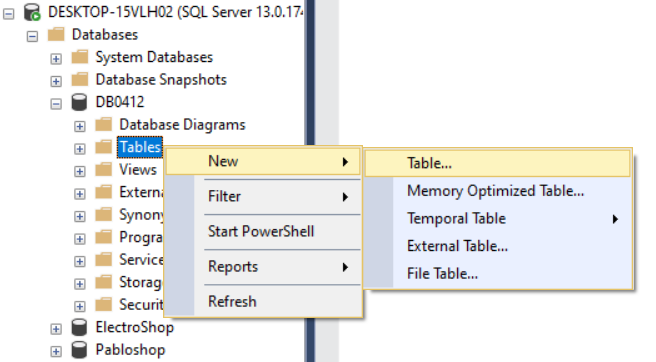


Type your database name inside the database name text box. Then press OK

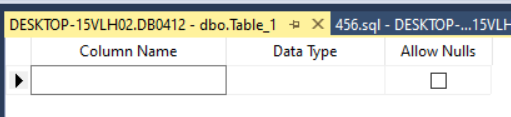


* **Create Table**

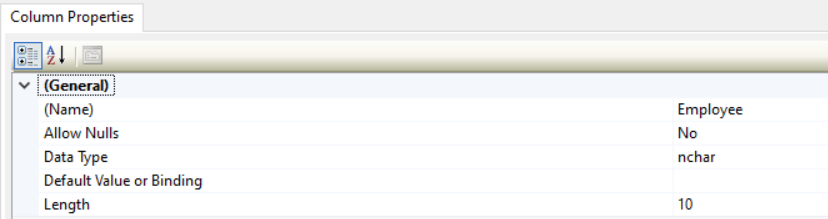
Left-click on database. Then right-click on table.



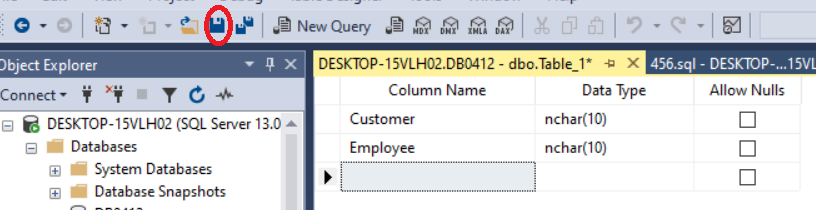
This page will appear to insert data, First column is your column name, second column is your data types for that column, last column you have to tick allow nulls or not.



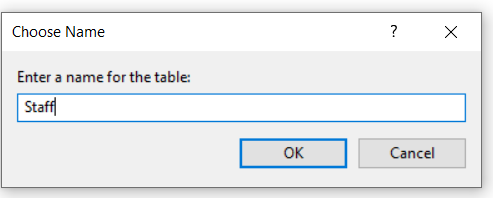
If you insert data into that table, at the of the page, you will see your column properties.



Then click save button, or you can use Ctrl + S



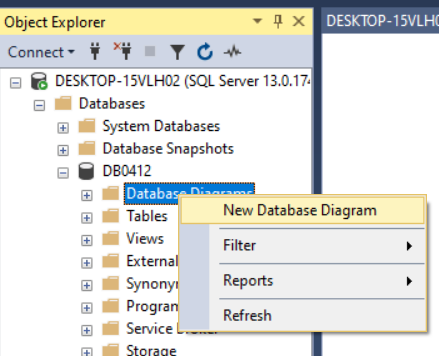
Then choose a name for your table, and click OK



* **Create database diagrams**

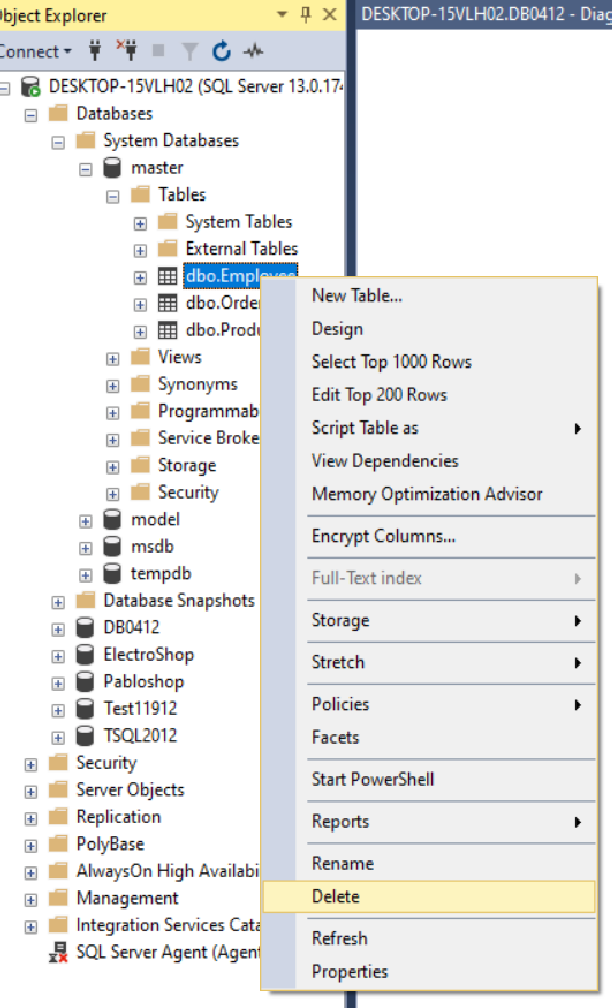
Next is relationship between 2 table. Before making a relationship, you must ensure that you have. Primary key and foreign key.

Left-click on the plus icon near your database, then right-click on Database Diagrams and choose New Database Diagram. And then you click add, so you have a diagram.



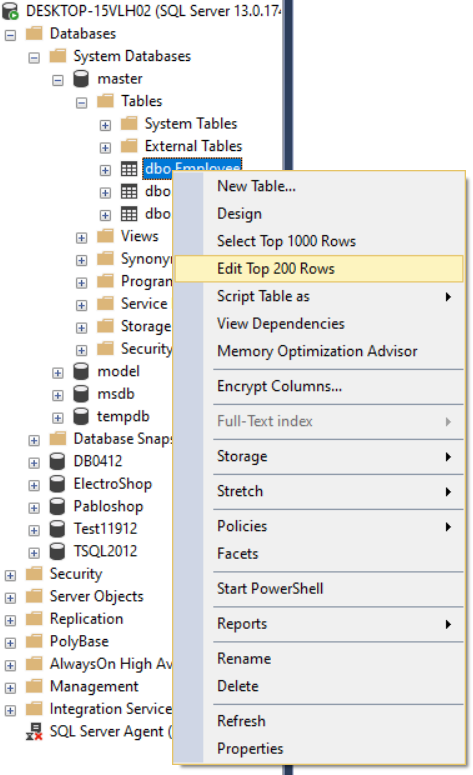
* **Delete table**

Left-click on plus icon near databases, left-click again on plus icon near your database name, left-click on plus icon near your table you need to delete. Right-click on that table, then choose delete.



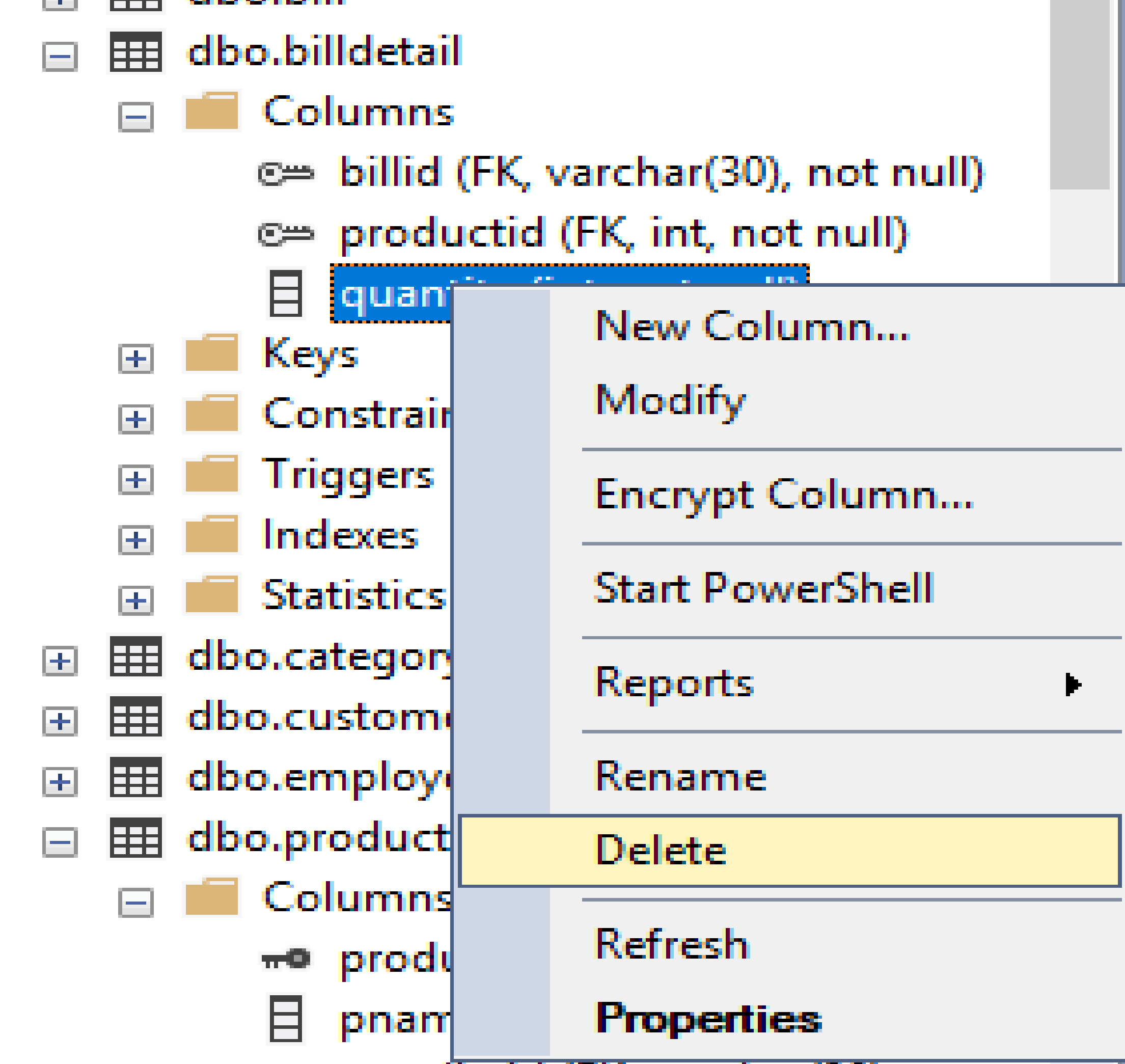
* **Insert information into table.**

You choose that table and then right-click on that table choose edit top 200 rows. Then you can insert information into that table.



* **Delete one column of table.**

Choose what column you want to delete then click the icon on the left then choose delete.



**REFERENCE**

Medium. (2019). *Database Normalization Explained*. [online] Available at: https://towardsdatascience.com/database-normalization-explained-53e60a494495 [Accessed 8 Sep. 2019].

**<ATTACHED EVIDENCE>**