**Charotar University of Science and Technology**

**Devang Patel Institute of Advance Technology and Research**

**Department of Computer Engineering**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Student ID** | **:** | **18DCE115** | **Student Name** | **:** | **KASHYAP SHAH** |
| **Subject Code** | **:** | **CE246** | **Subject Name** | **:** | **DATABASE MANAGEMENT SYSTEM** |
| **Date of exam** | **:** | **29-05-2020** | **Semester** | **:** | **4th Semester** |

**Definition:**

There is a warehouse of AMart. Create a table to manage the items available in AMart warehouse

Create a table: Warehouse\_data

Table: warehouse\_data

Field Name Data type

Item\_id Varchar2(10)

Item\_name Varchar2(20)

Price Number(8,2)

Insert minimum 5 records.

Create a field called Quantity. Take the data of quantity from user and store it into the table.

When a new item imported, check whether the item is available in the given table or not. If it’s

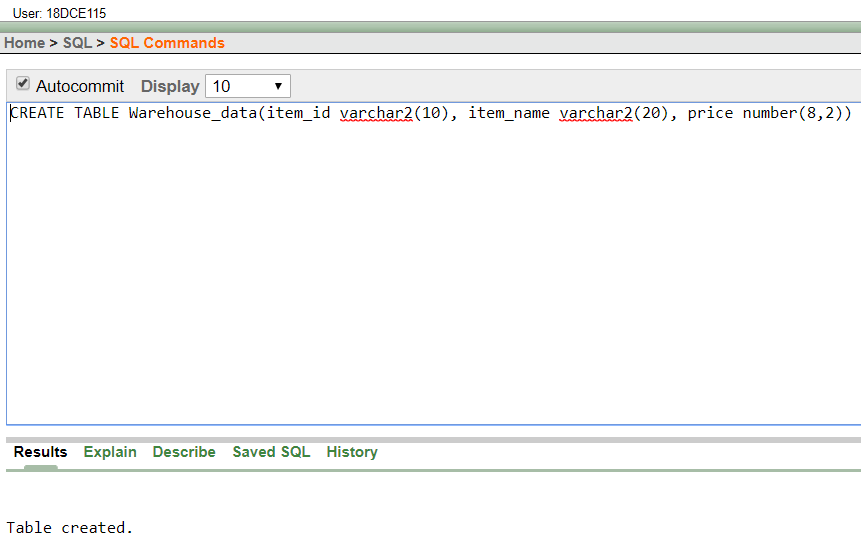
available, then update the quantity and if the item is not available then insert a new record into

warehouse\_data table.

At last, create a function to display the total price.

**Solution (code & screenshot):**

CREATE TABLE Warehouse\_data(item\_id varchar2(10), item\_name varchar2(20), price number(8,2))



INSERT INTO Warehouse\_data VALUES(1, 'TV', 1000)

INSERT INTO Warehouse\_data VALUES(2, 'Laptop', 2500)

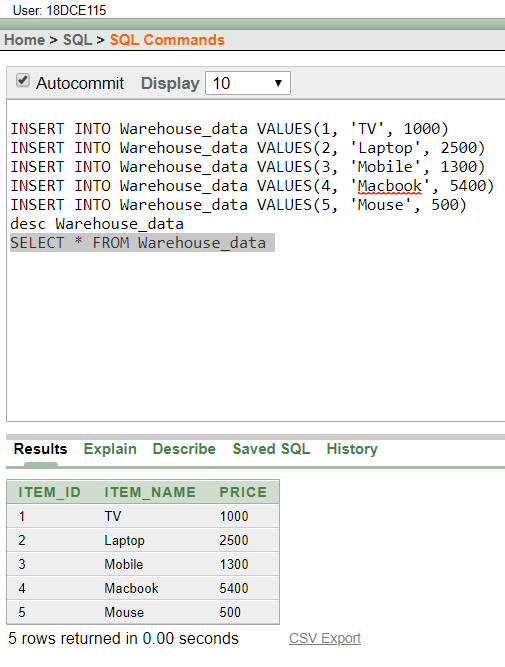
INSERT INTO Warehouse\_data VALUES(3, 'Mobile', 1300)

INSERT INTO Warehouse\_data VALUES(4, 'Macbook', 5400)

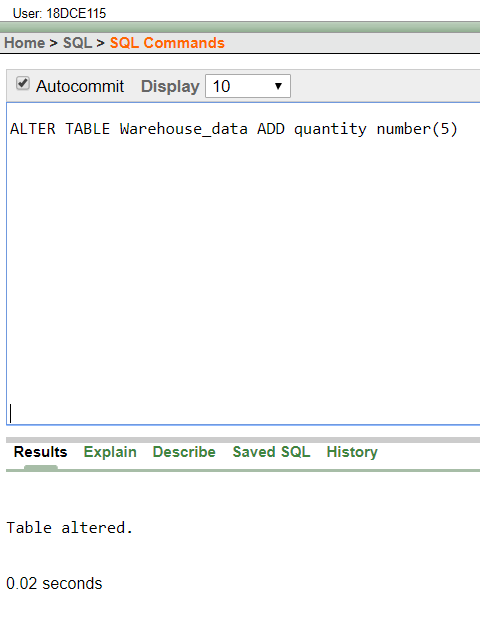
INSERT INTO Warehouse\_data VALUES(5, 'Mouse', 500)

desc Warehouse\_data

SELECT \* FROM Warehouse\_data



ALTER TABLE Warehouse\_data ADD quantity number(5)



Create or replace function add\_quan(id varchar2, quantity2 number) return number

Is

Begin

IF id<6 AND id>0 THEN

UPDATE Warehouse\_data SET quantity=quantity2 WHERE item\_id=id;

Return(quantity2);

ELSE

dbms\_output.put\_line('Does not exist');

Return(quantity2);

END IF;

End;

declare

quantity2 number;

i number;

q number;

begin

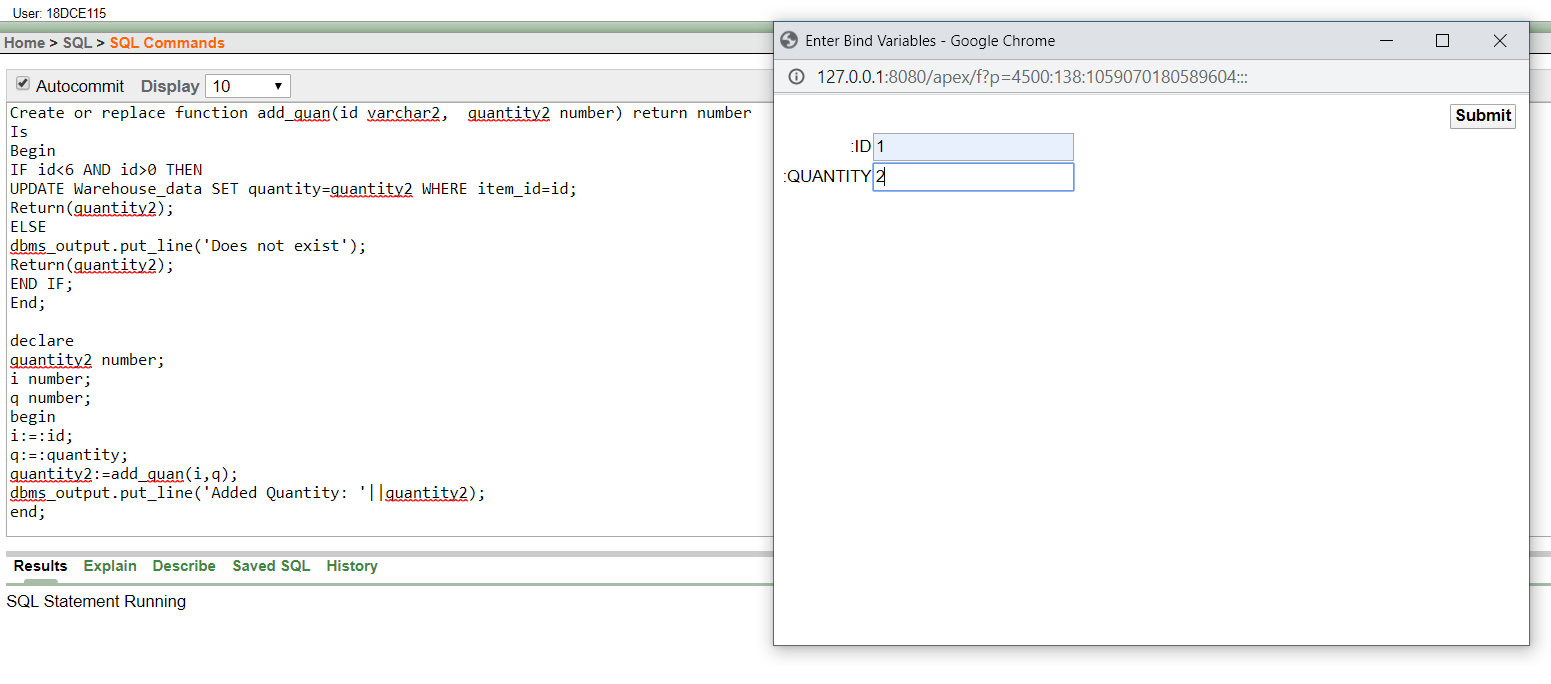
i:=:id;

q:=:quantity;

quantity2:=add\_quan(i,q);

dbms\_output.put\_line('Added Quantity: '||quantity2);

end;



CREATE OR REPLACE PROCEDURE update\_price(id IN varchar2, price IN number,new\_price OUT number)

Is

Begin

UPDATE Warehouse\_data SET price=price\*quantity where item\_id=id;

select price into new\_price from Warehouse\_data where item\_id=id;

End;

declare

new\_price number;

i number;

p number;

begin

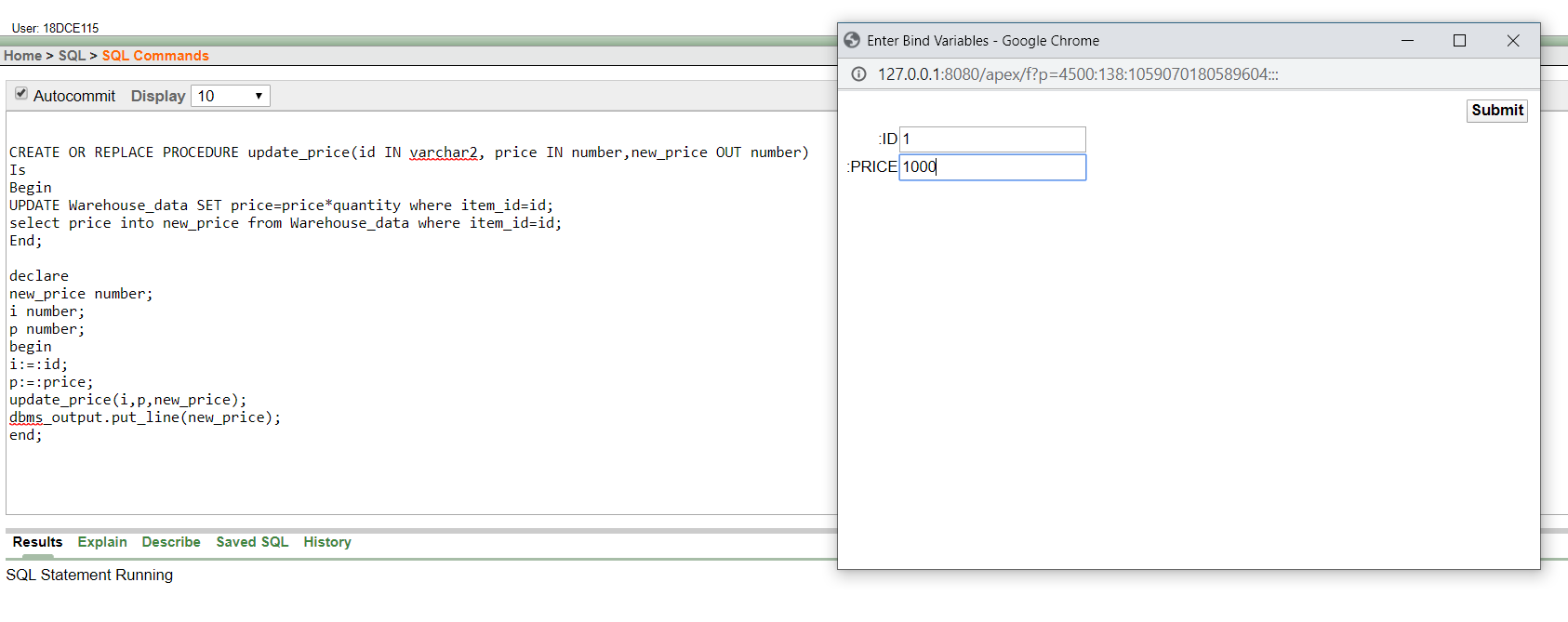
i:=:id;

p:=:price;

update\_price(i,p,new\_price);

dbms\_output.put\_line(new\_price);

end;



Create or replace function total(p number) return number

Is

a Warehouse\_data.price%type;

Begin

SELECT sum(price) INTO a FROM Warehouse\_data;

Return(a);

End;

DECLARE

cost Warehouse\_data.price%type;

BEGIN

cost:=total(1);

DBMS\_OUTPUT.PUT\_LINE('total cost: '||cost);

END;

