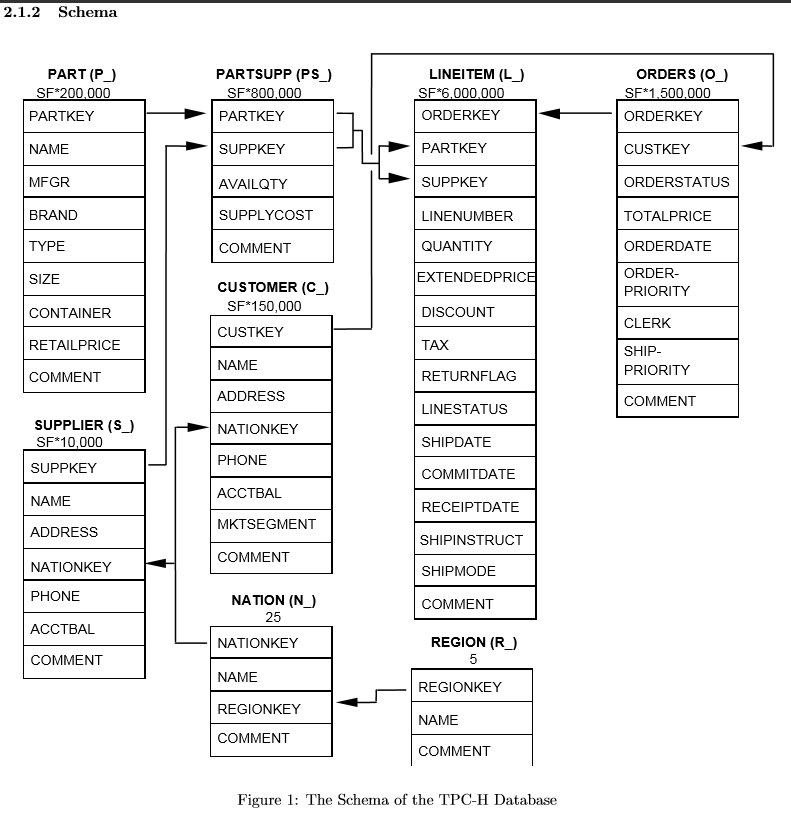
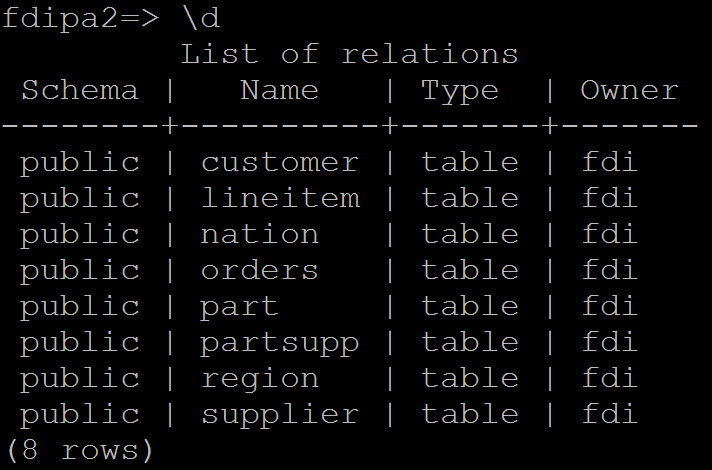
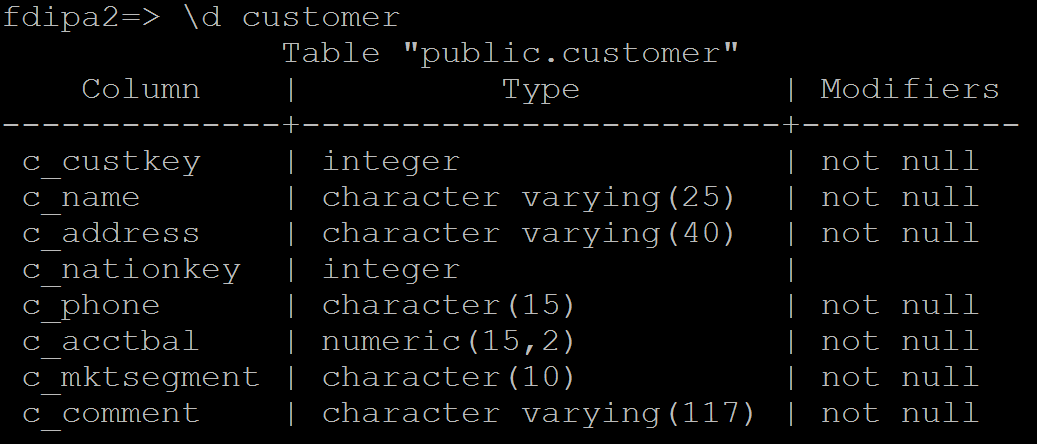
# List of Relations





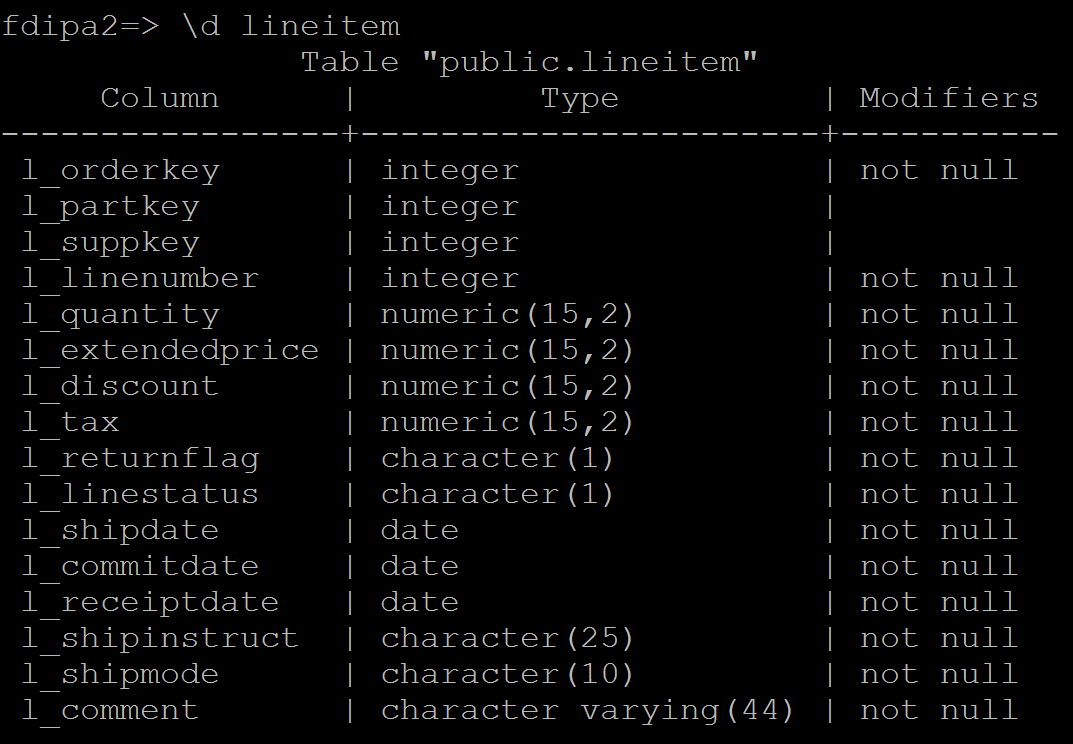
## Customer C\_

This table contains information regarding the customers of the manufacturing company. Information stored for each customer includes his/her name (name), address (address) and nation he/she resides in (nationkey), as well as a contact phone number (phone), and the balance of the customers account with the company (acctbal).



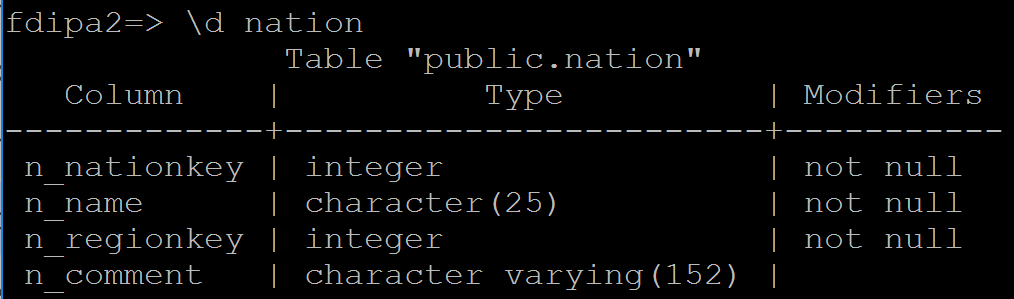
## Lineitem L\_

This table maintains each individual part order contained in a purchase order. Every row represents some ordered part (identiﬁed by the part (partkey) and the supplier it was ordered from (suppkey)) the quantity ordered (quantity), the total price paid for the parts (extendedprice) before tax and a discount, the date when the parts were shipped (shipdate), the date when the parts were promised by the supplier (commitdate), the date when the parts were received (receiptdate), the discount received (discount) and so on. As well, each lineitem contains an attribute (linestatus) that has a value of CO if the part order is still open and CF if the part order has been fulﬁlled.



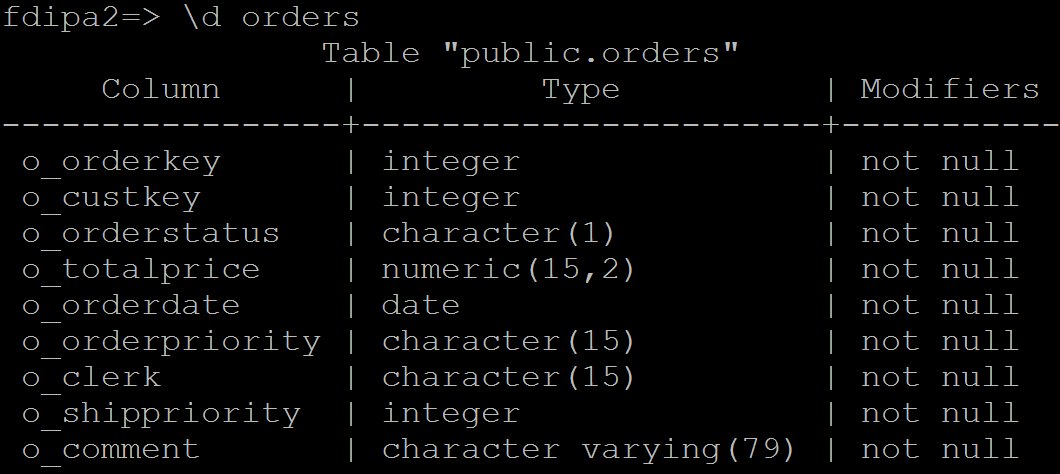
## Nation N\_

This table lists all nations the manufacturing company deals with. Every nation has a name (name), a region in which it is contained (regionkey) and a comment (comment).



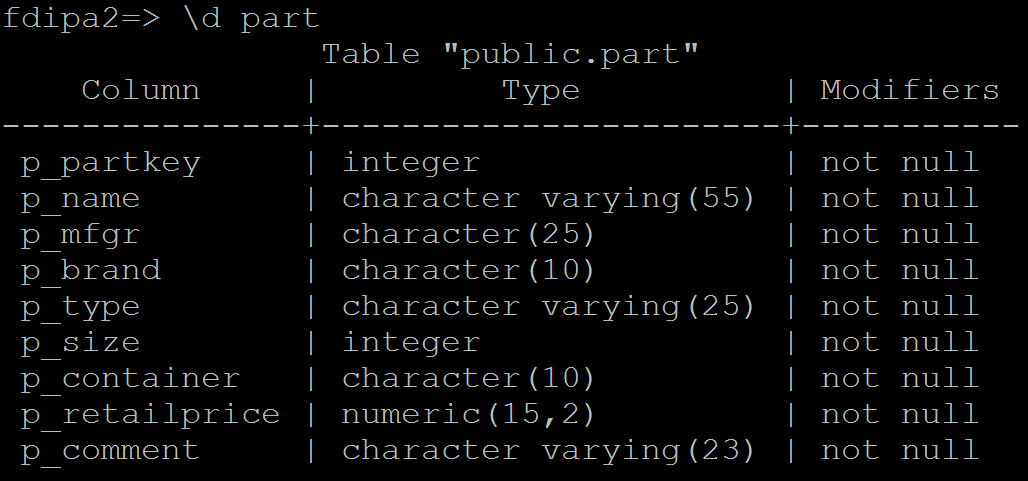
## Orders O\_

This table contains details about every purchase order (or just, order). An order has a single customer (though a customer can maintain multiple orders) and is made up of one or more lineitems. An order status (orderstatus) is set to CO if it is open (i.e., if all lineitems it contains have an open status), CF if the order is fulﬁlled (i.e., every lineitem it contains is fulﬁlled), and CP if it is partially fulﬁlled (i.e., if some but not all lineitems in the order have been fulﬁlled). Order records also include the total price of the order (totalprice) and the date the order was placed (orderdate).



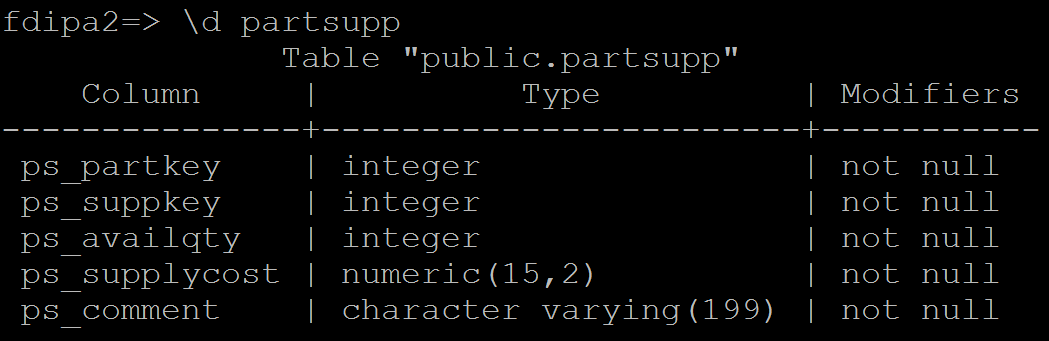
## Part P\_

This table holds information about all parts needed by the manufacturing company. Relevant data for a part includes its name (name), the parts manufacturer (mfgr), and its retail price (retailprice).



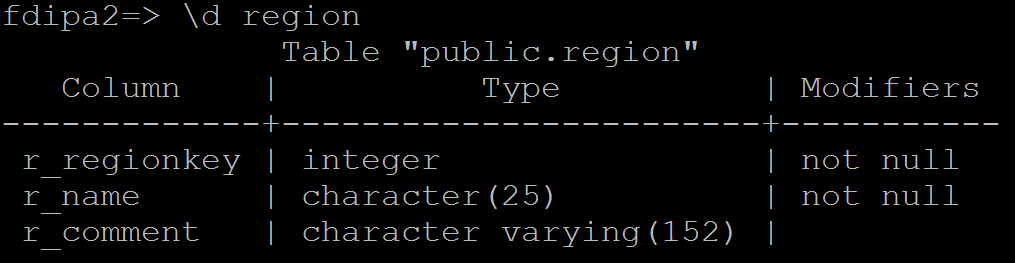
## Partsupp PS\_

A row in this table represents a given part (partkey) that is supplied to the manufacturer by a given supplier (suppkey). For each part and supplier, additional data includes the number of parts available from the supplier (availqty) and the suppliers cost for the part (supplycost).



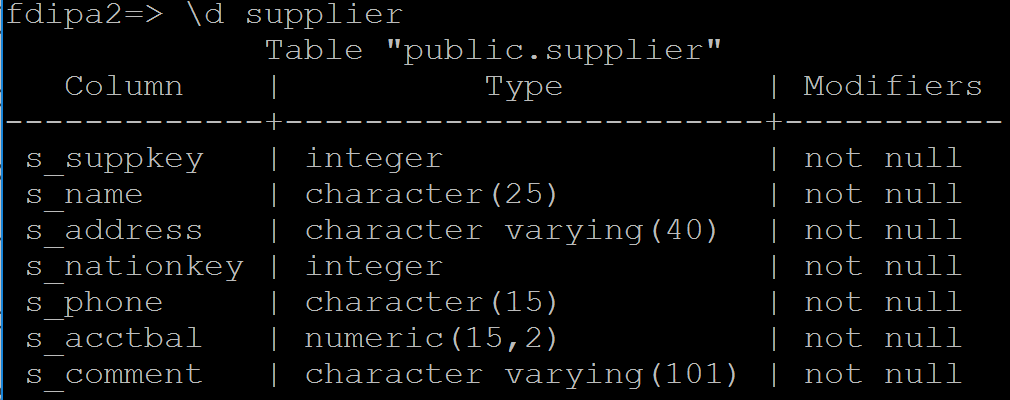
## Region R\_

This table lists the 5 regions (i.e., continents) in the world the data in this database pertains to. Each region has a name (name) and a comment permitting annotations for the region (comment).



## Supplier S\_

This table lists all suppliers of interest for the manufacturing company. Data for a supplier includes its name (name), address (address), the nation where it is located (nationkey), phone (phone), account balance (acctbal) and comments about the supplier (comment).



# Trigger 1

ALTER TABLE Customer DROP CONSTRAINT fk1;

ALTER TABLE Customer ADD CONSTRAINT fk1 FOREIGN KEY (c\_nationkey) REFERENCES Nation (n\_nationkey) ON UPDATE CASCADE;

ALTER TABLE Supplier DROP CONSTRAINT fk1;

ALTER TABLE Supplier ADD CONSTRAINT fk1 FOREIGN KEY (s\_nationkey) REFERENCES Nation (n\_nationkey) ON UPDATE CASCADE;

# Trigger 2

CREATE FUNCTION function2() RETURNS TRIGGER AS $function2$

BEGIN

UPDATE partsupp

SET ps\_supplycost = ps\_supplycost + NEW.p\_retailprice - OLD.p\_retailprice

WHERE ps\_partkey = NEW.p\_partkey;

RETURN NEW;

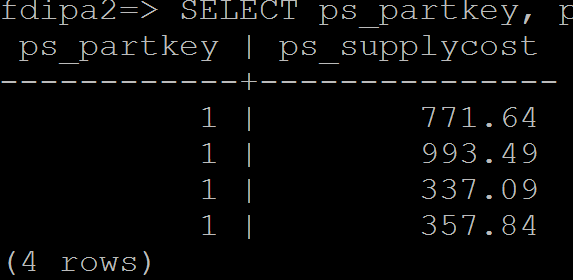
END;

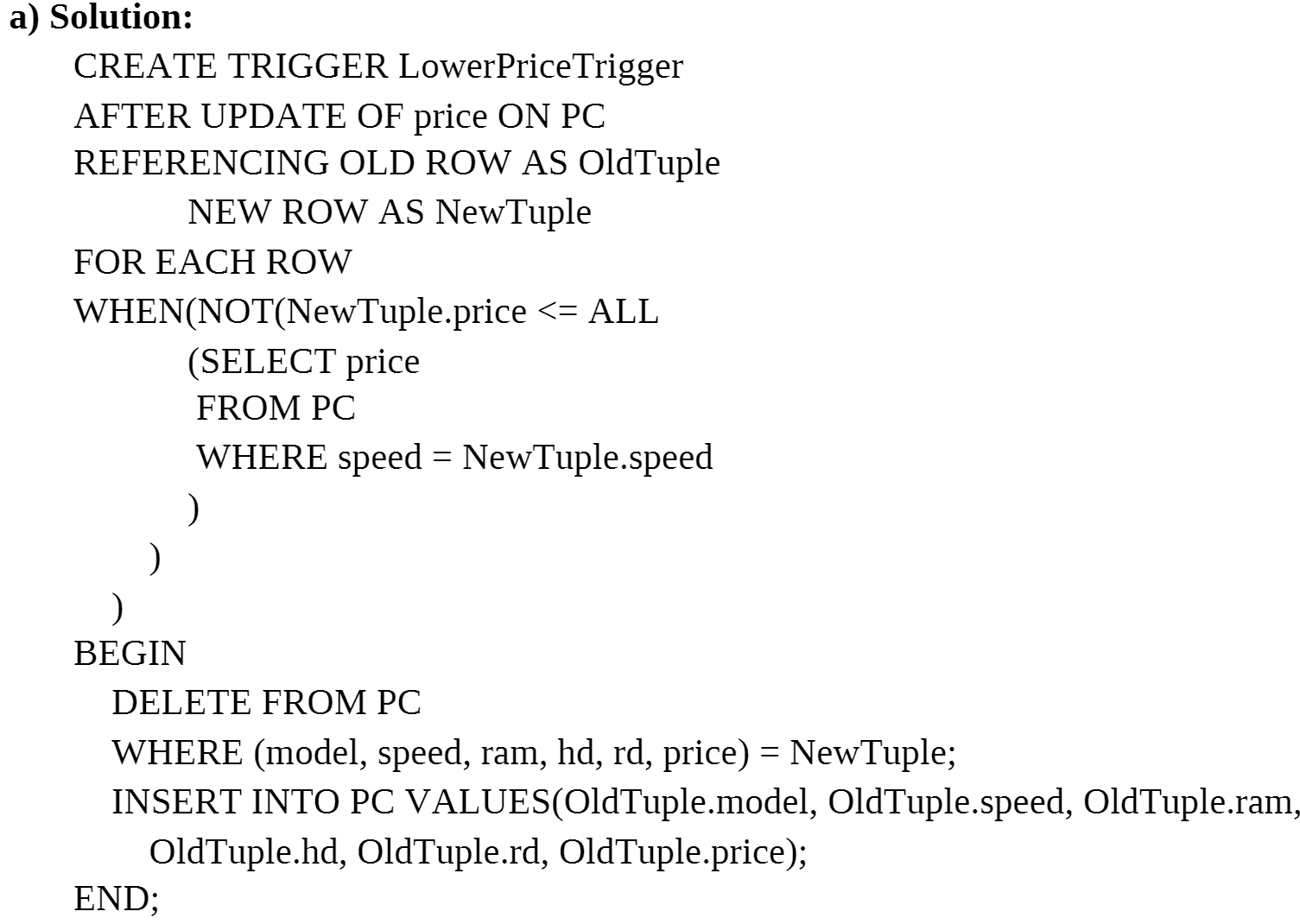
$function2$ LANGUAGE plpgsql;

CREATE TRIGGER trigger2

BEFORE UPDATE ON part

FOR EACH ROW EXECUTE PROCEDURE function2();





# Trigger 3

# Trigger 6

1. If Delete a region
2. Delete all nations in the region
3. Delete suppliers located in any of the nations
4. Delete partsupp record associated with suppliers
5. Do not delete
6. Customers set to Null
7. Lineitem set to Null

# Trigger 7