Database Triggers

Team-16

Ishan Mistry - 18BIT033

Alister Rodrigues - 18BIT041

Khush Joshi - 18BIT056

An analogy awaits

Ever watched a line of dominoes fall when you knock the first one down? This is basically a trigger.

Your touch of the first tile triggered the chain reaction that knocked down the rest. That’s what database triggers do!!!

A single event on a database element, like a table, can cause a chain reaction of many other events in the database.

Definition:

A trigger, in database terms, is a set of instructions that are 'fired' by some specific event, normally a command issued through the database's Data Manipulation Language (DML)

Syntax:

```
create trigger [trigger_name]

[before | after]

{insert | update | delete}

on [table_name]

[for each row]

[trigger_body]
```

Why use triggers when we have constraints?

INTERESTING....

Constraints and DML triggers both have certain types of things they do well. Constraints are great at maintaining database integrity.

Triggers are great for checking past vs. current values and making decisions based on that data.

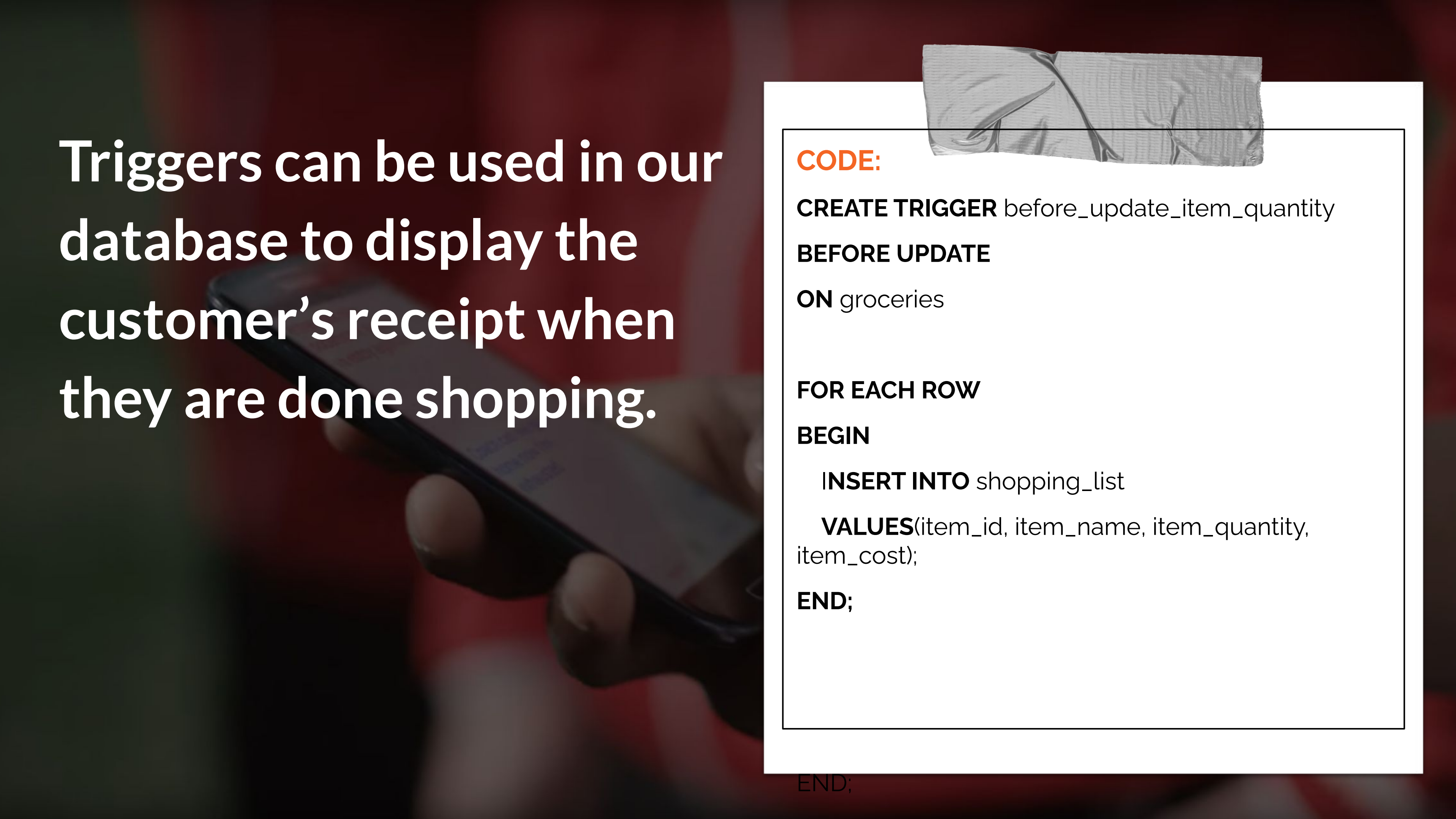
It is more advantageous to use a constraint than a trigger (if you can use a constraint in your situation). You can always write a trigger to do the same work that a type of constraint can do, but it typically doesn't make sense to do so..

CODE:

```
CREATE TRIGGER after_update_weights
AFTER UPDATE
ON Band_Tag
FOR EACH ROW
BEGIN
  IF OLD.Vessel_Weight<>new.Vessel_Weight  THEN
      SET new.Vessel_Weight = OLD.Vessel_weight -
             new.Vessel_weight
  END IF
END;
```

A trigger that can be used to update the empty vessel weight with the differnce to get the quantity of product filled in

END;



Triggers can be used in our database to display the customer’s receipt when they are done shopping.

CODE:

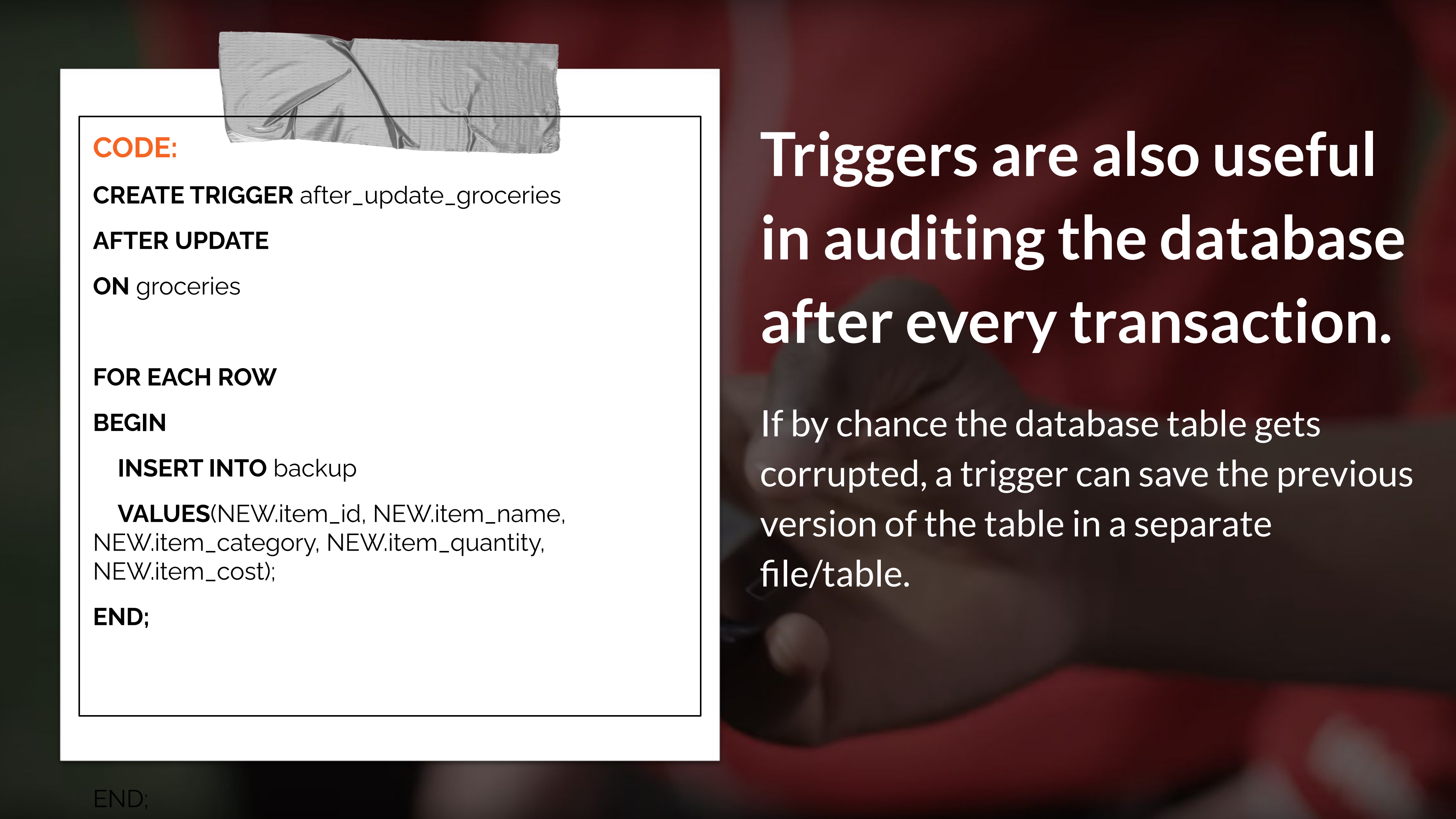
```
CREATE TRIGGER before_update_item_quantity
BEFORE UPDATE
ON groceries


FOR EACH ROW
BEGIN
    INSERT INTO shopping_list
    VALUES(item_id, item_name, item_quantity,
item_cost);
END;
```

END;

Triggers are also useful in auditing the database after every transaction.

If by chance the database table gets corrupted, a trigger can save the previous version of the table in a separate file/table.

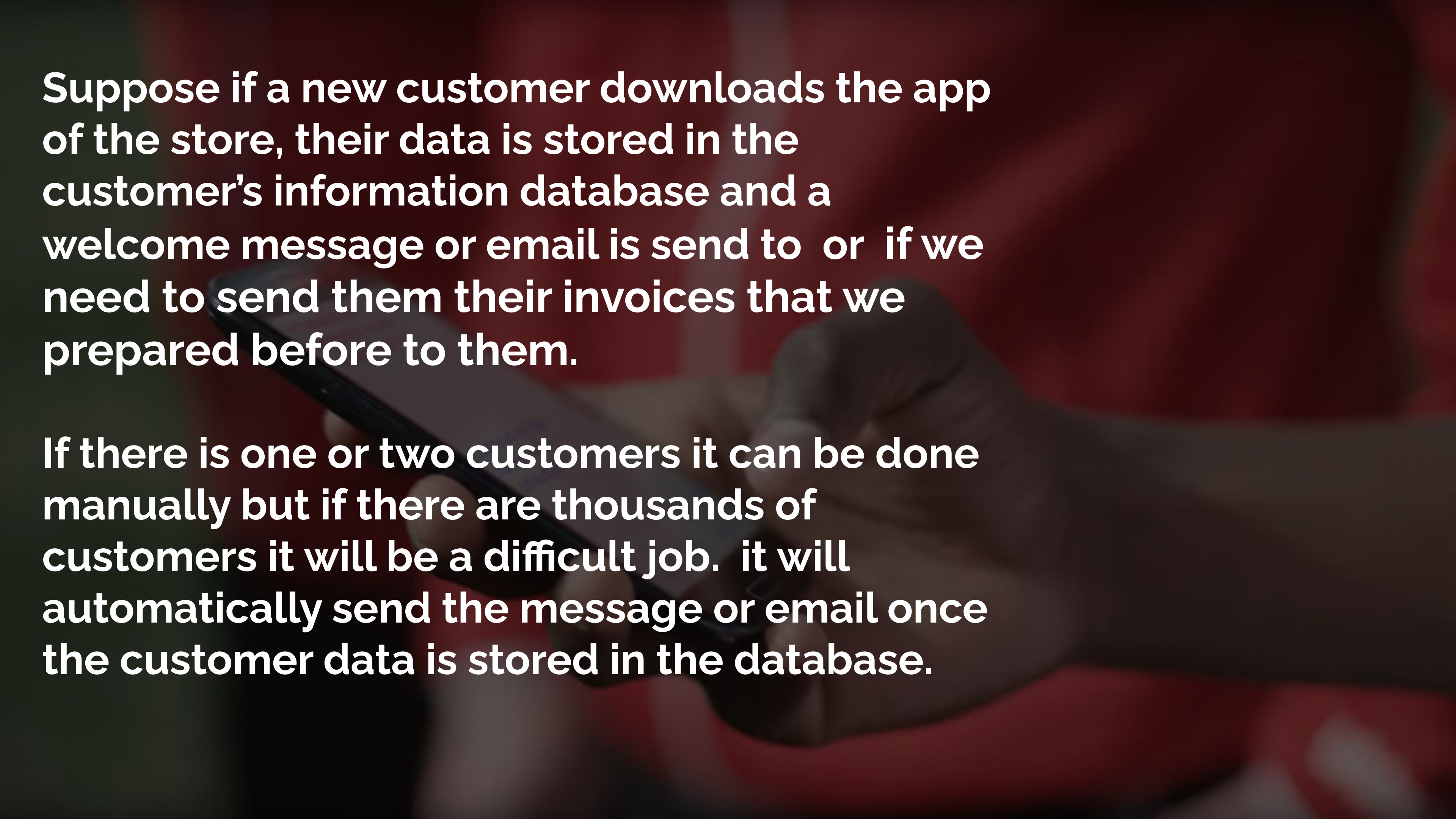


CODE:

```
CREATE TRIGGER after_update_groceries
AFTER UPDATE
ON groceries

FOR EACH ROW
BEGIN
   INSERT INTO backup
   VALUES(NEW.item_id, NEW.item_name,
NEW.item_category, NEW.item_quantity,
NEW.item_cost);
END;
```

END;



Suppose if a new customer downloads the app of the store, their data is stored in the customer's information database and a welcome message or email is send to or if we need to send them their invoices that we prepared before to them.

If there is one or two customers it can be done manually but if there are thousands of customers it will be a difficult job. it will automatically send the message or email once the customer data is stored in the database.

Advantages from a Business P.O.V

Faster application development.

As triggers are stored in the database there's no individual codes

Easier maintenance.

If a business policy changes, you need to change only the trigger program instead of each application program.

TRIGGERS

Global enforcement of business rules.

Define a trigger once and then reuse it for any application that uses the database.

Improve performance in client/server environment.

All rules run in the server before the result returns.

Advantages of Triggers

- Forcing security approvals on the tables.
- It checks the integrity of the data.
- It helps in counteracting the invalid exchanges.
- It handles error.
- They are useful in inspecting the data.

Disadvantages of Triggers

- Triggers can only provide extended validation, i.e. not all kind of validations, for simple validations we cannot use NOT_NULL, UNIQUE, CHECK and Foreign_Key constraints in a given trigger.
- Triggers may increase the overhead of the database.
- Triggers can be difficult to troubleshoot because they execute automatically in the database this may not be visible to client applications.