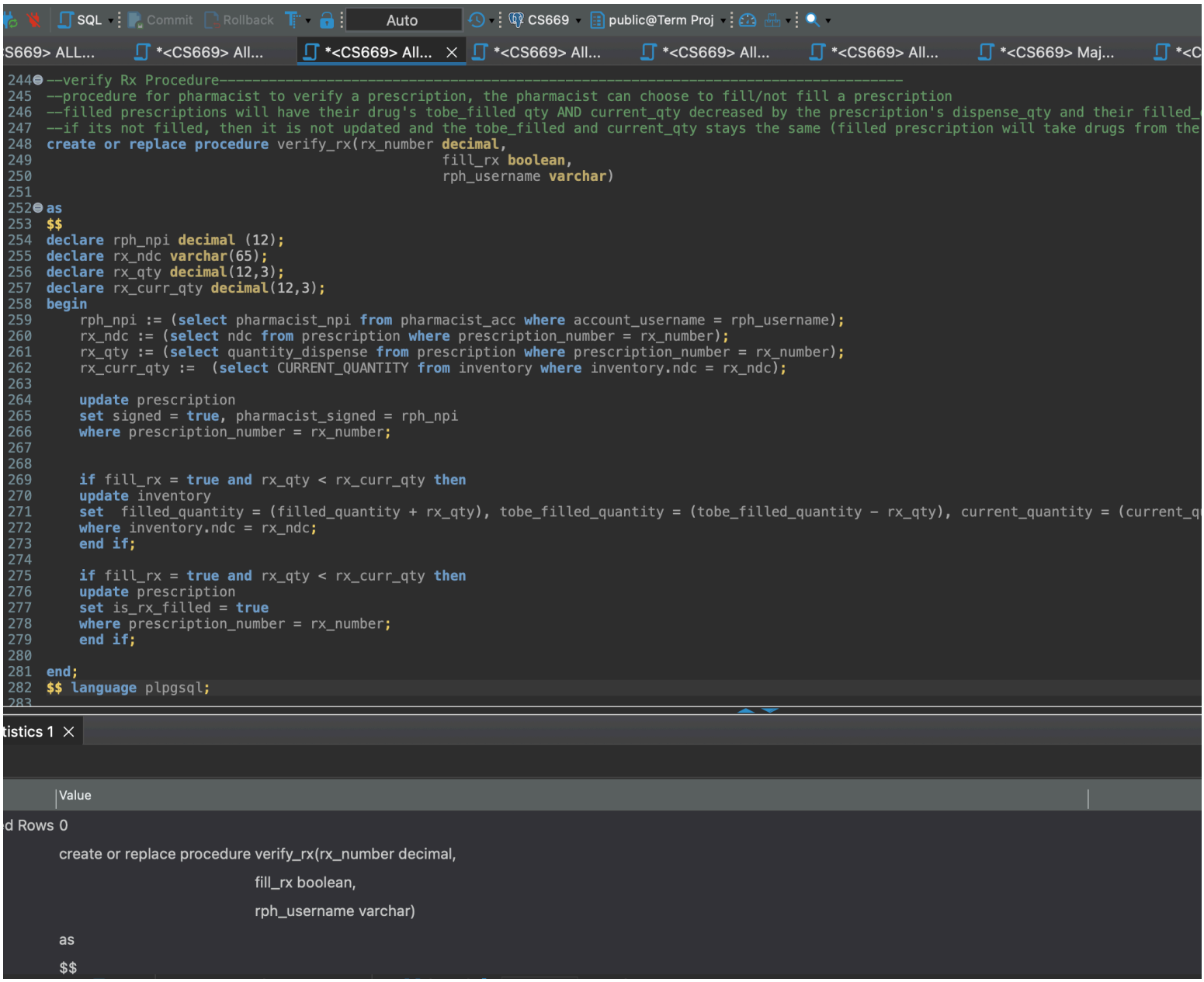
Verify Prescription stored procedure



```
--verify Rx Procedure----------------------------------------------------------------------------
--procedure for pharmacist to verify a prescription, the pharmacist can choose to fill/not fill a prescription
--filled prescriptions will have their drug's tobe_filled qty AND current_qty decreased by the prescription's dispense_qty and their filled_
--if its not filled, then it is not updated and the tobe_filled and current_qty stays the same (filled prescription will take drugs from the
create or replace procedure verify_rx(rx_number decimal,
                                      fill_rx boolean,
                                      rph_username varchar)

as
$$
declare rph_npi decimal (12);
declare rx_ndc varchar(65);
declare rx_qty decimal(12,3);
declare rx_curr_qty decimal(12,3);
begin
    rph_npi := (select pharmacist_npi from pharmacist_acc where account_username = rph_username);
    rx_ndc := (select ndc from prescription where prescription_number = rx_number);
    rx_qty := (select quantity_dispense from prescription where prescription_number = rx_number);
    rx_curr_qty :=  (select CURRENT_QUANTITY from inventory where inventory.ndc = rx_ndc);

    update prescription
    set signed = true, pharmacist_signed = rph_npi
    where prescription_number = rx_number;


    if fill_rx = true and rx_qty < rx_curr_qty then
    update inventory
    set  filled_quantity = (filled_quantity + rx_qty), tobe_filled_quantity = (tobe_filled_quantity - rx_qty), current_quantity = (current_q
    where inventory.ndc = rx_ndc;
    end if;

    if fill_rx = true and rx_qty < rx_curr_qty then
    update prescription
    set is_rx_filled = true
    where prescription_number = rx_number;
    end if;

end;
$$ language plpgsql;
```

```
Value
d Rows 0
    create or replace procedure verify_rx(rx_number decimal,
                                          fill_rx boolean,
                                          rph_username varchar)
    as
    $$
```

This stored procedure will indicate which prescription has been looked at by a pharmacist. I've declared a few variables to shorten the query. The store procedure will take the prescription number, a filled_rx boolean, and the pharmacist's username. Given the pharmacist's username, a query is used to obtain the pharmacist's npi from the pharmacist_acc table and assigns it as the rph_npi variable. The rx_ndc variable returns the ndc from the prescription table given the prescription number argument. The rx_qty variable returns the quantity_dispense from the prescription table given the prescription number argument. The

rx_curr_qty returns the currenty_quantity from the inventory table given the ndc from the earlier variable rx_ndc.

The stored procedure updates the prescription table where only the prescription_number matches the rx_number argument. The 'signed' and 'pharmacist_signed' attributes are update to 'true' and rph_npi variable. This just means that the prescription is valid and the pharmacist's signature (his/her npi) is attached to the prescription number.

The stored procedure has an option to fill the prescription or not despite having a pharmacist validating the prescription. Sometimes a pharmacist may refrain from filling the prescription due to inventory constrains. If the boolean 'fill_rx' is false, then nothing happens the procedure ends after the prescription update. But if the 'fill_rx' is true, two conditions will happen, either the inventory have enough to fill the prescription quantity or it does not. Only when the inventory have enough quantity will additional functions results.

When the 'fill_rx' is true and the inventory carries enough quantity, then the 'filled_quantity' in the inventory table will update to the sum of 'filled_quantity' (if any) and the rx_qty, the 'tobe_filled_quantity' will be less by rx_qty amount (since a fill_rx is no longer a tobe_fill), and the current_quantity is less by rx_qty as well (since a filled prescription takes away the available quantity). The 'Is_Rx_filled' in the prescription table will also be updated to 'True' for the corresponding rx_number argument.

```
--verify prescription-----------------------------------------------------------------------------------------------------------------------------------------
--argument: rx_number, fill_rx?, rph_username
--when verified -> rph control to be filled or not -> if filled -> change tobe_filled_qty to filled_qty
start transaction;
do
$$
begin
    call verify_rx(1, true, 'Mary_Lowe90');

    call verify_rx(2, true,'John8844rph');

    call verify_rx(3, true, 'Mary_Lowe90');

    call verify_rx(4, true, 'Mary_Lowe90');

    call verify_rx(5, true, 'Mary_Lowe90');

    call verify_rx(6, true,'John8844rph');

    call verify_rx(7, true,'John8844rph');

    call verify_rx(8, true,'John8844rph');

    call verify_rx(9, true,'John8844rph');

    call verify_rx(10, true,'John8844rph');

    call verify_rx(11, true,'John8844rph');

    call verify_rx(12, true,'John8844rph');

end$$;
commit transaction;
```

Statistics 2 ×

Value

ws 0

```
start transaction;
do
$$
begin
    call verify_rx(1, true, 'Mary_Lowe90');
    call verify_rx(2, true,'John8844rph');
    call verify_rx(3, true, 'Mary_Lowe90');
    call verify_rx(4, true, 'Mary_Lowe90');
    call verify_rx(5, true, 'Mary_Lowe90');
    call verify_rx(6, true,'John8844rph');
    call verify_rx(7, true,'John8844rph');
    call verify_rx(8, true,'John8844rph');
    call verify_rx(9, true,'John8844rph');
    call verify_rx(10, true,'John8844rph');
```