Assignment 4 (Chapter 4, Chapter5)

4.13 Consider a view v whose definition references only relation r.

(1) If a user is granted select authorization on v, does that user need to have select authorization on r as well? Why or why not?

**Answer: No, the user does not need to have select authorization on r.**

**(Chapter 4, page 4.54), "Granting a privilege on a view does not imply granting any privileges on the underlying relations."**

(2) If a user is granted update authorization on v, does that user need to have update authorization on r as well? Why or why not?

**Answer: Yes, the user does need to have update authorization on r.**

**If v is updatable, an UPDATE operation on v translates to an UPDATE on r. To update a view, the user must have the appropriate privileges for the view and the underlying tables.**

5.6 Consider the bank database of Figure 5.21. Let us define a view branch\_cust as follows:

**create view** branch\_cust as (

**selec**t branch\_name, customer\_name

**from** depositor, account

**where** depositor.account\_number = account.account\_number);

Suppose that the view is materialized (so you can regard branch\_cust as a base table); that is, the view is computed and stored. Write triggers to maintain the view, that is, to keep it up-to-date on insertions to depositor or account. It is not necessary to handle deletions or updates. Note that, for simplicity, we have not required the elimination of duplicates.

branch (branch\_name, branch\_city, assets)

customer (customer\_name, customer\_street, customer\_city)

loan (loan\_number, branch\_name, amount)

borrower (customer\_name, loan\_number)

account (account\_number, branch\_name, balance )

depositor (customer\_name, account\_number)

Figure 5.21 Banking database for Exercise 5.6.

**Answer: Trigger on depositor:**

**CREATE TRIGGER maintain\_branch\_cust\_after\_insert\_depositor**

**AFTER INSERT ON depositor**

**FOR EACH ROW**

**BEGIN**

**INSERT INTO branch\_cust (branch\_name, customer\_name)**

**SELECT account.branch\_name, NEW.customer\_name**

**FROM account**

**WHERE account.account\_number = NEW.account\_number;**

**END;**

**Trigger on account:  
No trigger is needed for insertions into account. When a new account is inserted, no matching depositor tuples exist yet, so the join condition depositor.account\_number = account.account\_number produces no results for the new account\_number.**