

Relational Database Operations in SQL - Part III - Database Views

Hui Chen ^a

^aCUNY Brooklyn College, Brooklyn, NY, USA

May 3, 2022

Outline

- 1 Recap and Project
 - Project
 - Recap: SQL and Relational Algebra
- 2 Database Views
 - Virtual Views
 - Materialized Views
- 3 Summary
- 4 Assignment

Outline

- 1 Recap and Project
 - Project
 - Recap: SQL and Relational Algebra
- 2 Database Views
 - Virtual Views
 - Materialized Views
- 3 Summary
- 4 Assignment

Reminder: Project Meeting

Before final project demo, each group should schedule a meeting with me in this or the next week – more scheduling details will be on Blackboard.

Agenda and Objectives

- ▶ Discuss group and individual progress
- ▶ Identify gaps and improvements
- ▶ Prepare for the final and a successful project demo and presentation
- ▶ Any issues you may have regarding the class

Selected Topics in SQL

Discussed

- ▶ Ordering the Output
- ▶ Eliminating Duplicates
- ▶ Aggregate Processing
- ▶ Grouping
- ▶ Subquery

Now discuss

- ▶ Database Views

and do some exercises in class, and continue on,

- ▶ Procedural SQL

Outline

- 1 Recap and Project
 - Project
 - Recap: SQL and Relational Algebra
- 2 Database Views
 - Virtual Views
 - Materialized Views
- 3 Summary
- 4 Assignment

Database Views

Database views are relations that are defined by a query over other relations.

- ▶ **Virtual Views.** Not stored in the database, but can be queried as if they existed.
- ▶ **Materialized Views.** Constructed periodically from the database and stored there.

Why views?

Creating Virtual Views

To create a virtual view, use

```
CREATE VIEW <view-name> AS <view-definition>;
```

where the view definition is a SQL query

Creating Virtual Views: Example

To create a virtual view, use

```
CREATE VIEW StudentLists AS  
(  
  SELECT name, phone, address, cidnum  
  FROM Students AS s INNER JOIN Enrollment AS e  
  WHERE s.name = e.sname AND s.phone = e.sphone  
);
```

Using Virtual Views: Example

```
SELECT *  
FROM StudentLists  
WHERE cidnum='1111';
```

Creating Virtual Views: Renaming Attributes

Sometimes, we wish to give a view's attributes names of our own choosing, rather than use the names that come out of the query defining the view.

```
CREATE VIEW <view-name(list-of-attributes)> AS <view-definition>
```

Creating Virtual Views with Renamed Attributes: Example

To create a virtual view, use

```
CREATE VIEW StudentLists(stu_name, stu_phone, stu_addr)
(  
  SELECT name, phone, address, cidnum  
  FROM Students AS s INNER JOIN Enrollment AS e  
  WHERE s.name = e.sname AND s.phone = e.sphone  
);
```

Modifying Views

- ▶ Drop views?
- ▶ Insert, delete, update on views?

Removing Views

To remove a view, use

```
DROP VIEW <view-name>;
```

Removing Views: Example

To remove a view, use

```
DROP VIEW StudentLists;
```

Updating Views

Some views are updatable – roughly, the views that are defined by selecting (but not `SELECT DISTINCT`) some attributes from one relation `R` (which may itself be an updatable view), more specifically,

- ▶ The `WHERE` clause must not involve `R` in a subquery.
- ▶ The `FROM` clause can only consist of one occurrence of `R` and no other relation.
- ▶ The list in the `SELECT` clause must include enough attributes that for every tuple inserted into the view, we can fill the other attributes out with `NULL` values or the proper default.

Materialized Views

Views can be materialized – to maintain its value at all times.

- ▶ Maintaining a materialized means we must recompute parts of the materialized view each time one of the underlying base tables changes – that is a computational cost.
- ▶ Why materialized views? – if used often, pre-computing the views can save query times
- ▶ Cost vs. benefits?

Creating Materialized Views

To create a materialized virtual view, use

```
CREATE MATERIALIZED VIEW <view-name> AS <view-definition>;
```

where the view definition is a SQL query

Outline

- 1 Recap and Project
 - Project
 - Recap: SQL and Relational Algebra
- 2 Database Views
 - Virtual Views
 - Materialized Views
- 3 Summary
- 4 Assignment

Questions and Summary

- ▶ Virtual Views
- ▶ Materialized Views

Outline

- 1 Recap and Project
 - Project
 - Recap: SQL and Relational Algebra
- 2 Database Views
 - Virtual Views
 - Materialized Views
- 3 Summary
- 4 **Assignment**

Assignment

Let's work on an exercise using paper and pen/pencil ...