CS122 Using Relational Databases and SQL

Views, Indexes, and Transactions

Chengyu Sun
California State University, Los Angeles

Views

create view view_name as *query*;

- A virtual table consists of the results of a query
- Example: create a view `members_salespeople_view (member_name, salesperson_name)`

About Views

- The data in a view is dynamically computed
  - Changes to *base tables* are automatically reflected in the view
- A view can be used as a table in SQL queries
- Views cannot be updated (except in some very rare cases)

Why Views

- Present the data in a different way
- Simplify SQL queries
- Security reasons
  - E.g. expose only part of the data to certain type of users

Indexes

- Make query execution more efficient

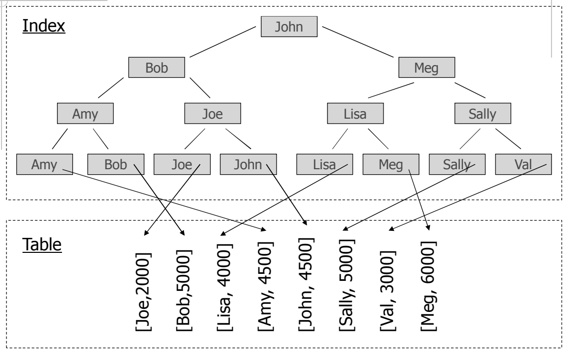
Query Example

select salary from employees where name = 'Sally';

employees

name	salary
Joe	2000
Bob	5000
Lisa	4000
Amy	4500
John	4500
Sally	5000
Val	3000
Meg	6000

Search with an Index



Create Index

```
create index index_name
        on table_name (col_name,…);
```

- Example: create an index on the `name` column of the `employees` table

```
create index emp_name_idx on employees (name);
```

The Need for Transaction

- Example: transfer $100 from account A to account B

accounts

account	balance
A	134.60
B	122.21
C	3300.00
D	256.79

SQL Statements Involved in A Transfer

```
-- Check whether account A has enough money
select balance from accounts where account = 'A';
-- Take $100 from account A
update account set balance = balance - 100
        where account = 'A';
-- Add $100 to account B
update account set balance = balance + 100
        where account = 'B';
```

Things Could Go Wrong

```
-- Check whether account A has enough money
select balance from accounts where account = 'A';
-- Take $100 from account A
update account set balance = balance - 100
        where account = 'A';
```

System Crash!

Transaction

- A group of statements that are treated as a whole, i.e. either all operations in the group are performed or none of them are – the Atomicity property.

Transaction Syntax in MySQL

```
begin;  -- start of a transaction

select balance from accounts where account = 'A';

update account set balance = balance - 100
        where account = 'A';

update account set balance = balance + 100
        where account = 'B';

commit; -- end of a transaction
```

(or rollback;)

ACID Properties of Database Transactions

- Atomic
- Consistent
- Isolated
- Durable