SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

SQL language

Course

Guillaume Claret

May 2021

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

B-trees

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

- used for indexes in tables
- balanced-trees structure
- large number of leafs (thousands, similar to a disk block size)
- search, insert, delete in logarithmic time
- insert and delete require balancing

Relational databases

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

- database engines (PostgreSQL, Oracle, SQLite, ...)
- query language (SQL)
- data as sets of tuples

Relational databases

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

- relational calculus (describe query)
- relational algebra (how to compute)

They are equivalent.

Relational algebra operators

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

- selection σ
- projection π
- join $\bowtie$
- set operators (union, intersection, difference)
- division A/B

Functional dependencies

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

- a set of columns can depend on another set
- super-key / key
- redundancies when not a super-key
- closure algorithm to find all functional dependencies

Normal forms

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

- 1NF: atomic columns
- 2NF: 1NF + all the key (on non-key attributes)
- 3NF: 2NF + only the key (on non-key attributes)
- BCNF: the only (non-trivial) functional dependencies are from a super-key
- algorithm to decompose to BCNF

Entity-relationships

- entity sets (elements)
- releationship sets (set entity tuples)
- attributes on entities or relationships
- primary key for the entities
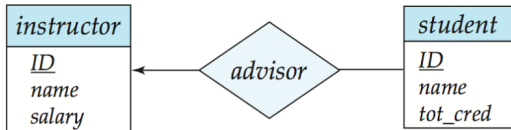- weak-entity sets

Entity-relationships

- diagrams
- one-to-one, one-to-many, many-to-many relationships
- total / partial relationships
- arity range on relationships
- method to convert to a relational schema:
    - one table per entities / relationships
    - elimination of redundancies

Others

- transactions (BEGIN, COMMIT, ROLLBACK)
- views:
  - similar to tables
  - new abstraction level
  - mostly read-only
- access-rights (users, roles, privileges)

SQL language

Presentation

- Structured Query Language
- to manipulate a database (especially run queries)
- supposed to be human readable
- highly-optimised implementations

Presentation

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

- several pronounciations
- several versions (SQL86, 92,... 2019)
- several implementations (Oracle, MySQL, Microsoft SQL, PostgreSQL, ...)

We focus on SQLite.

Parts of SQL

- Data Manipulation Language (DML):
  SELECT, INSERT, UPDATE, DELETE
- Data Definition Language (DDL):
  CREATE / ALTER / DROP TABLE
- Data Control Language (DCL):
  access-rights

Table schemas

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

Table schemas

```
Planes(
  ID: INT,
  Model: CHAR,
  Built_date: DATE,
  Number_of_seats: INT
)
```

- table name
- attribute names
- attribute types

Attribute type

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

- in SQLite: INTEGER, REAL, TEXT (Unicode), BLOB
- more precise: INT2, VARCHAR(100)
- advanced: JSON, polygons, ...
- NULL value: special value usable with any types

Constraints

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

- NOT NULL - Ensures that a column cannot have a NULL value
- UNIQUE - Ensures that all values in a column are different
- PRIMARY KEY - A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table
- FOREIGN KEY - Uniquely identifies a row/record in another table
- CHECK - Ensures that all values in a column satisfies a specific condition
- DEFAULT - Sets a default value for a column when no value is specified
- INDEX - Use to create and retrieve data from the database very quickly

Data Manipulation Language

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

- SELECT
- INSERT
- UPDATE
- DELETE

SELECT

To read data:

```
SELECT fields FROM tables WHERE conditions
```

Example:

```
SELECT email
FROM Users
WHERE firstname = 'Mike'
```

SELECT

More general form:

```
SELECT fields
FROM tables
WHERE conditions
GROUP BY fields
HAVING conditions
ORDER BY fields ASC / DESC
LIMIT / OFFSET n
```

SELECT

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

Can combine many SELECT with:

- INTERSECT, UNION, UNION ALL, EXCEPT operators
- SELECT on other SELECT
- IN and EXISTS operators

INSERT

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

To insert new data:

```
INSERT INTO table VALUES (v1, ..., vn)
INSERT INTO table SELECT ...
```

Example:

```
INSERT INTO Users
VALUES ('Mike', 'Bar', 'mike.bar@foo.com')
```

UPDATE

To change existing data:

```
UPDATE table
SET column_1 = value_1, ..., column_n = value_n
WHERE condition
```

Example:

```
UPDATE Users
SET email = 'mike.bar@foo.fr'
WHERE firstname = 'Mike' AND lastname = 'Bar'
```

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

DELETE

To remove data:

```
DELETE FROM table
WHERE condition
```

Example:

```
DELETE FROM Users
WHERE email = 'mike.bar@foo.fr'
```

Data Definition Language

- CREATE TABLE
- CREATE INDEX
- ALTER TABLE
- DROP TABLE

CREATE TABLE

To create a new table:

```
CREATE TABLE table (
  column_1 type_1 [local_constraint_1]
  ...
  column_n type_n [local_constraint_n]
  constraint_1
  ...
  constraint_m
)
```

CREATE INDEX

SQL language

Guillaume
Claret

Summary of
previous
classes

SQL language

Add an index to an existing table for optimization purposes.

ALTER TABLE

Add, remove or rename a column. May be slow.

DROP TABLE

Remove a table schema with all its content.