C++ MySQL Implementation

ECE141B Software Foundations II course project

References : Rick Gessner's slides and homework specs

Chi-Hsin Lo Feb 9, 2023

Project Overview - Weekly Progress
Week 1-3

Week 1
APPLICATION FRAMEWORK

1. Application commands

  a. Version

  b. Help

  c. Quit

2. Start on tableview

  a. This view will show table data results

Week 2
DATABASE COMMAND PROCESSING

DB COMMANDS

1. Create database

2. Use database

3. Drop database

4. Show databases

NEW COMPONENTS

1. Database

2. Storage

3. Index

WEEK 3
STORAGE AND ENTITIES

NEW COMMANDS.

1. Create table

2. Drop table

3. Describe table

4. Show tables

NEW COMPONENTS

1. Attributes and entities

Project Overview - Weekly Progress
Week 4-6

WEEK 4
SOL COMMAND PROCESSING

NEW COMMANDS

1. Insert into...

NEW COMPONENTS

1. SQL Statements

2. SQL Processor

WEEK 5
SELECTING RECORDS

NEW COMMANDS

1. SELECT * FROM

WEEK 6
UPDATE AND DELETE RECORDS

NEW COMMANDS

1. UPDATE «table» set fieldname="value" WHERE conditions...
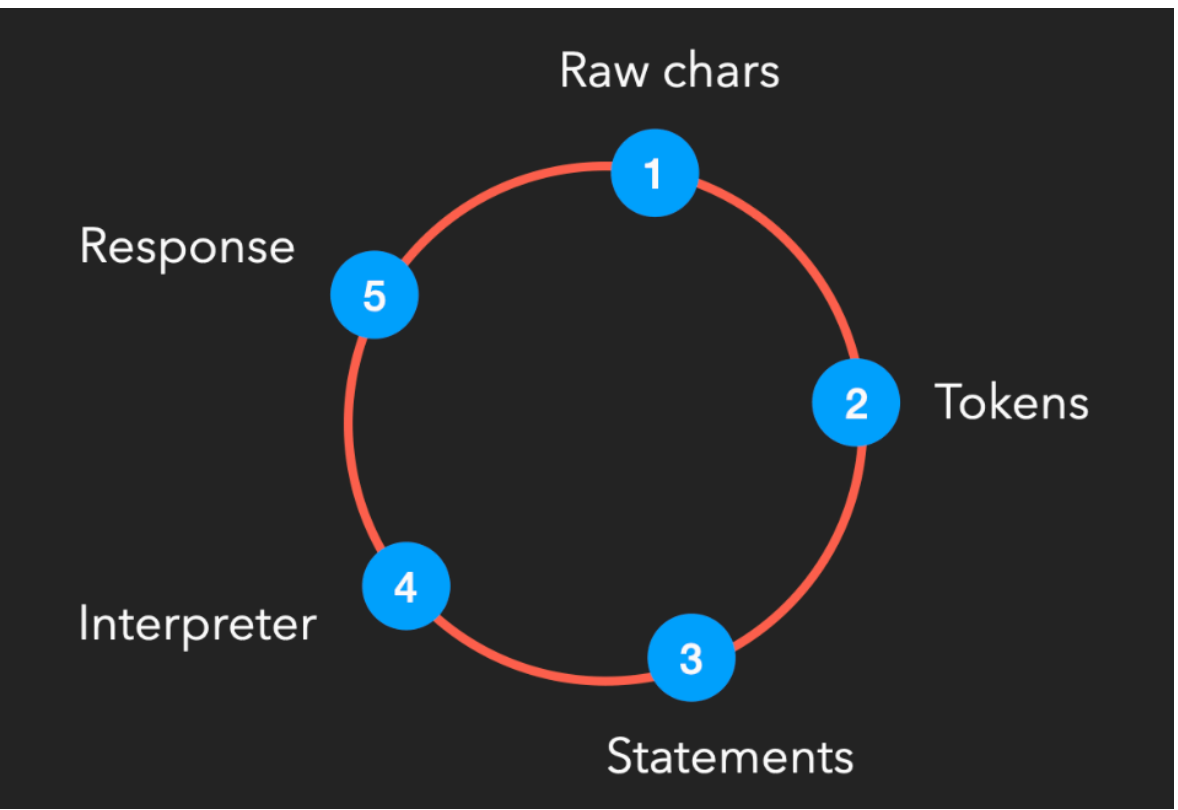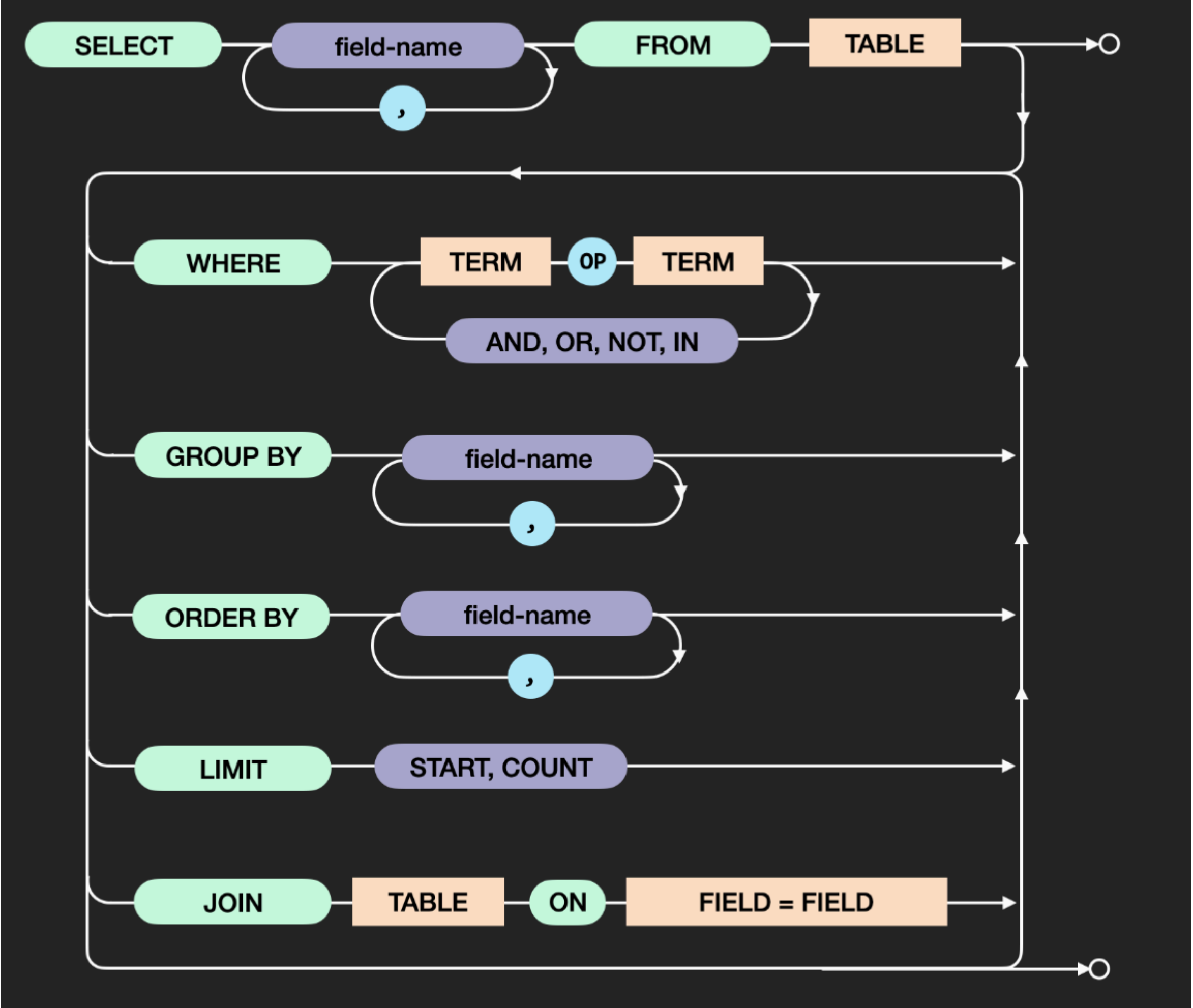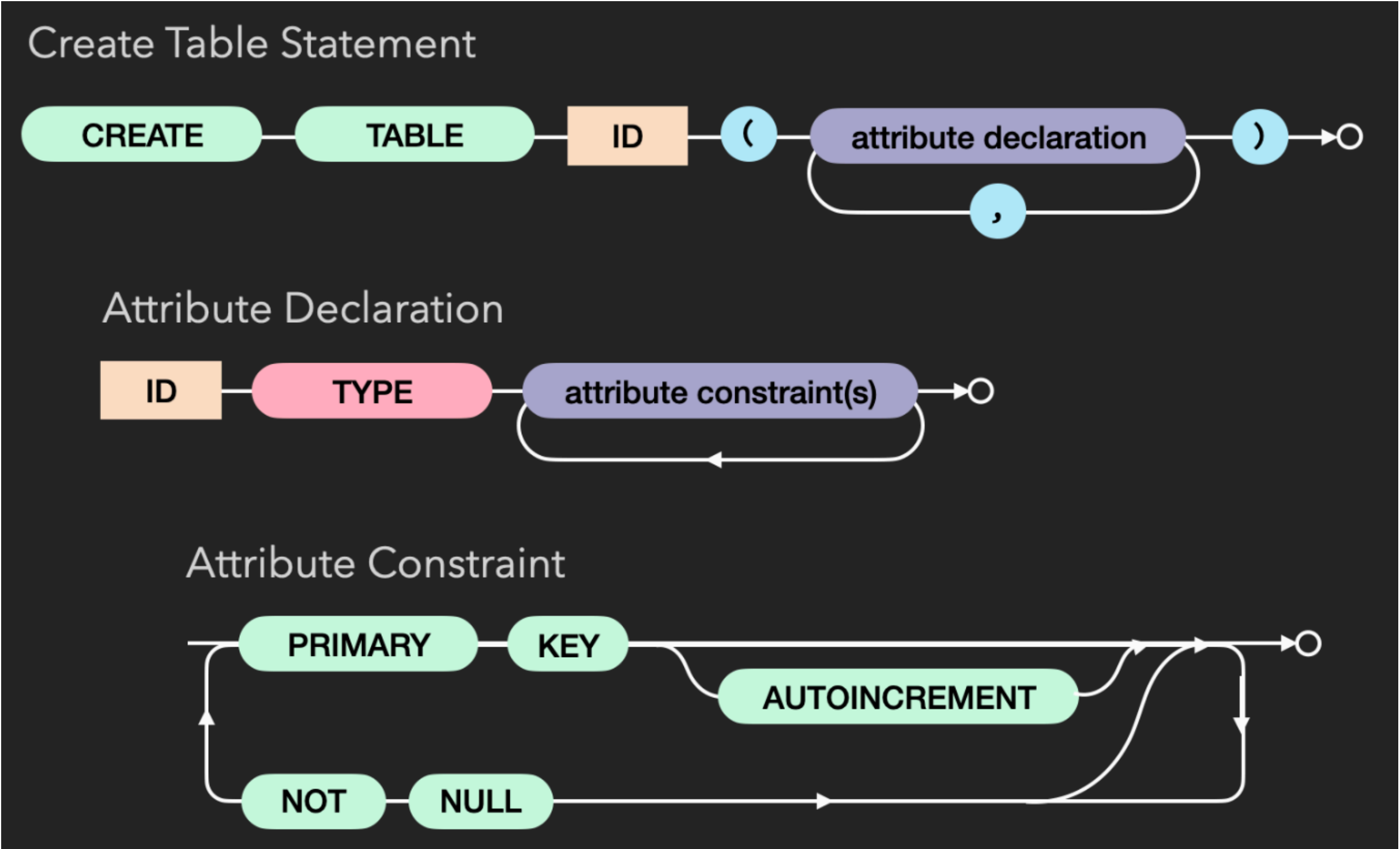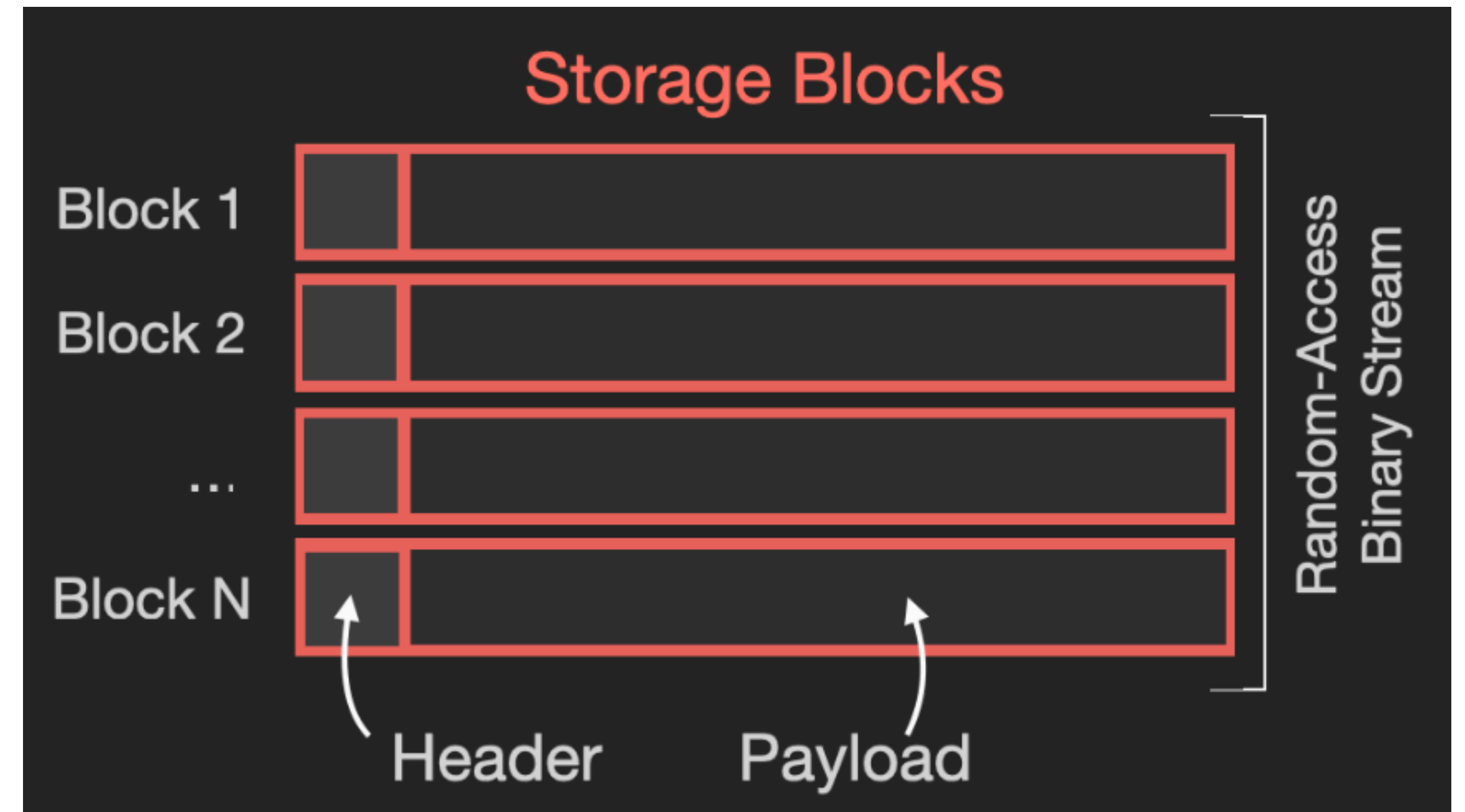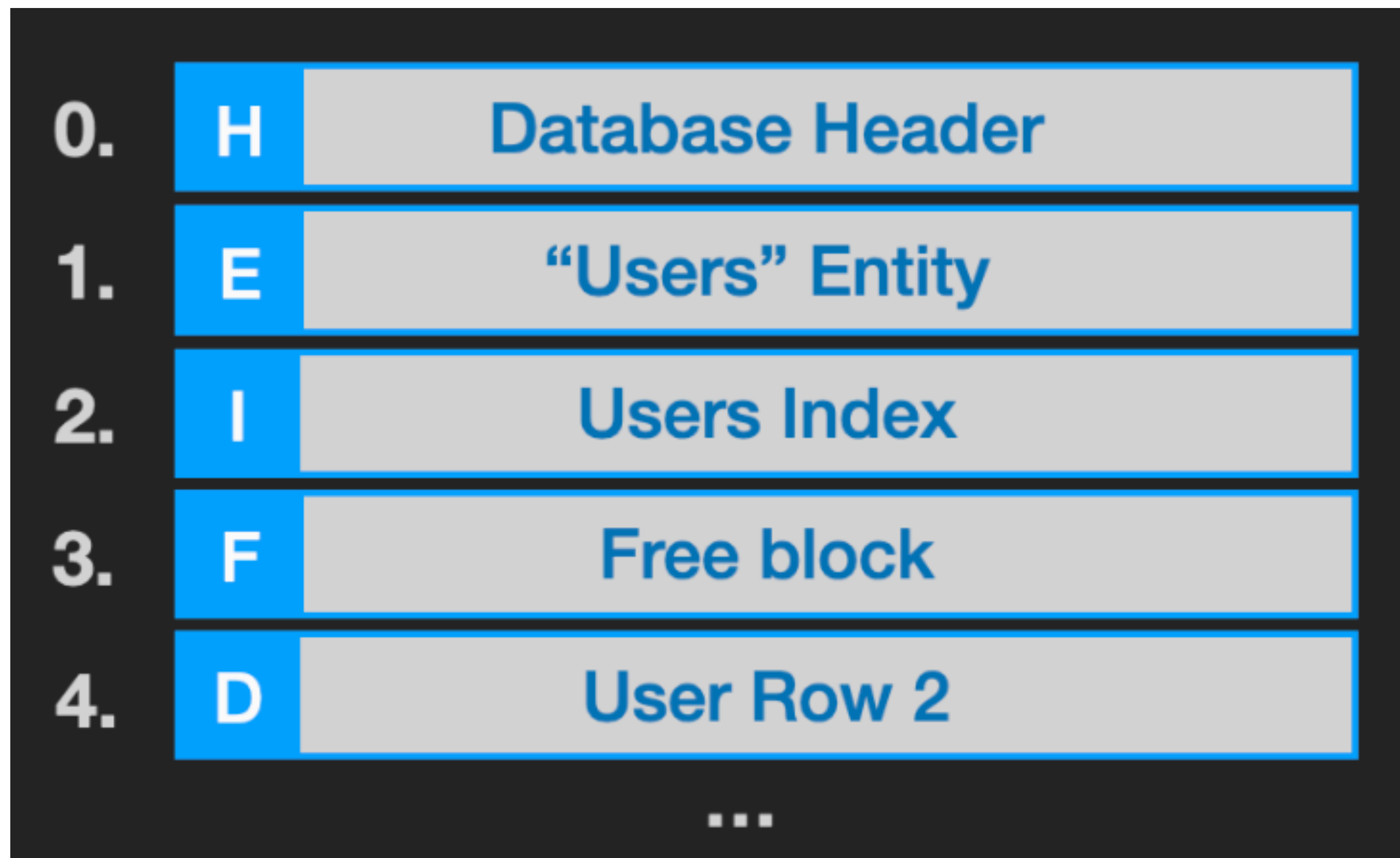
2. DELETE FROM <table> WHERE conditions...

CAPABILITIES

1. Marking storage rows as "free/unused'

Project Overview - Weekly Progress
Week 7-9

WEEK 7
INDEXING

NEW CAPABILITIES

1. creating an memory indexes

2. saving indexes to storage

3. loading indexes from storage (on demand)

WEEK 8
JOINING TABLES

NEW COMMANDS

1. SELECT * FROM <table1> LEFT JOIN «table2» ON ...

NEW CAPABILITIES
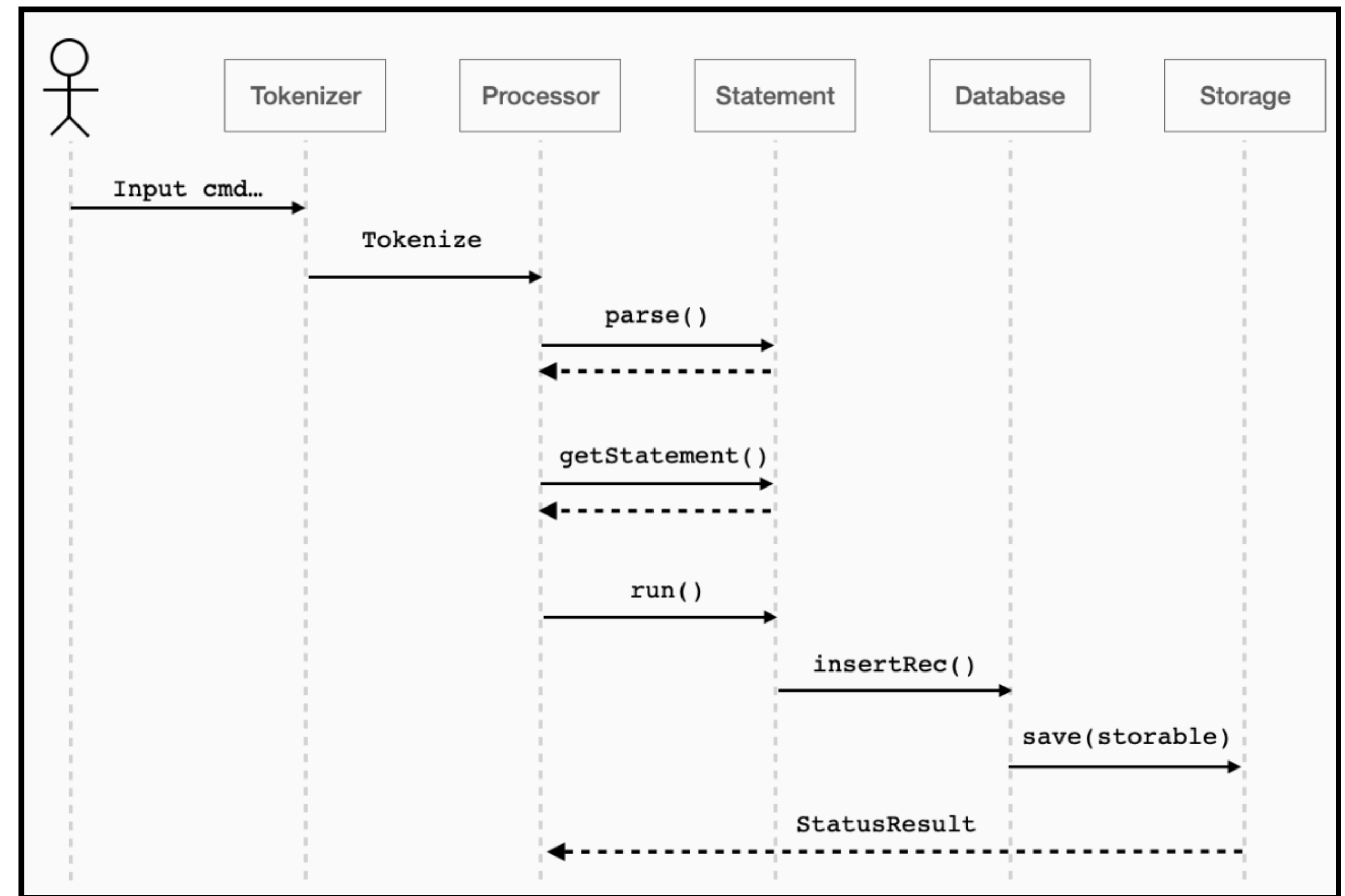
1. Database joins of tables (combining
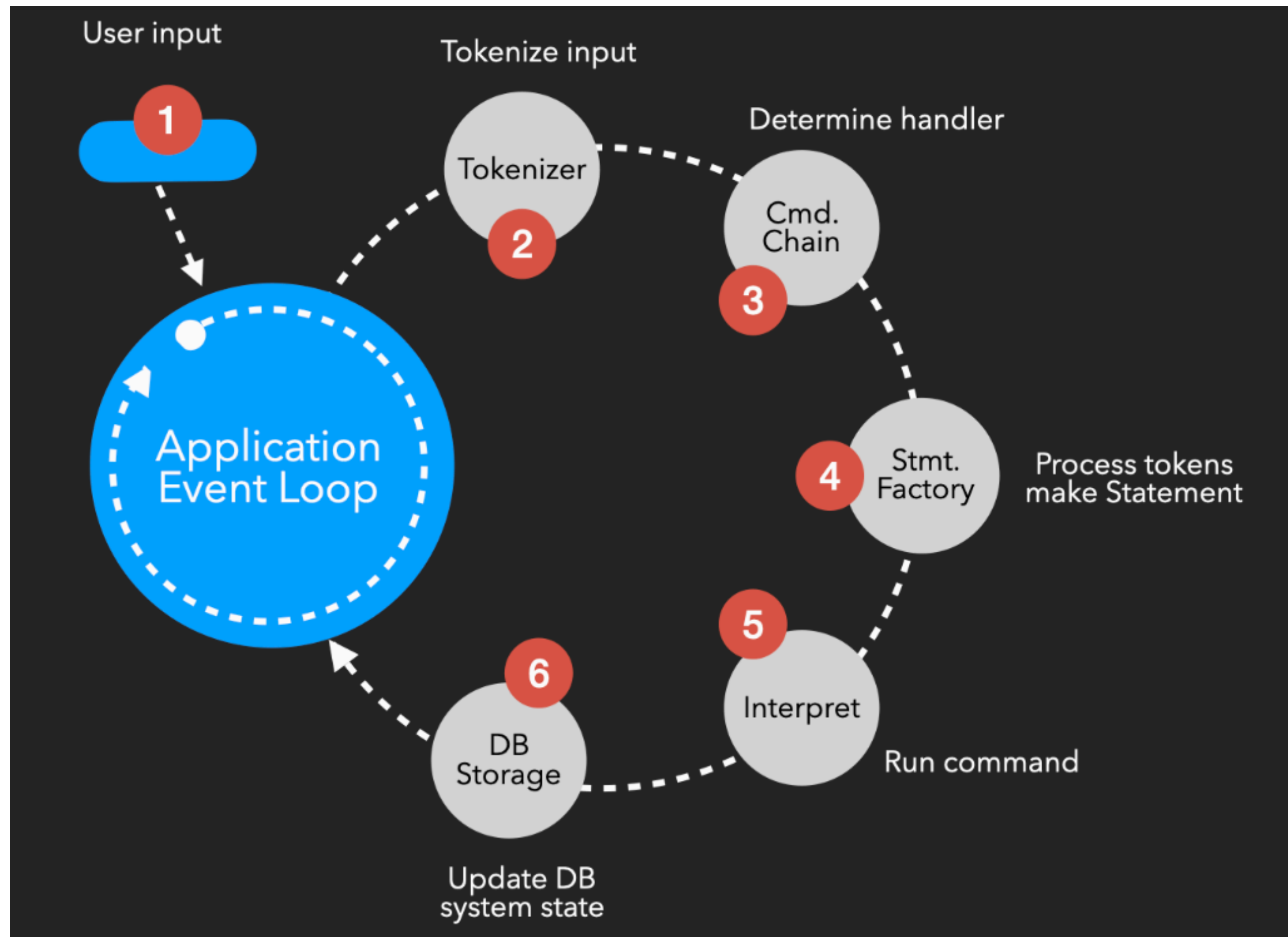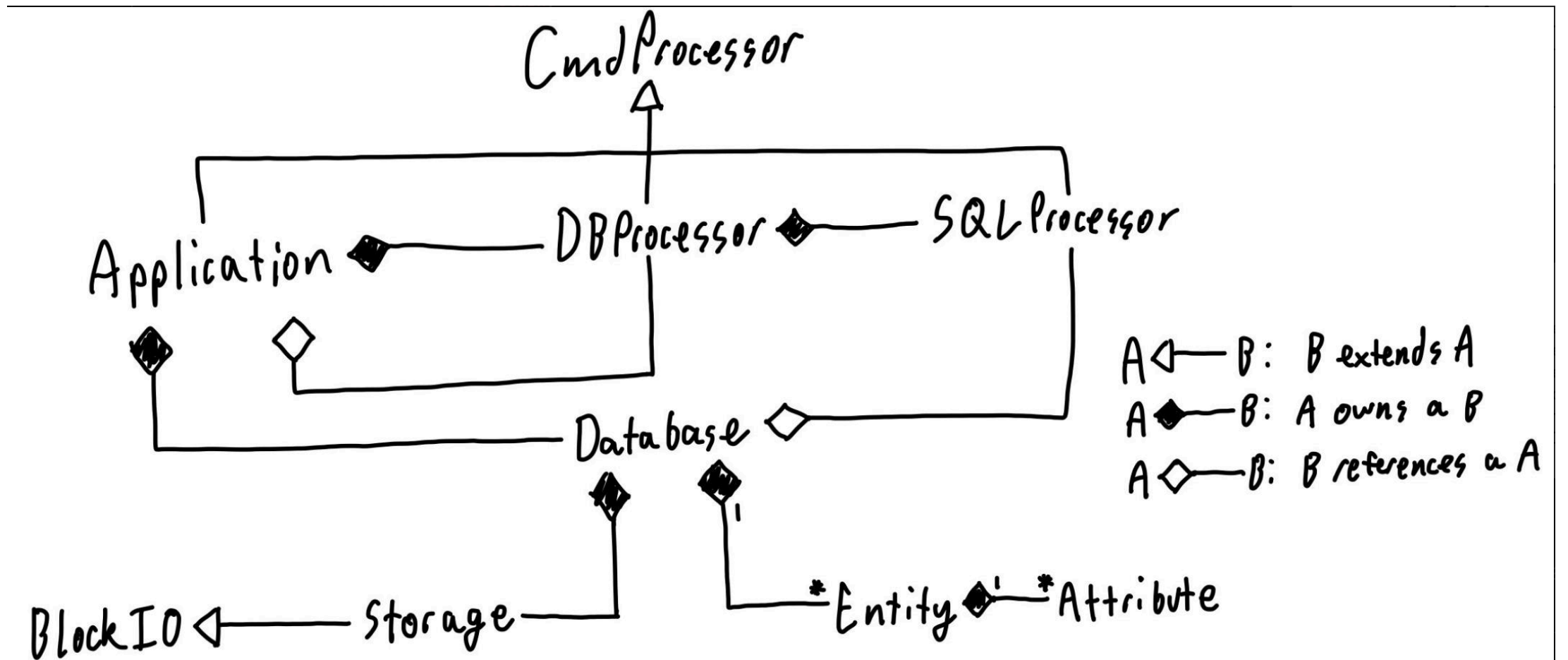
WEEK 9
CACHING AND PERFORMANCE

NEW CAPABILITIES

1. LRU cache

2. Caching rows (and collections)

The Application
The Top-Level Command Processor

A ◁— B: B extends A
A ◆— B: A owns a B
A ◇— B: B references a A

Scan, Tokenize, Parse
Interpreter Pattern

chars.	float	theFloat	=	theInt	*	3.14	;
tokens	type	identifier	op.	identifier	op.	num.	op.

Create Table Statement

CREATE — TABLE — ID — (— attribute declaration — , —) — ○

Attribute Declaration

ID — TYPE — attribute constraint(s) — ○

Attribute Constraint

PRIMARY — KEY — AUTOINCREMENT — ○

NOT — NULL

SELECT — field-name — , — FROM — TABLE — ○

WHERE — TERM — OP — TERM — AND, OR, NOT, IN

GROUP BY — field-name — ,

ORDER BY — field-name — ,

LIMIT — START, COUNT

JOIN — TABLE — ON — FIELD = FIELD

Storage
Storable, Blocks

0.	H	Database Header
1.	E	"Users" Entity
2.	I	Users Index
3.	F	Free block
4.	D	User Row 2
		...

Storage Blocks

Block 1

Block 2

...

Block N

Header   Payload

Random-Access Binary Stream

MVC Architecture
Mode, View, Controller

CQRS
Command Query Responsibility Segregation

DB Commands
Week 1-3

help	Help system ready!
version	Version 0.1
CREATE DATABASE foo;	Query OK, 1 row affected (0.01 secs)
DROP DATABASE foo;	Query OK, 0 rows affected (0.00)
| show databases; | ```
+--------------------+
| Database           |
+--------------------+
| foo                |
| bar                |
| wiltz              |
+--------------------+
3 rows in set (0.02 sec)
``` |
| use foo; | Database changed |
| dump database foo; | ```
+---------------+--------+--------------+
| Type          | Id     | Extra        |
+---------------+--------+--------------+
| Meta          | 0      |              |
+---------------+--------+--------------+
| Entity        | 1      | Users        |
+---------------+--------+--------------+
| Data          | 1      | Users        |
+---------------+--------+--------------+
| Free          | 0      |              |
+---------------+--------+--------------+
| Data          | 3      | Users        |
+---------------+--------+--------------+
5 rows in set (0.02 sec)
``` |

create table users (id int NOT NULL auto_increment primary key, first_name varchar(50) NOT NULL, last_name varchar(50));	Query OK, 1 row affected (0.002 sec)
CREATE TABLE tasks (id INT AUTO_INCREMENT PRIMARY KEY, title VARCHAR(100) NOT NULL, price FLOAT DEFAULT 0.0, due_date DATETIME, status BOOLEAN DEFAULT FALSE);	Query OK, 1 row affected (0.002 sec)
| show tables; | ```
+---------------------+
| Tables_in_mydb      |
+---------------------+
| users               |
| tasks               |
+---------------------+
2 rows in set (0.000025 sec.)
``` |
| drop table users; | Query OK, 0 rows affected (0.02 sec) |
| DESCRIBE tasks; | ```
+-----------+--------------+------+-----+---------+-------------------------------+
| Field     | Type         | Null | Key | Default | Extra                         |
+-----------+--------------+------+-----+---------+-------------------------------+
| id        | integer      | NO   | YES | NULL    | auto_increment primary key    |
| title     | varchar(100) | NO   |     | NULL    |                               |
| price     | float        | YES  |     | 0.0     |                               |
| due_date  | date         | YES  |     | NULL    |                               |
| status    | boolean      | YES  |     | FALSE   |                               |
+-----------+--------------+------+-----+---------+-------------------------------+
5 rows in set (0.000043 sec.)
``` |

DB Commands
Week 4-7

insert into users (first_name, last_name) values (David, He), (Rick, Gessner);

dump database foo;

```
+-----+--------+-----+
| #   | Type   | Id  |
+-----+--------+-----+
| 0   | meta   |     |
| 1   | entity |     |
| 2   | data   | 1   |
| 3   | data   | 2   |
+-----+--------+-----+
4 rows in set (0.00024 secs)
```

SELECT * from users;

```
+-----+------------+------------+
| id  | first_name | last_name  |
+-----+------------+------------+
| 1   | David      | He         |
| 3   | Rick       | Gessner    |
+-----+------------+------------+
3 rows in set (0.00231 sec)
```

SELECT first_name, last_name from users order by last_name;

```
+---------------+---------------+
| first_name    | last_name     |
+---------------+---------------+
| David         | He            |
| Rick          | Gessner       |
+---------------+---------------+
2 rows in set (0.00 sec.)
```

create table Users (id int NOT NULL auto_increment primary key, first_name varchar(50) NOT NULL, last_name varchar(50), age INT, zipcode INT);

insert into Users (first_name, last_name, age, zipcode) values (Anirudh, Swaninthan, 27, 92126), (Pu, Cheng, 30, 92127), (Stewie, Griffin, 1, 92100), (Chi, Lo, 26, 92100), (David, He, 24, 92100), (Rick, Gessner, 54, 92100);

SELECT * from Users where age>20 AND zipcode=92100 ORDER BY last_name;

```
+-----+------------+-----------+------+---------+
| id  | first_name | last_name | age  | zipcode |
+-----+------------+-----------+------+---------+
| 6   | Rick       | Gessner   | 54   | 92100   |
| 5   | David      | He        | 24   | 92100   |
| 4   | Chi        | Lo        | 26   | 92100   |
+-----+------------+-----------+------+---------+
3 rows in set (0.00 sec.)
```

SELECT id, first_name, last_name, zipcode from Users where zipcode>92120 order by zipcode LIMIT 1

```
+-----+------------+-----------+---------+
| id  | first_name | last_name | zipcode |
+-----+------------+-----------+---------+
| 1   | Anirudh    | Swaninthan | 92126  |
+-----+------------+-----------+---------+
1 rows in set (0.00 sec.)
```

UPDATE Users SET "zipcode" = 92127 WHERE zipcode>92100;

Query Ok. 2 rows affected (0.000087 sec)

DELETE from Users where zipcode>92124;

Query Ok. 1 rows affected (0.000023 sec)

Select * from Users;

show indexes

```
+---------------+---------------+
| table         | field(s)      |
+---------------+---------------+
| users         | id            |
+---------------+---------------+
1 rows in set (nnnn secs)
```

SHOW index id FROM Users;

```
+---------------+---------------+
| key           | block#        |
+---------------+---------------+
| 1             | 35            |
+---------------+---------------+
| 2             | 36            |
+---------------+---------------+
| 3             | 47            |
+---------------+---------------+
3 rows in set (nnnn secs)
```

drop table Users;

Query Ok. 5 rows affected (0.000023 sec)

DB Commands
Week 7-9

```
create table Authors (id int NOT
NULL auto_increment primary
key, first_name varchar(50) NOT
NULL, last_name varchar(50));

insert into Authors (first_name,
last_name) values (Stephen,
King), (JK, Rowling), (Truong,
Nguyen);

select * from Authors;
```

```
+----+------------+-----------+
| id | first_name | last_name |
+----+------------+-----------+
|  1 | Stephen    | King      |
|  2 | JK         | Rowling   |
|  3 | Truong     | Nguyen    |
+----+------------+-----------+
3 rows in set (0.00 sec)
```

```
create table Books (id int NOT
NULL auto_increment primary
key, title varchar(50) NOT NULL,
author_id INT);

insert into Books (title, author_id)
values (Harry Potter and the
Sorcerer's Stone, 2), (Harry
Potter and the Philosopher's
Stone, 2), (Harry Potter and the
Prisoner of Azkaban, 2), (Harry
Potter and the Chamber of
Secrets, 2), (Harry Potter and the
Goblet of Fire, 2), (Harry Potter
and the Order of the Phoenix, 2),
(Harry Potter and the Half-Blood
Prince, 2), (Carrie, 1), (The Dark
Tower, 1), (The Green Mile, 1),
(Wavelets and Filter Banks, 0);

select * from Books;
```

```
+----+------------------------------------+-----------+
| id | title                              | author_id |
+----+------------------------------------+-----------+
|  1 | Harry Potter and the Sorcerer's Stone     |        2 |
|  2 | Harry Potter and the Philosopher's Stone  |        2 |
|  3 | Harry Potter and the Prisoner of Azkaban  |        2 |
|  4 | Harry Potter and the Chamber of Secrets   |        2 |
|  5 | Harry Potter and the Goblet of Fire       |        2 |
|  6 | Harry Potter and the Order of the Phoenix |        2 |
|  7 | Harry Potter and the Half-Blood Prince    |        2 |
|  8 | Carrie                             |        1 |
|  9 | The Dark Tower                     |        1 |
| 10 | The Green Mile                     |        1 |
| 11 | Wavelets and Filter Banks          |        0 |
+----+------------------------------------+-----------+
11 rows in set (0.00 sec)
```

```
select last_name, title from Authors
left join Books on
Authors.id=Books.author_id;
```

```
+-----------+-------------------------------------------+
| last_name | title                                     |
+-----------+-------------------------------------------+
| Rowling   | Harry Potter and the Sorcerer's Stone     |
| Rowling   | Harry Potter and the Philosopher's Stone  |
| Rowling   | Harry Potter and the Prisoner of Azkaban  |
| Rowling   | Harry Potter and the Chamber of Secrets   |
| Rowling   | Harry Potter and the Goblet of Fire       |
| Rowling   | Harry Potter and the Order of the Phoenix |
| Rowling   | Harry Potter and the Half-Blood Prince    |
| King      | Carrie                                    |
| King      | The Dark Tower                            |
| King      | The Green Mile                            |
| Nguyen    | NULL                                      |
+-----------+-------------------------------------------+
11 rows in set (0.00 sec)
```

```
select last_name, title from Authors
right join Books on
Authors.id=Books.author_id;
```

```
+-----------+-------------------------------------------+
| last_name | title                                     |
+-----------+-------------------------------------------+
| King      | Carrie                                    |
| King      | The Dark Tower                            |
| King      | The Green Mile                            |
| Rowling   | Harry Potter and the Sorcerer's Stone     |
| Rowling   | Harry Potter and the Philosopher's Stone  |
| Rowling   | Harry Potter and the Prisoner of Azkaban  |
| Rowling   | Harry Potter and the Chamber of Secrets   |
| Rowling   | Harry Potter and the Goblet of Fire       |
| Rowling   | Harry Potter and the Order of the Phoenix |
| Rowling   | Harry Potter and the Half-Blood Prince    |
| NULL      | Wavelets and Filter Banks                 |
+-----------+-------------------------------------------+
11 rows in set (0.00 sec)
```