

# ECS 165A

# Milestone I

Group 4: Zhengyu Wu, Leran Chen, Tedder Lao, Xingyu Pan, Jiaming Mai



## What was accomplished and how?

02

---

Goal 1: Building an L-Store-based single-threaded storage database.

- Data Model

Goal 2: Adding SQL-like operations to the database.

- Query Interface

Goal 3: Achieving a faster L-Store Implementations.

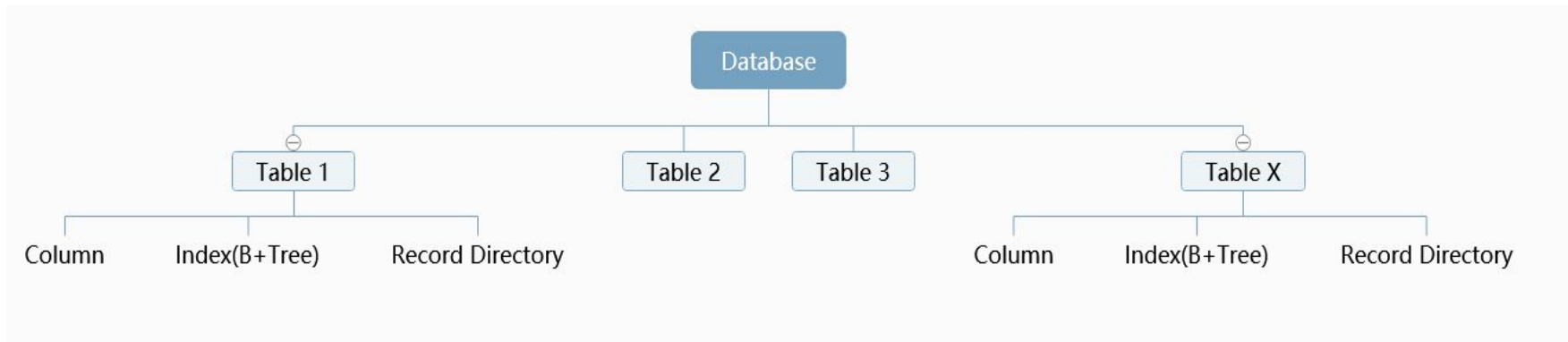
- B+ Tree & Index
- Bufferpool Management

# Presentation Highlights

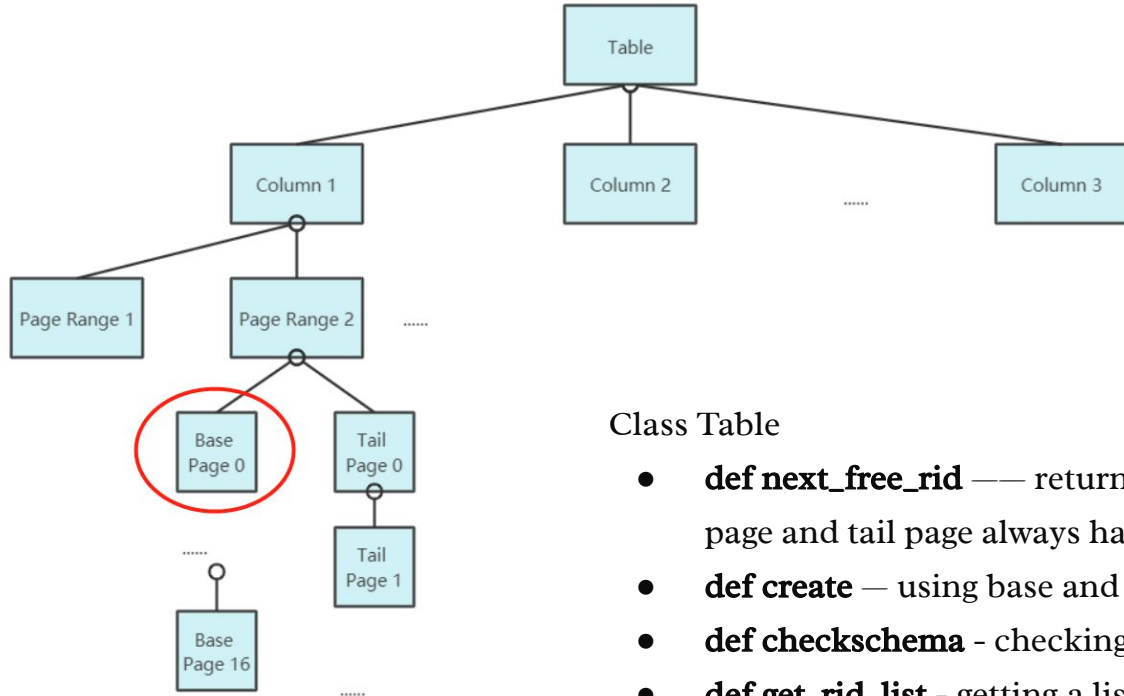
# Our Design & Solution

## Data Model

---



# Data Model - Table - Columns —



Implementation & Solution:

Table.py

Class Prange (Page Range)

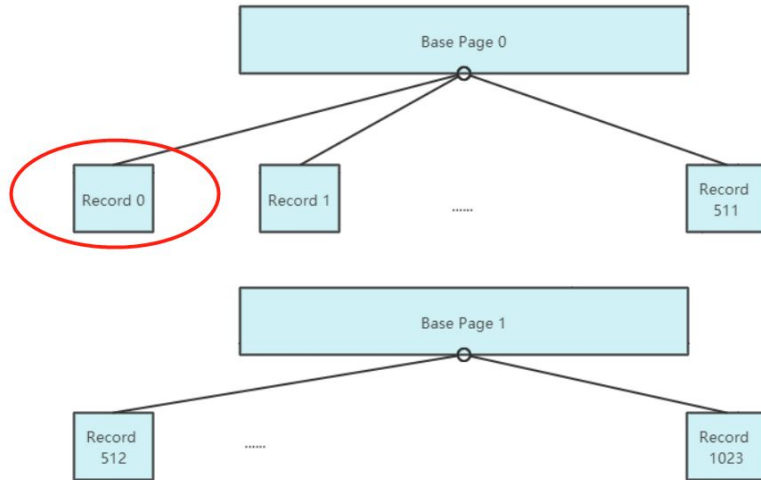
- `def append_page`
- `def insert_page_to`
- `def update_page_to`
- `def add_prange`

Class Table

- **`def next_free_rid`** — return next RID and make sure Base page and tail page always have different RID
- **`def create`** — using base and tail page to create page range
- **`def checkschema`** - checking schema between records
- **`def get_rid_list`** - getting a list of RID
- **`def get_data`** - returning specific record value

# Data Model - Base Page

---



Implementation & Solution:  
page. py

Class Page

- def create\_record
- def update\_record
- def read\_record
- def insert\_record

# Data Model - Record

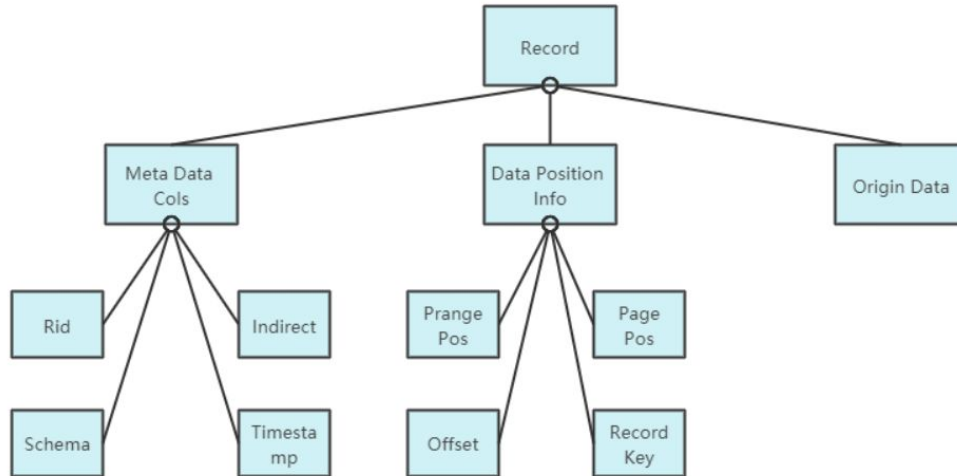
---

Implementation & Solution:

table.py

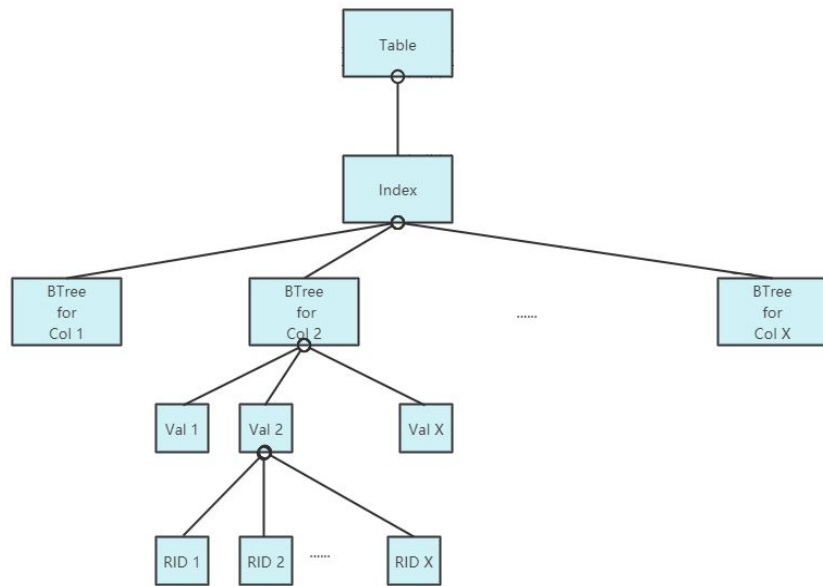
Class Record

- def get\_meta
- def create\_record
- def update\_record
- def read\_record



# Data Model - Table -Index

---



Implementation & Solution:

Index.py

- def locate
- def locate\_range

New:

- def insert - insert new index
- def delete - delete new index
- def update - update new index
- def create\_index - create new BTree

# Bufferpool Management

03

Example from our code:

```
"""  
def __init__(self, name, num_columns, Table_key):  
    self.name = name  
    self.Table_key = Table_key  
    self.num_columns = num_columns  
    self.prange_directory = {} # 'col_num': 'page_range_list'  
    self.page_directory = {} # 'RID': 'record obj'  
    self.index = Index(self)  
    self.prange_num = 0  
    self.free_brid = 0  
    self.free_trid = 0  
    self.rid_list = []
```

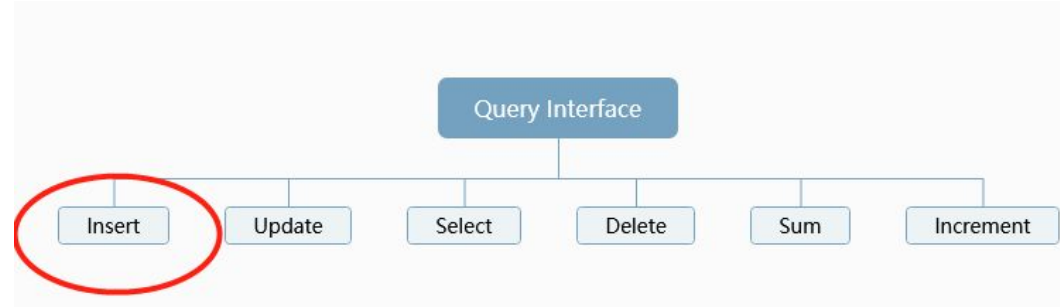
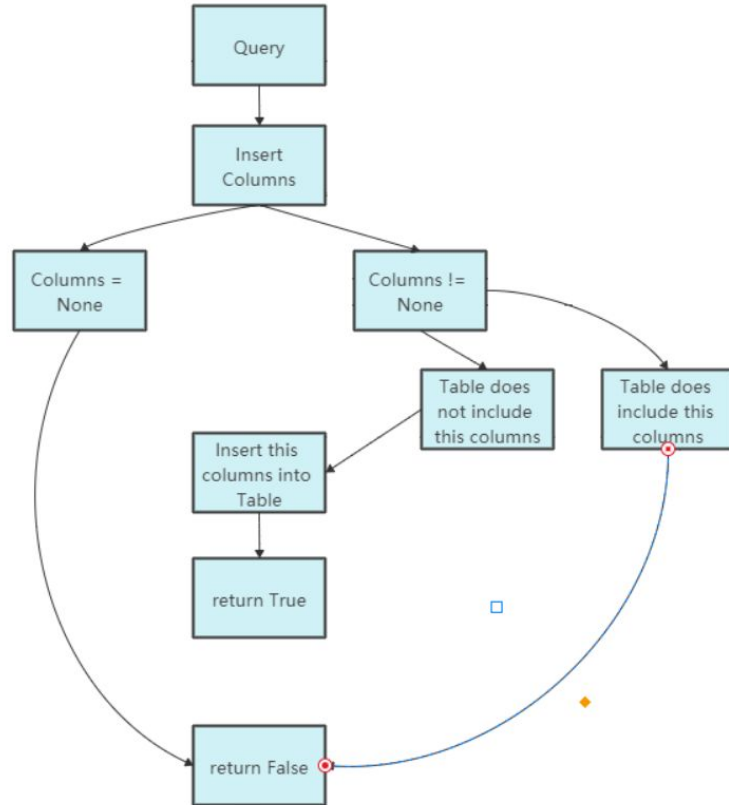
```
def add_prange(self, column):  
    b_page = Page(0)  
    t_page = Page(0)  
    prange = Prange(b_page, t_page, self.prange_num)  
    self.prange_directory.get(column).append(prange)
```

```
def update_page_to(self, ith_column, ith_prange):  
    prange = self.prange_directory.get(ith_column)[ith_prange]  
    prange.append_page(1)
```



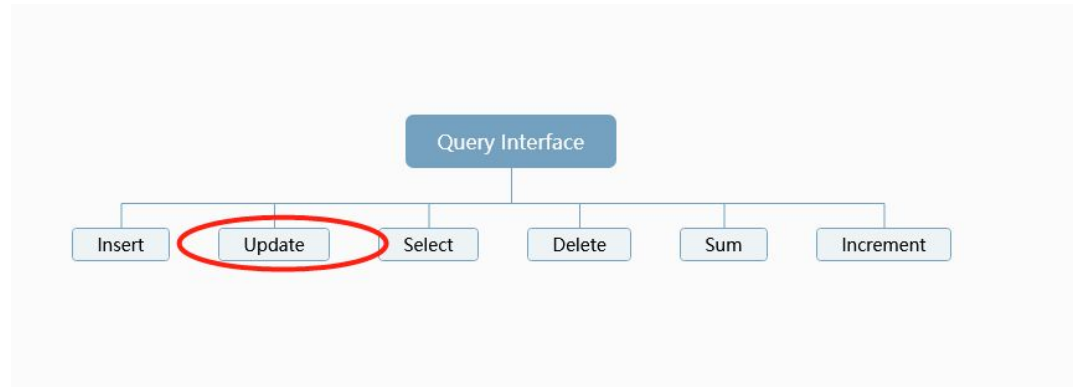
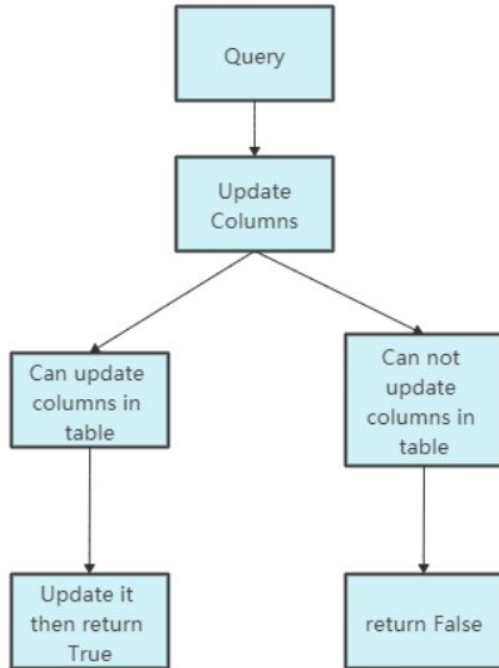
# Query Interface - Overview

---



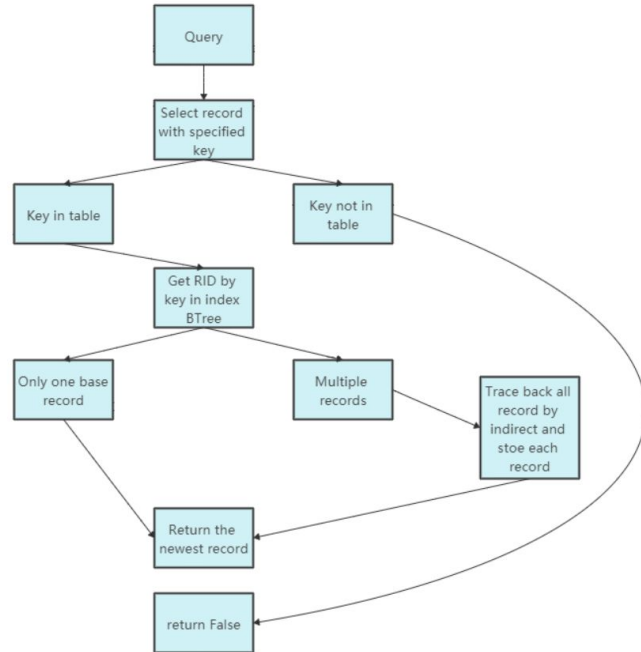
# Query Interface - Overview

---



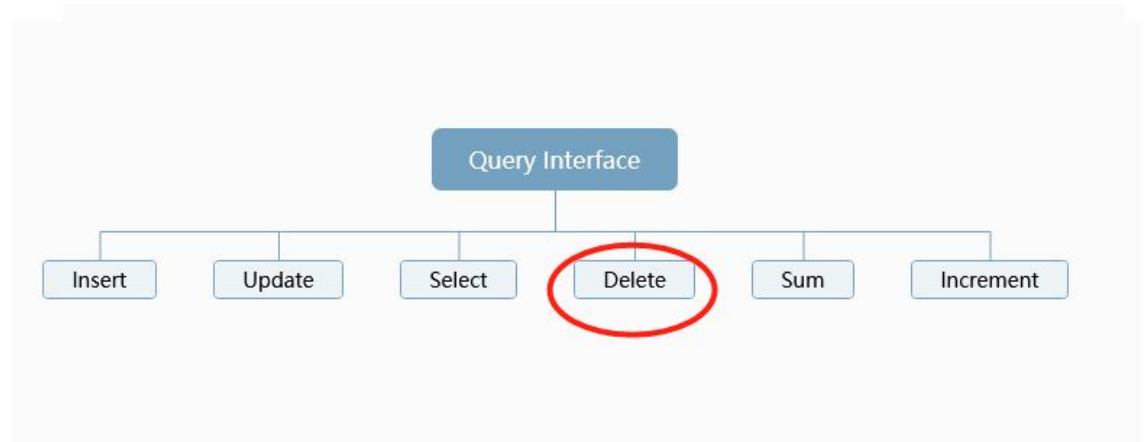
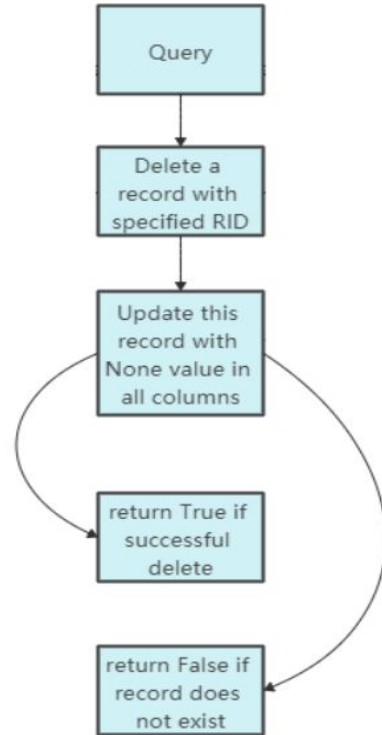
# Query Interface - Overview

---



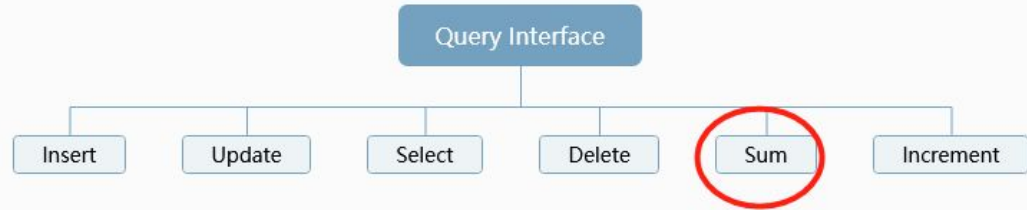
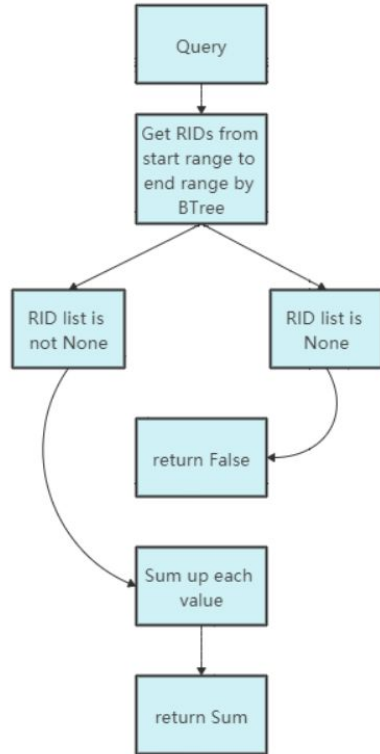
# Query Interface - Overview

---



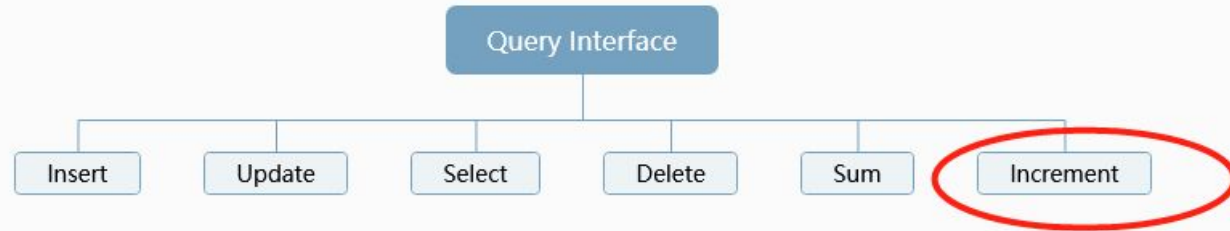
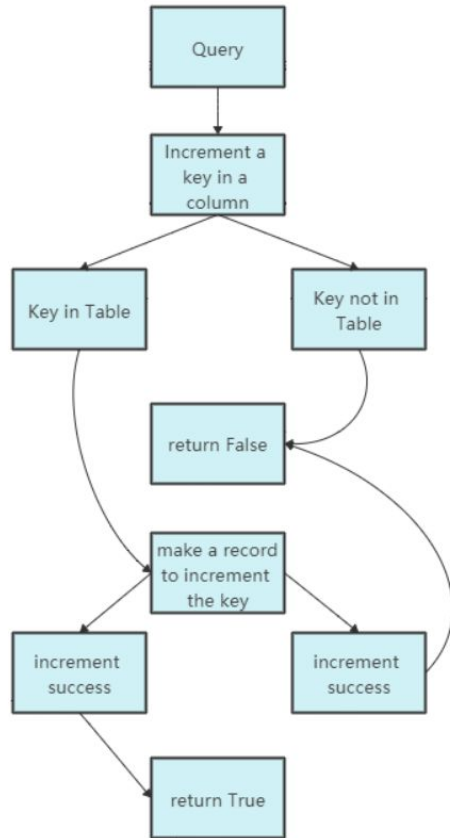
# Query Interface - Overview

---



# Query Interface - Overview

---



Thank you