香港中文大學（深圳）
The Chinese University of Hong Kong, Shenzhen

SCHOOL OF
DATA SCIENCE
數據科學學院

CSC3170
Tutorial 1

School of Data Science
The Chinese University of Hong Kong, Shenzhen

SCHOOL OF DATA SCIENCE

Outline

- Relational Model
  - Exercise
- SQL (Part A, and a little bit about Part B)
  - OJ (Online Judge System) Registration
  - Exercise

SCHOOL OF DATA SCIENCE

Relational Model

- Concepts:
  - Relation: Artist
  - Attribute:

name	year	origin

  - Tuple:

The Chainsmokers	2012	US

  - Schema:

name	year	origin

  - Instance:

name	year	origin
The Chainsmokers	2012	US
Imagine Dragon	2008	US
Coldplay	1997	UK

  - Domain: The set of allowable values

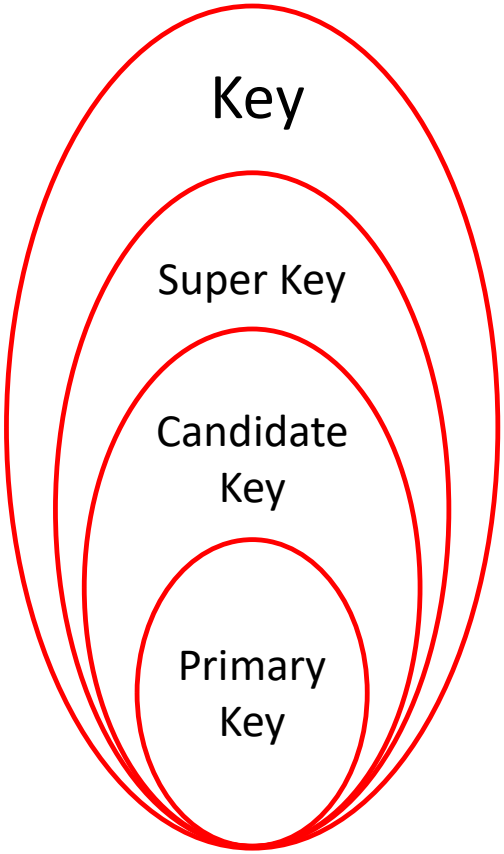
Artist (name, year, origin)

name	year	origin
The Chainsmokers	2012	US
Imagine Dragon	2008	US
Coldplay	1997	UK

SCHOOL OF DATA SCIENCE

Key

- Key: an attribute or a group of attributes
  - Primary key: uniquely identifies a single tuple
    - ~~Any~~ key that can uniquely identify a single tuple is a primary key ✗
  - Super key: Any set of attributes whose values, taken together, uniquely identify each row (tuple) of a table.
  - Candidate key: A minimal super key (that is, if AB is a super key, then ABC is a super key but not a candidate key).
  - Primary key: A specific candidate key we pick as the unique identifier.



SCHOOL OF DATA SCIENCE

Foreign Key

- A foreign key is a column or columns in a table that that are **linked to a primary key** in a different table.

Student

Student ID	Name	Department
12*******	Alan	SDS
12*******	Tom	SSE
12*******	Richard	SME

Foreign Key

Department

Department	Location	…
SDS	DY Building	…
…	…	…
…	…	…

Relational Algebra

name	year	origin
The Chainsmokers	2012	US
Imagine Dragon	2008	US
Coldplay	1997	UK

SCHOOL OF DATA SCIENCE

7 Fundamental operations:

- Select σ: Filters $\sigma_{year<2000}(Artist) \rightarrow$

name	year	origin
Coldplay	1997	UK

- Projection $\prod$: Pick attributes $\prod_{name}(Artist) \rightarrow$

name
The Chainsmokers
Imagine Dragon
Coldplay

- Union $\cup$
- Intersection $\cap$

} Tuples that appear in either/ both

- Difference $-$: Tuples that appear in the first but not the second
- Product $\times$: All combinations
- Join $\bowtie$: "Composite of Select and Product"
  - $artist \bowtie_{artist.name=company.name} company$
  - $\sigma_{artist.name=company.name}(artist \times company)$

name	company
The Chainsmokers	A
The Chainsmokers	B
Coldplay	A

SCHOOL OF DATA SCIENCE

Practice

Customer (name, age, gender)

Buys (name, drink)

Sells (store, drink, price)

Visits (name, store)

- Find all stores that are visited by at least 1 customer under 18;
- Find the names of all female customers that bought either apple juice or orange juice (or both);
- Find all stores that sells at least 1 kind of drink that Alex bought before and is more expansive than 10$.

Customer (<u>name</u>, age, gender)

Buys (<u>name</u>, drink)

Sells (<u>store</u>, <u>drink</u>, price)

Visits (<u>name</u>, <u>store</u>)

SCHOOL OF DATA SCIENCE

- Find all stores that are visited by at least 1 customer under 18;
  - $\prod_{store}(\sigma_{age<18}(Customer) \bowtie_{customer.name=visits.name} Visits)$
- Find the names of all female customers that bought either apple juice or orange juice (or both);
  - $\prod_{name}(\sigma_{gender='female'}(Customer) \bowtie \sigma_{drink='apple\ juice' \vee drink='orange\ juice'}(Buys))$
- Find all stores that sells at least 1 kind of drink that Alex bought before and is more expansive than 10$.
  - $\prod_{store}(\sigma_{name='Alex'}(Buys) \bowtie \sigma_{price>10}(Sells))$

SCHOOL OF DATA SCIENCE

SQL

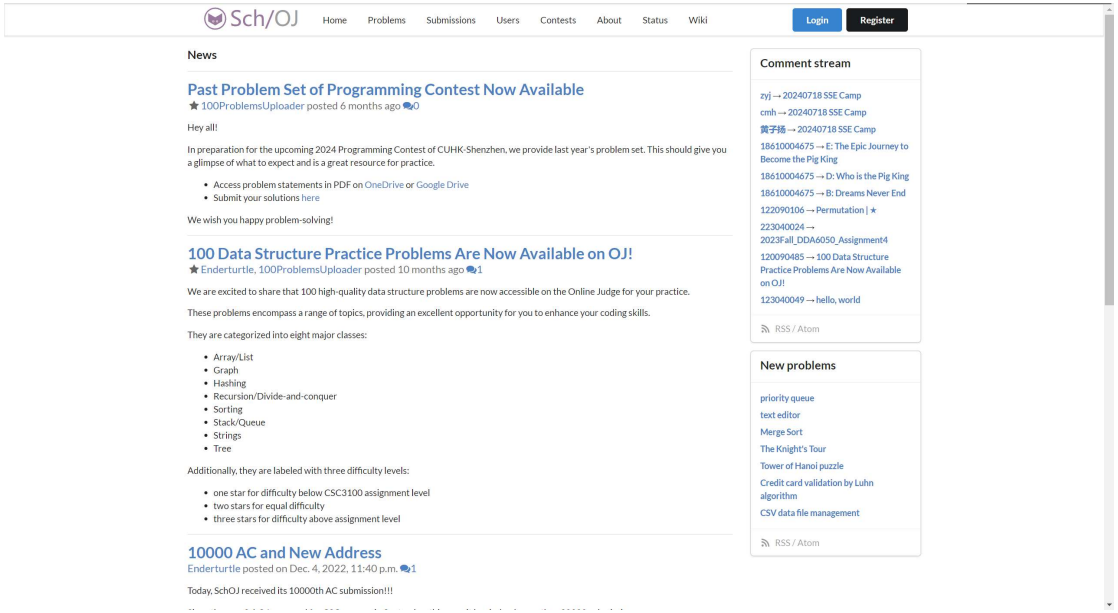
- Data Definition Language (DDL)
  - CREATE TABLE: define a table; DROP TABLE: delete a table;
  - ALTER TABLE: modify columns of a table;
  - Other keywords: (type), NOT NULL, PRIMARY KEY, FOREIGN KEY, etc.
- Data Manipulation Language (DML)
  - INSERT INTO: insert records; COPY: insert a large amount of records;
  - DELETE FROM / UPDATE: usually used with WHERE clause;
  - Other keywords: CASE – WHEN – THEN – ELSE, etc.
- Data Query Language (DQL)
  - SELECT (attributes) AS (renaming)
    FROM (relations)
    WHERE (conditions)
- Aggregates
  - `AVG(col)` → Return the average col value.
  - `MIN(col)` → Return minimum col value.
  - `MAX(col)` → Return maximum col value.
  - `SUM(col)` → Return sum of values in col.
  - `COUNT(col)` → Return # of values for col.

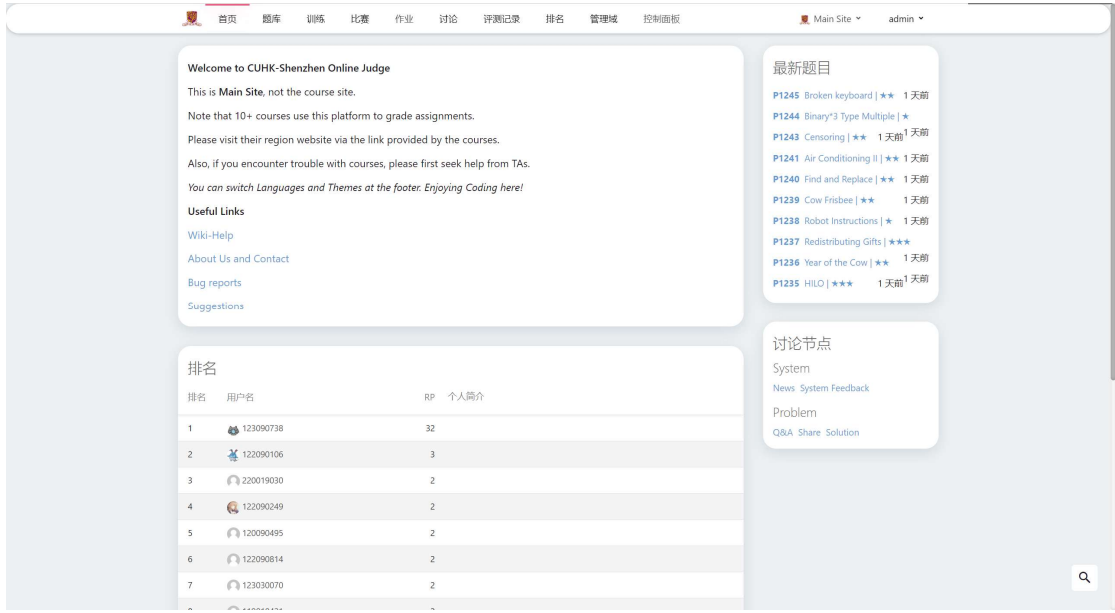
SCHOOL OF DATA SCIENCE

OJ System

- OJ link: http://oj.cuhk.edu.cn/d/csc3170_2025_fall/ (neither "https://oj.cuhk.edu.cn" nor "https://oj.cuhk.edu.cn/d/csc3170_2025_fall" works! You need to copy the whole link!)
- How to register:
  - Case 1: You have the experience of using **new OJ** -> Directly login
  - Case 2: You have never used new OJ system:
    - Step 1: Access http://oj.cuhk.edu.cn/d/csc3170_2025_fall/
    - Step 2: Click "Login" button at the top-right corner
    - Step 3: Click "Forgot password or username?"
    - Step 4: Enter your email (student_id@link.cuhk.edu.cn) and send reset email
    - Step 5: Check your email and reset your password, and then login (username is your student id instead of your email).

OJ System





Former OJ

New OJ

OJ issue: USTF Tim timweicuhksz@outlook.com

SCHOOL OF DATA SCIENCE

Practice

- You are encouraged to complete the problem on OJ (Sample Problem C-E).

You are given a database for a bookstore. The database has a table named “Books” with the following attributes:

‘book_id’ (integer)
‘title’ (varchar)
‘author’ (varchar)
‘genre’ (varchar)
‘price’ (decimal)
‘stock_quantity’ (integer)

Write a SQL query to:

1. Find all books written by “Mark Twain” that are in the genre of “Fiction”;
2. Calculate the total value (num * price, named as ‘total_value’) of all “Textbook” books in stock;
3. Update the stock quantity by adding 5 to the current stock for the book title “The Great Gatsby”.

SCHOOL OF DATA SCIENCE

Practice

Write a SQL query to:

1. Find all books written by “Mark Twain” that are in the genre of “Fiction”;

```
SELECT * FROM Books
WHERE author = ‘Mark Twain’ AND genre = ‘Fiction’
```

2. Calculate the total value (num * price, named as ‘total_value’) of all “Textbook” books in stock;

```
SELECT SUM(price * stock_quantity) AS total_value
FROM Books
WHERE genre = ‘Textbook’
```

3. Update the stock quantity by adding 5 to the current stock for the book title “The Great Gatsby”.

```
UPDATE Books SET stock_quantity = stock_quantity + 5
WHERE title = ‘The Great Gatsby’
```

SCHOOL OF DATA SCIENCE

Q&A

Thanks to the previous CSC3170 teaching team from which part of the content was sourced.