

# Lab 5: Database Design and Selecting a Primary Key

## What you should takeaway from this lab?

- Definitions: 1) primary key 2) surrogate key
- How should I choose a primary key for a table?
- For the same table, how different primay key strategies affect the performance of the database at large scale?
- What about foregin key strategy?

Operation	Time Complexity
INSERT	
GET	

## A simple database for our lab use.

Let's imagine that we have a LeetCode Contest that all students in Northeastern Universtiy can participate in.

- The contest run 24 hours.
- The estimated number of students is 1000.
- The contest last for 12 hours.
- There will be 10,000 commits in total.

We want to record the records in a table called **records**:

RecordID	Student	ProblemID	PassOrFail	SubmitTime	Runtime
1	Fac	111	true	11:00	12
2	Lin	222	true	12:00	17
3	Ama	222	false	11:00	17
4	Goo	333	false	12:00	12

- RecordID: int NOT NULL AUTO\_INCREMENT
- Student: varchar(255) NOT NULL
- ProblemID: int

- PassOrFail: boolean
- SubmitTime: time
- Runtime: int

## What are the primary keys you can choose?

- What should I choose for my primary key?  
<http://web.archive.org/web/20150511162734/http://databases.aspfaq.com/database/what-should-i-choose-for-my-primary-key.html>
- Google other articles about “primary key” if you are still confused.

Please list your primary key strategies somewhere you can refer to :

- Option 1:
- Option 2:
- Option 3:
- ...

Please make assumptions of the performance of each strategy in your mind or write them down.

## Try different primary key strategies and analysis their performances and why?

- Example 1: use a auto increment record ID as primary key
- Example 2: use a [Student, ProblemID, SubmitTime] as primary key.

## Test your primary key strategies's performance with *mysqlslap*

[Official Documentation on mysqlslap](#)

- Step 1: Open your MySQL Workbench
- Step 2: Create a database connection on your local host port 3306.
- Step 3: Open **mysqlslap** on your terminal and execute the following command line:  
 (The following is a windows command. Mac users can take as a reference.)

Please figure out what each flag means before executing the command.

```
PS C:\Program Files\MySQL\MySQL Server 8.0\bin>.mysqlslap --concurrency=1 --iteration
s=1 --create='C:\Users\jxyzs\Desktop\create_table1.sql' --query='C:\Users\jxyzs\Desкто
p\fake_records.sql' --password --delimiter=";" --no-drop
```

- Step 4: Wait ..... Wait ..... until you see a report on your command line
- Step 5: Go to your MySQL Workbench, open the local connection, open **Schemas** on the left

column (It's beside **Administration**). refresh SCHEMAS. You will see **mysqlslap** as one of your schemas. Inspect the table called **records**.

- Step 6: drop schema **mysqlslap**
- Step 7: Try the same command line with **create\_table2.sql** and check both the command line report and table **records** in your MySQL Workbench. You would see the difference. Please think about their impact on your project if the number of records is in millions.
- Step 8: Explore yourself and think about different primary key strategies' impact on other types of queries in terms of time complexity and space complexity.