_____

# Final Take-Home Exam: Time Travel Queries in L-Store
## ECS 165A - Winter 2022

The main objective of this milestone is to implement **Time Travel Queries:** retrieve an older version of the record from the **L-Store** [Paper, Slides]. You can implement time travel queries as an extension to your current implementation of Milestone 1, Milestone 2, or Milestone 3. If you are extending your **Milestone 1** to implement the time travel queries, the **maximum grade** you can obtain on the final take-home exam **will be 80%**. Likewise, if you're extending your **Milestone 2,** the **maximum grade will be 90%**.
**Bonus:** Kindly note that the fastest L-Store implementations (the top three groups) will be rewarded.

*Think Long-term, Plan Carefully.*
*Be curious, Be creative!*

# Time Travel Queries
As we update the records in the database, it is important to retain the previous versions of the records to enable the analysis of historical records. With the time travel queries, the database should provide the option to retrieve the earlier version of the record. For this, you will be required to retain and traverse the **tail records backward** to fetch the appropriate version of the record.

*Conceptual Example* (not a representation of base vs. tail page design)
Assume you have the following records that were initially inserted in your L-Store.

Student Table

| ID (Primary Key) | fee_paid |
|---|---|
| 1 | 300 |
| 2 | 600 |
| 3 | 200 |

The sum of fee_paid of **version 0** (the latest version) must now be 300+600+200 = 1100.

_____

Suppose that every record is updated once, we will have like below,

<div align="center">

Student Table

| ID (Primary Key) | fee_paid |
|:---:|:---:|
| 1 | 100 |
| 2 | 500 |
| 3 | 200 |

</div>

The sum of fee_paid of **version 0** (latest version) must now be 100+500+200 = 800.
The sum of fee_paid of **version -1** (previous version) must be 1100.

Both the select and sum queries, intuitively, fetch the appropriate previous version of records. You will be required to implement select and sum queries that return appropriate results given the relative version number. If the given relative version does not exist, e.g., relative version -2 in the given example, the oldest version of the record must be used (-1 in the given example).

# Query semantics
You are expected to implement the following two functions in the *Query* class:
- query.select_version - Works like query.select, but accepts one more parameter, referred to as ***relative version (rv)*** to retrieve a corresponding version of the record, for example,
  - *rv = 0* returns the latest version,
  - *rv = -1* returns the previous version,
  - *rv = -k* returns the $k^{th}$ previous version, if the record has not been updated $k^{th}$ times, then the oldest version of the record (the original version) is returned.
- query.sum_version - Works like query.sum, but similar to query.select_version accepts one more parameter ***relative version (rv)*** to retrieve the sum of a corresponding version of the record column.

**Instructor:** Mohammad Sadoghi
**TAs:** Sajjad Rahnama
Shesha Vishnu Prasad

**Due Date:** March 16, 2022
**Submission Method:** Canvas
**Score:** 20%
_____

# Milestone Extension Penalties

You are allowed to extend your Milestone 1, Milestone 2, or Milestone 3 to implement Time Travel Queries. However, the maximum grade you can obtain from extending the chosen Milestone is:

- Extending Milestone 1: 80%
- Extending Milestone 2: 90%
- Extending Milestone 3: 100%

# Implementation

We have provided a code skeleton that can be used as a baseline for developing your project. This skeleton is merely a suggestion and you are free and even encouraged to come up with your own design.

**Deliverables/Grading Scheme: What to submit?**
This is an **individual** take-home exam that must extend **your own L-Store project**. Each person must write and submit their own code. However, you are allowed to discuss with your teammates the general design ideas. Grading of the final exam will be purely based on auto-marker. There are three **sample extended testers** provided with code skeleton which correspond with your choice of extending from Milestone 1, Milestone 2, or Milestone 3; the final auto-marker may include additional test cases. You are expected to submit your individual assignment before the due date.

**Late Policy**
No late submissions will be accepted.

**Course Policy**
In this class, we adopt the UC Davis Code of Academic Conduct available here.

**Disclaimer**
The external links and resources that are being provided on this handout serve merely as a convenience and for informational purposes only; they do not constitute an endorsement or approval of their products, services, or opinions of the corporation or organization, or individual. As a student, developer, or researcher, it is your sole responsibility to learn how to assess the accuracy and validity of any external site. This is a crucial skill in the age of the Internet, **where anyone can publish anything!**

_____

_____

**Changelog:**

Milestone Handout Version v1: March 10, 2022 (initial posted version)