

# ACM-DB Lab3

---

This is the answer document for lab3.

## Design Choices

---

1. Exercise 1:
  - Simple nested loop for join.
2. Exercise 2:
  - Use HashMap to record aggregation results.
3. Exercise 3:
  - InsertTuple: find HeapPage with empty slots and insert, or create a new HeapPage.
  - DeleteTuple: just call HeapPage.deleteTuple.
4. Exercise 4:
  - Call DbFile.insertTuple to insert and DbFile.deleteTuple to delete.
  - Add dirty pages into BufferPool pages.

## Changes to API

---

- A new static method `merge()` in Tuple.java, used to merge two tuples into a new one.
- New method `getPage()`, `getEmptyPage()` in HeapFile.java, similar to these in BTreeFile.java

## Missing Components

---

I believe None

## Time Spent

---

1 week

- Stuck on HashEquijoin.java for some time, confused about what it is.
- Stuck on a weird bug in BufferPoolWriteTest.java where a changed HeapPage can't be found during the check process. It turned out that these changed pages are not recorded in BufferPool in my implementation since they're created manually. Therefore, we need to add dirty pages into BufferPool in BufferPool.insertTuple.