Today ...

- Dynamic SQL intro

Homework

- HW 8 out
Using SQL within a Program

Most of the world’s SQL is generated from software applications!

- Typically through libraries or frameworks
- SQL statements reference “host variables” ... program vars passed as args

SQL query results can be large

- Don’t want programs to “hold” large query results
- So DBMSs support “Result Sets”
  - Similar to I/O streams
  - And to “iterators” ... like hasNext(), next() in Java

Different approaches for using SQL within a program

“Dynamic” SQL ... what we’ll discuss
  - SQL statements as strings, passed to API calls

Object-Relational Mappings (ORMs) ... (e.g., Hibernate, Django)
  - automatic mapping to/from tables and class objects

“Embedded” SQL ... not as common today
  - preprocessor for SQL “embedded” in the programming language
Result Sets

Features vary from one DBMS to another

- Move forward only
- Move forward or backward one row at a time
- Move to arbitrary locations
- Modify or delete records that have been retrieved

Q: What are some advantages/disadvantages?

Result Set placement

- Placed BEFORE the first row of the result
- Getting the next result (logically) “pulls”\(^1\) the value from the DBMS
- Some notification of when last record is obtained

Our plan …

- Briefly look at how this works in three different languages
- Start with Java and JDBC
- Then look at Python and C++
- Then more examples
- Finally, basic HTML w/ PHP (on barney)

\(^1\)might not be implemented this way