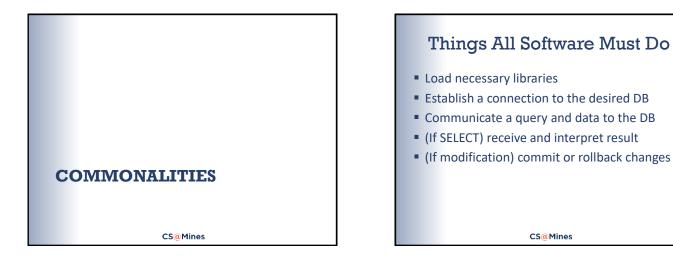
## CSCI 403 Database Management

20 – Programming Against the Database

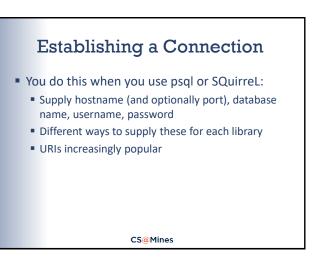
#### This Lecture

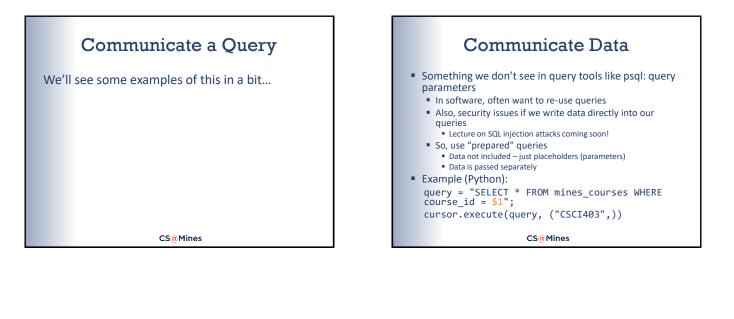
An overview of various approaches to writing software programs that connect to a SQL database.

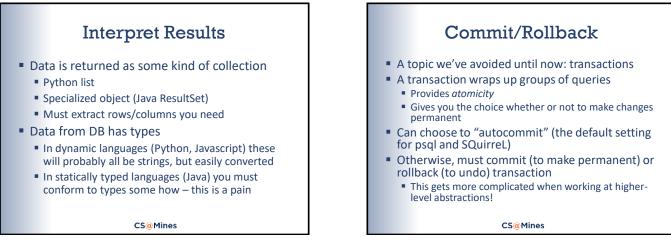
CS@Mines

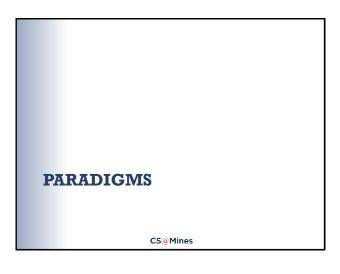














## SQL Is the Foundation

- At the base level of abstraction is SQL
  - Libraries transmit SQL strings and data to DB
  - Results returned in dynamic structures
  - No lower level access (this may be surprising)
- Other libraries build on top
  - E.g., higher level abstractions translate to SQL first

CS@Mines

#### Example (Python)

connection = ...
cursor = connection.cursor()
query = "SELECT \* FROM
mines\_courses WHERE course\_id =
'CSCI403'"

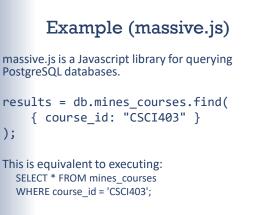
cursor.execute(query)
results = cursor.fetchall()

CS@Mines

## **Functional Mapping**

- SQL is not part of (most) programming languages
  - So need some way to communicate SQL
  - Base level is creating string SQL commands some people find this messy/unintuitive
- Simple SQL can be replaced with function calls
  - Works for many use cases
  - Must fall back to raw SQL for complex queries

CS@Mines



**CS@**Mines

#### **Object-Relational Mapping**

- Object-oriented programming
  - Very popular programming paradigm
  - Outside the scope of this course 😕
- Object-relational "impedance mismatch"
  - Objects store relationships with other objects
  - SQL databases allow ad-hoc relationships
  - This difference causes friction at the interface
     Object oriented programmers want to deal with
- Object-oriented programmers want to deal with data as objects
  - Special purpose "OODBMSes", not very popular/successful
  - ORM tries to treat data in RDBMS as objects

CS@Mines

### Example (SqlAlchemy)

SqlAlchemy is a Python ORM; this snippet really doesn't show all that goes into an ORM.

Guest lecture on this topic in a few weeks!

session.query(Course).filter(Course .id == 'CSCI403').all()

CS@Mines

# Next Time

Programming against the database in Python.

CS@Mines