

Aug 27 Queries 1-row case

```
EXEC SQL SELECT E.name
INTO :name
FROM EMP E
WHERE E.ssn = :ssn;
```

Queries many row case

Use a cursor

```
EXEC SQL DECLARE my_cursor CURSOR FOR
SELECT E.name, E.age
FROM EMP E
WHERE E.age > 30;
```

```
EXEC SQL OPEN my_cursor;
```

```
EXEC SQL FETCH my_cursor INTO
:name, :age;
```

```
EXEC SQL CLOSE my_cursor;
```

gets one row

use while and do SQLCODE = 02000 to get all rows

Prepared Statements

```
char stmt = "DELETE FROM EMP
WHERE SALARY < 100000";
```

```
EXEC SQL PREPARE
:stmt FROM :stmt;
EXEC SQL :stmt;
```

~~Needed~~ Needed. Header `#include <sqlca.h>`  
Connecting to a database

```
EXEC SQL CONNECT  
:login IDENTIFIED BY :password  
USING :connectString;
```

When done:

if → EXEC SQL COMMIT RELEASE;

things went ~~well~~ correct  
EXEC SQL ROLLBACK; ← if things failed

SEE examples on web  
from previous time taught ISFB  
and when taught at Clark  
University.

## SQLJ

Very much liked embed sql but for  
JAVA.

Two steps to compile:

sqlj file.sqlj

javac file.java.

Need to import:

```
java.io.*
java.sql.*
sqlj.runtime.*
sqlj.runtime.ref.*;
oracle.sqlj.runtime.*;
```

To connect:

```
Oracle.connect(connectString, login, password, false)
```

To close connection

```
Oracle.close();
```

To execute single statements

```
#sql { statement };
```

Ex

```
#sql { UPDATE EMP SET age=:age
      WHERE SSN=:SSN };
```

```
#sql { COMMIT RELEASE }
```

cursors

Iterators (like ~~prepared statements~~)

```
#sql iterator Book (String title, Float price)
```

```
Book books;
```

```
#sql books = {
```

```
  SELECT title, price INTO :title, :price
  FROM Books WHERE author=:author };
```

```
while (books.next())
```

```
  { System.out.println (books.title + " " +
                        books.price);
```