

Systems for Design and Implementation

Outline

- JDBC
- Reflection - cont.

JDBC

java.sql

1. Key interfaces

- Driver
- Connection
- Statement/PreparedStatement
- ResultSet

pgAdmin

IntelliJ Idea Database Tools

2. Connecting to a database

The JDBC URL

jdbc:postgresql://localhost:5432/catalog

Getting a Database *Connection*

DriverManager.getConnection(...)

postgresql-42.2.11.jar

compile **group: 'org.postgresql', name: 'postgresql', version: '42.2.11'**

3. Getting a *Statement/PreparedStatement*

~~connection.createStatement()~~

connection.prepareStatement(sqlString)

preparedStatement.setXXX(paramIndex, paramValue)

//paramIndex starts with 1

4. Getting a *ResultSet*

preparedStatement.executeQuery()

Executing a statement

preparedStatement.executeUpdate()

Getting Data from a *ResultSet*

```
while(resultSet.next()){
    XXX value = resultSet.getXXX(columnName)
    //or
    XXX value = resultSet.getXXX(columnIndex)
}
```

?Describe what is happening in the following situations

```
1. ResultSet rs = stmt.executeQuery("select count(*)
   from student");
2. if (rs.next()){ int id = rs.getInt(0);}
```

```
1. ResultSet rs = stmt.executeQuery("select * from
   student where name = 'does not exist'");
2. rs.next();
3. int id = rs.getInt(1);
```

```
1. ResultSet rs = stmt.executeQuery("select count(*)
   from student");
2. int id = rs.getInt(1);
```

5. Closing DB resources

How many resources are closed in the following code?

```
try (Connection conn =
    DriverManager.getConnection(url);
    Statement stmt = conn.createStatement();
    ResultSet rs = stmt.executeQuery("...")){

    if (rs.next()){System.out.println("hello1");}
    ResultSet rs2 = stmt.executeQuery("...");
    int num = stmt.executeUpdate("...");
}
```

jdbc

Reflection

- a language's ability to inspect and dynamically call classes, methods, attributes at runtime
- applications: IDEs, debuggers, testing frameworks etc

`java.lang.Class`

`java.lang.reflect`

[reflectionclassdiagram](#)

[reflection](#)