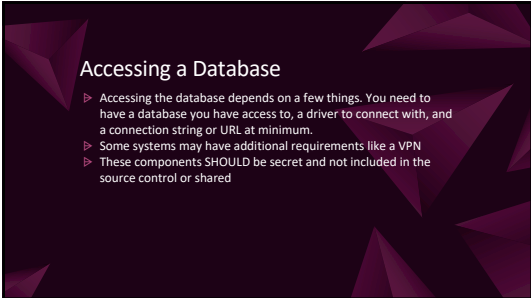


1



2



3

Driver

- The driver is the vendor specific software that allows your Java program to talk to the specific database.
- The driver can either be a jar file(s) local to the project or handled with a dependency management tool like Maven
- You NEED this!
- Do this BEFORE configuring your connection string

4

Connection String

- This is the URL to access the database. Depending on the database vendor the components and syntax will differ.

Components include:

- Protocol
- Hostname
- Port
- User
- Password
- SID

5

MySQL Connection String Example

```
MySQL Connection String Example
this.connectionMySQL = "jdbc:mysql://localhost/"
    + DataSecrets.SQL_DATABASE + "?users="
    + DataSecrets.USERNAME_SQL + "&password="
    + DataSecrets.PASSWORD;
```

6

Oracle Connection String Example

```
Oracle Connection String Example
this.connectionString = "jdbc:oracle:thin:"
+ DatabaseSecrets.USERNAME
+ "/"
+ DatabaseSecrets.PASSWORD
+ "@"
+ DatabaseSecrets.HOST_NAME
+ ":"
+ DatabaseSecrets.PORT
+ ";";
+ DatabaseSecrets.ORACLE_SID;
```

7

MongoDB Connection String Example

```
MongoDB Connection String Example
this.connectionMongoDB = "mongodb+srv://"
+ DatabaseSecrets.USERNAME_MONGO
+ ":" + DatabaseSecrets.PASSWORD_MONGO
+ "@"
+ DatabaseSecrets.MONGO_DB
+ "?retryWrites=true&majorityAvailability="
+ DatabaseSecrets.MONGO_DB;
```

8

SQLException

- We are able to get a bit more information from a SQLException than the message from a regular Exception
- SQLState
 - The state of the database. For example 42000 is a syntax error in the query
- ErrorCode
 - This is the database vendor specific (Oracle, MariaDB, etc) error code that is thrown

9

Indices

- ▶ Unlike native Java data types the indices of MOST database associated components start at 1!!!!
- ▶ This means if you are linking an index based structure like an array, ArrayList, or Vector you will need to check and/or modify the index value appropriately!

10

Connection

- ▶ The Connection is how your Java program communicates with the database. Their access should be limited to only when active to support stability, security, and good coding practice.
- ▶ I recommend using a private static method for maintaining secure access to the Connection
- ▶ Connections are accessed via the static factory method DriverManager.getConnection when supplied with the Connection String

11

Oracle Connection Example

```
private static Connection connectToOracle(Controller app, String connectionString)
{
    Connection myOracleConnection = null;
    try
    {
        myOracleConnection = DriverManager.getConnection(connectionString);
    }
    catch (SQLException error)
    {
        app.handleError(error);
    }
    return myOracleConnection;
}
```

12

Statement

- This is initialized based on the Connection and should be closed when finished!!
- If you want to more easily process the associated ResultSet, initialize it with parameters supporting scroll insensitivity and read only concurrency
- Used to execute the query

13

ResultSet

- This is the data type for the results of any executed Statement query.
- By default it supports only one traversal from front to back
 - Use parameters to modify this!
- Not all values are retrieved at once! The total size of the result is not available until the entire ResultSet is processed!!

14

Process ResultSet

- Use a while loop based on the .next () call on the ResultSet
 - Returns true as long as data is in the set.
- Access fields in the result via method
 - getString (index)
 - getInt (index)
 - getDouble (index)
- Use getString if you are unsure of type!
- Access current row with .getRow ()
- Check if you are done with .last ()

15

ResultSetMetaData

- This is the metadata for the ResultSet
- This has information about the data columns including
 - Names
 - Data type
 - Label
 - Scale
 - Precision

16
