

JDBC

- Siga as instruções para instalar o banco de dados H2 (descritas no site);
- no projeto **Prograd**, crie um pacote que será responsável pelas conexões com o BD:
br.edu.ufabc.prograd.jdbc
- JDBC (Java DataBase Connectivity): conjunto de interfaces bem definidas dentro do pacote `java.sql`;
- driver JDBC: classes concretas que fazem ponte entre API JDBC e o BD;
- no pacote **br.edu.ufabc.prograd.jdbc**, crie uma classe chamada **ConnectionFactory** com o código:

```
package br.edu.ufabc.prograd.jdbc;
```

```
public class ConnectionFactory {  
    public Connection getConnection(){  
        System.out.println("Conectando ao banco de dados");  
        try {  
            // a linha abaixo não é mais necessária a partir do JDBC 4 (Java 6)  
            //Class.forName("org.h2.Driver");  
            return  
DriverManager.getConnection("jdbc:h2:tcp://localhost/~progweb", "admin", "admin");  
        } catch (Exception e) {  
            throw new RuntimeException(e);  
        }  
    }  
}
```

- adicione os imports necessários (`java.sql.Connection` e `java.sql.DriverManager`) usando o atalho **ctrl+I**;
- adicione o driver do H2 (um arquivo JAR. Pegue no TIDIA) contendo a implementação JDBC do H2 ao *classpath* do projeto. O driver do H2 chama-se `h2-1.3.170.jar` e encontra-se dentro da pasta `h2` -> pasta `bin`. Siga os passos: clique direito sobre o projeto, opção **Build Path -> Configure Build Path -> Libraries -> Add External JARs**
- teste a conexão com o código (acrescentar tratamento de exceção (*try-catch*) usando **ctrl+I**):

```
// testando conexão  
Connection connection = new ConnectionFactory().getConnection();  
System.out.println("Conexão estabelecida");  
connection.close();
```

- estabeleça e teste a conexão;
- sempre tratar exceções e sempre lembrar de fechar conexões do BD;
- os drivers podem ser baixados no site do fabricante;
- importância de criar uma fábrica de conexões: um lugar só para mudar se necessário, possibilidade de criar pooling de conexões, seguimos padrões de projeto sobre encapsular construção de objetos complicados (livro da Go4);

- no projeto **Prograd**, crie também um pacote que será responsável por objetos de acessos aos dados (DAO - Data Access Object): **br.edu.ufabc.prograd.dao**
- no pacote **br.edu.ufabc.prograd.dao**, crie um Javabeam **AlunoDAO**;
- lembre-se de importar a classe `java.sql` (cuidado para não confundir com a classe `com.sql`);
- coloque a conexão com o banco de dados no construtor do **AlunoDAO**:

```
public class AlunoDAO {
    private Connection connection;

    public AlunoDAO() {
        this.connection = new ConnectionFactory().getConnection();
    }
}
```

- implemente a inserção de registros:

```
public void insere(Aluno aluno) {
    String sql = "insert into alunos (nome,email,endereco) values (?,?,?)";

    try { // prepared statement para inserção
        PreparedStatement stmt = connection.prepareStatement(sql);

        // seta valores
        stmt.setString(1, aluno.getNome());
        stmt.setString(2, aluno.getEmail());
        stmt.setString(3, aluno.getEndereco());

        //executa
        stmt.execute();

        // fecha statement
        stmt.close();
    } catch (SQLException e) {
        throw new RuntimeException(e);
    }
}
```

- modifique a classe **CriaAluno**, acrescentando um código para testar o método de inserção:

```
// gravando registro
AlunoDAO dao = new AlunoDAO();
dao.insere(aluno);
System.out.println("Gravado!");
```

- problemas com a SQL:

```
String sql = "insert into alunos (nome,email,endereco) values ('"+nome+"', '"+email+"'+'"+endereco+"')";
```

- difícil de ler;

- provável de ter erros;
- preconceito contra Joana D'Arc;
- SQL Injection: técnica usada para atacar aplicações orientadas a dados.
Exemplo: <http://xkcd.com/327/>
- sanitizar entradas do BD: tratar entradas (caracteres especiais, etc);
- mais genérico e mais seguro:

```
String sql = "insert into alunos (nome,email,endereco) values (?,?,?  
);";
```

- PreparedStatement: interface para executar as cláusulas, faz sanitização das entradas do BD;