

Java + JDBC: Connessione semplice a un Database

1. Cos'è JDBC?

JDBC (Java Database Connectivity) è una API Java per connettere le applicazioni Java ai database (es. MySQL, PostgreSQL, SQLite...).

Permette di:

- Collegarsi a un database
- Eseguire query SQL (SELECT, INSERT, UPDATE, DELETE)
- Gestire i risultati

2. Struttura base della connessione JDBC

Passi fondamentali:

1. Caricare il driver JDBC (non sempre necessario con JDBC 4.0+)
2. Aprire una connessione (Connection)
3. Creare uno Statement/PreparedStatement
4. Eseguire la query SQL
5. Processare i risultati (solo per SELECT)
6. Chiudere la connessione

3. Costanti di connessione

Meglio usare una classe dedicata (DBConfig.java):

```
public class DBConfig {  
  
    public static final String URL = "jdbc:mysql://localhost:3306/miodb";
```

```
public static final String USER = "root";

public static final String PASSWORD = "1234";

}
```

4. Classe per connettersi al database

```
import java.sql.*;

public class DBConnection {

    public static Connection connect() {

        try {

            return DriverManager.getConnection(DBConfig.URL, DBConfig.USER,
DBConfig.PASSWORD);

        } catch (SQLException e) {

            e.printStackTrace();

            return null;

        }

    }

}
```

5. Esempio: Inserire un nuovo record

```
public class InserisciDati {

    public static void main(String[] args) {

        Connection conn = DBConnection.connect();

        if (conn != null) {

            try {
```

```

String sql = "INSERT INTO utenti (nome, email) VALUES (?, ?)";

PreparedStatement stmt = conn.prepareStatement(sql);

stmt.setString(1, "Mario");

stmt.setString(2, "mario@email.com");

stmt.executeUpdate();

System.out.println("Dati inseriti con successo.");

conn.close();

} catch (SQLException e) {

    e.printStackTrace();

}

}

}

}

```

6. Esempio: Leggere i dati dal database

```

public class LeggiDati {

    public static void main(String[] args) {

        Connection conn = DBConnection.connect();

        if (conn != null) {

            try {

                String sql = "SELECT * FROM utenti";

                Statement stmt = conn.createStatement();

                ResultSet rs = stmt.executeQuery(sql);

                while (rs.next()) {

                    System.out.println("ID: " + rs.getInt("id"));

                    System.out.println("Nome: " + rs.getString("nome"));

```

```
        System.out.println("Email: " + rs.getString("email"));
    }
    conn.close();
} catch (SQLException e) {
    e.printStackTrace();
}
}
}
```

7. Note finali

- Devi avere il driver JDBC giusto (es. mysql-connector-java.jar nel classpath)
- La tabella deve esistere nel DB
- Puoi creare tabelle anche via codice, ma è meglio farlo con SQL inizialmente