



FPT POLYTECHNIC



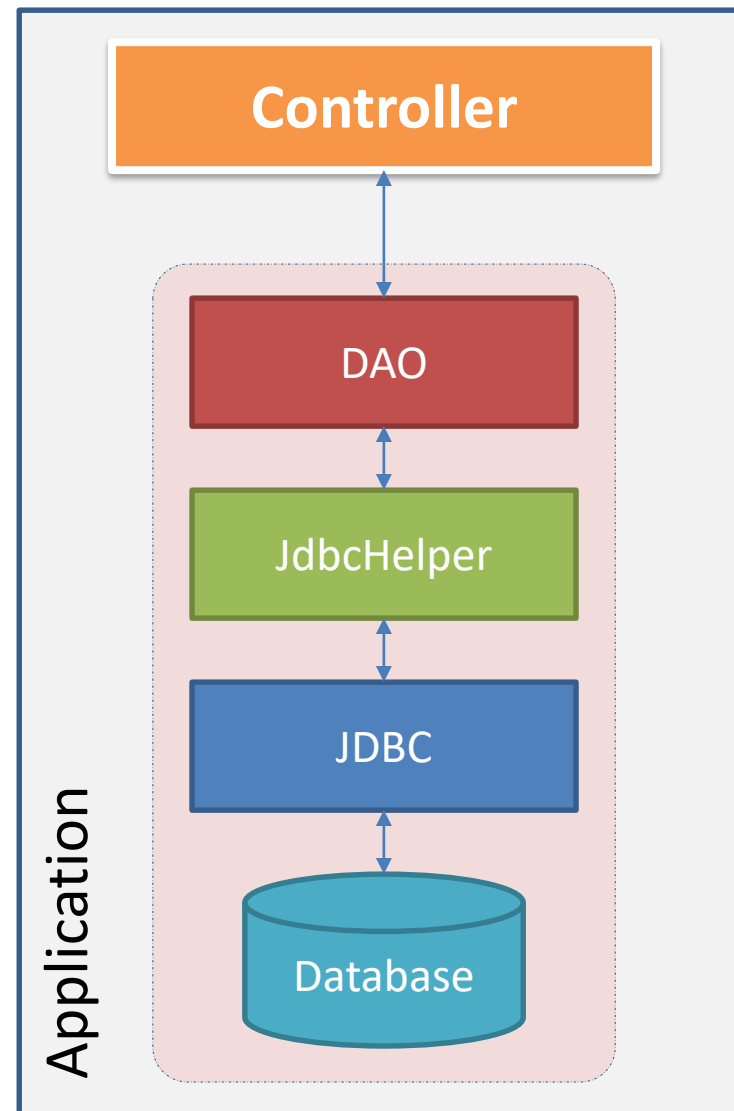
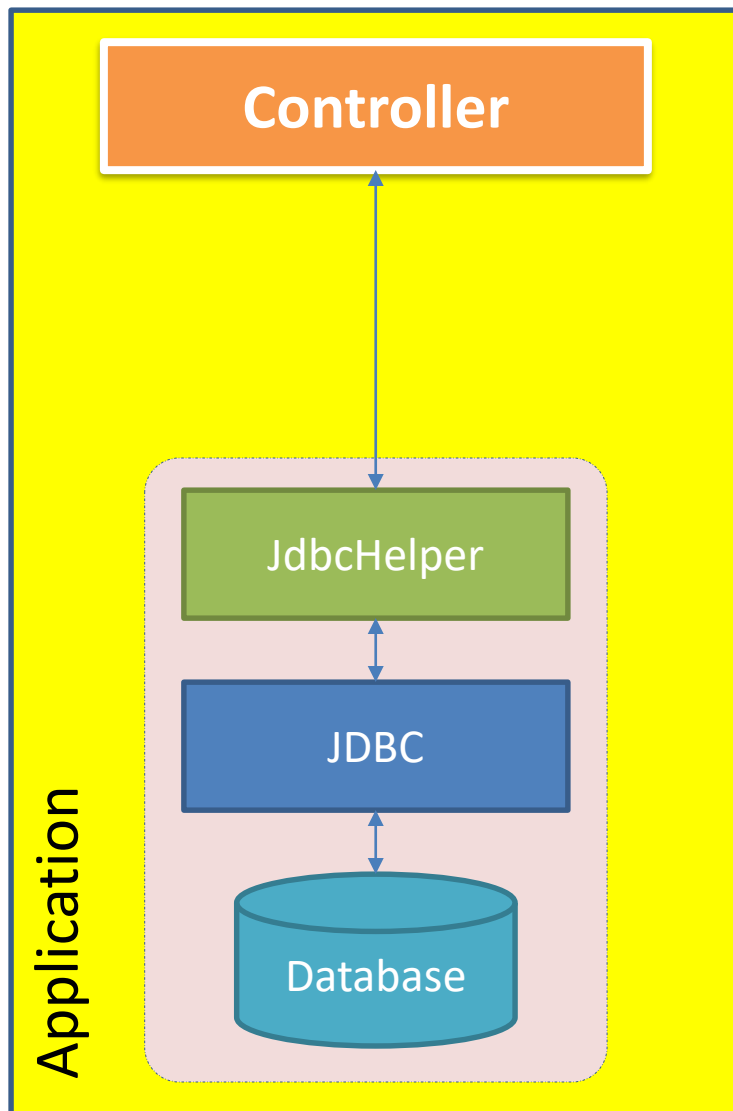
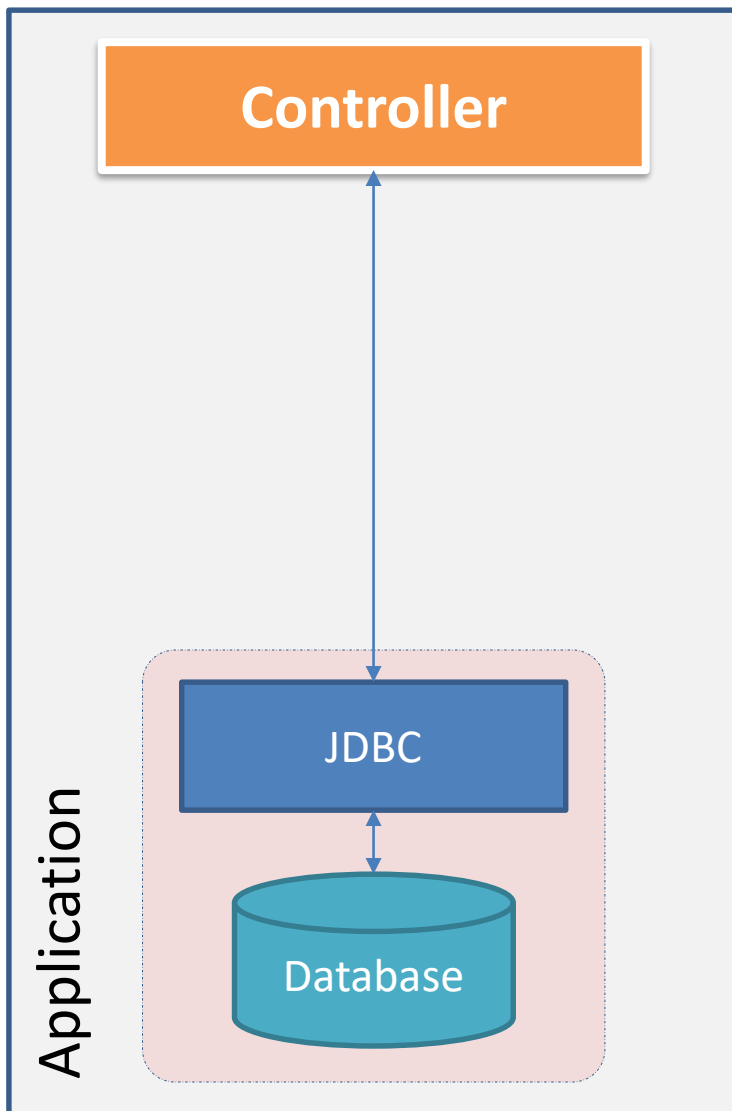
Conceive Design Implement Operate

LẬP TRÌNH CSDL

GIẢNG VIÊN: NGUYỄN NGHIỆM

www.poly.edu.vn

MỨC ĐỘ TÁI SỬ DỤNG TRONG LẬP TRÌNH JDBC





FPT POLYTECHNIC



Conceive Design Implement Operate

XÂY DỰNG JDBC HELPER

GIẢNG VIÊN: NGUYỄN NGHIỆM

www.poly.edu.vn

- ⊙ Kết thúc môn học này, bạn có khả năng
 - ⊙ Đơn giản hóa lập trình JDBC
 - ⊙ Tái sử dụng được mã nguồn



❑ Statement, PreparedStatement hay CallableStatement

- ❖ Thao tác

- ❖ Truy vấn

❑ Truy vấn

- ❖ ResultSet

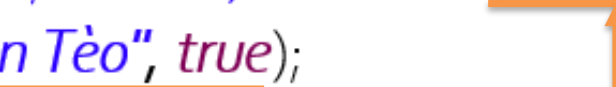
- ❖ Value

```
query(sql, args): ResultSet
```

```
update(sql, args): int
```

```
value(sql, args): Object
```

```
String sql = "INSERT INTO NhanVien (MaNV, MatKhai, HoTen, VaiTro) VALUES (?, ?, ?, ?)";  
int count = JdbcHelper.update(sql, "TeoNV", "123", "Nguyễn Văn Tèo", true);
```



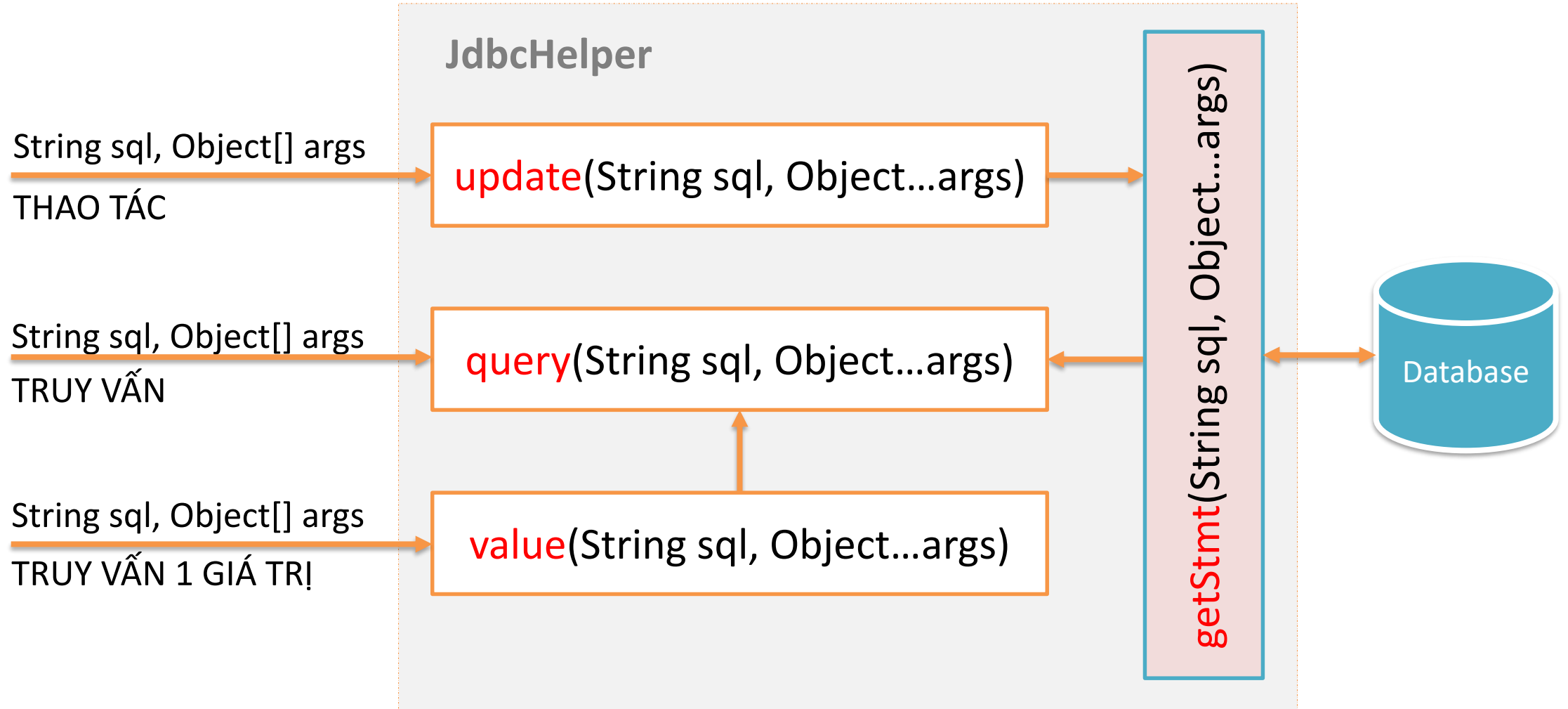
```
String sql = "SELECT * FROM ChuyenDe WHERE HocPhi BETWEEN ? AND ?";  
ResultSet rs = JdbcHelper.query(sql, 500.0, 700.0);
```



```
String sql = "{CALL sp_BangDiem (?)}";  
ResultSet rs = JdbcHelper.query(sql, 1000);
```

```
String sql = "SELECT max(MaKH) FROM KhoaHoc";  
int count = (Integer) JdbcHelper.value(sql);
```

- ❑ JdbcHelper là lớp tiện ích chứa các phương thức **update()**, **query()** và **value()** cho phép truy vấn và thao tác CSDL đơn giản.



CẤU TRÚC TỔ CHỨC CỦA JDBCHELPER

```
public class JdbcHelper {  
    static String driver = "com.microsoft.sqlserver.SQLServerDriver";  
    static String dburl = "jdbc:sqlserver://localhost;database=EduSys";  
    static String user = "sa";  
    static String pass = "123";  
    static {  
        try {  
            Class.forName(driver);  
        } catch (Exception e) {  
            throw new RuntimeException(e);  
        }  
    }  
  
    public static PreparedStatement getStmt(String sql, Object... args) throws SQLException {...}  
    public static int update(String sql, Object... args) throws SQLException {...}  
    public static ResultSet query(String sql, Object... args) throws SQLException {...}  
    public static Object value(String sql, Object... args) {...}  
}
```



```
public static PreparedStatement getStmt(String sql, Object... args) throws SQLException {  
    Connection conn = DriverManager.getConnection(dburl, user, pass);  
    PreparedStatement stmt;  
    if (sql.trim().startsWith("{")) {  
        stmt = conn.prepareCall(sql); // PROC  
    } else {  
        stmt = conn.prepareStatement(sql); // SQL  
    }  
    for (int i = 0; i < args.length; i++) {  
        stmt.setObject(i + 1, args[i]);  
    }  
    return stmt;  
}
```

INSERT NhanVien(MaNV, HonTen,
MatKhu, VaiTro) VALUES(?, ?, ?, ?)

SELECT * FROM NhanVien

SELECT * FROM ChuyenDe WHERE
HocPhi BETWEEN ? AND ?

{CALL sp_BangDiem (?)}

String sql = "SQL...?, ?, ?...";

PreparedStatement stmt = JdbcHelper.getStmt(sql, a1, a2, a3...);

```
public static ResultSet query(String sql, Object... args) throws SQLException {  
    PreparedStatement stmt = JdbcHelper.getStmt(sql, args);  
    return stmt.executeQuery();  
}
```

```
String sql = "SELECT * FROM NhanVien";
```

```
ResultSet rs = JdbcHelper.query(sql);
```

```
String sql = "SELECT * FROM NhanVien WHERE MaNV=?";
```

```
ResultSet rs = JdbcHelper.query(sql, "TeoNV");
```

```
String sql = "SELECT * FROM ChuyenDe WHERE HocPhi BETWEEN ? AND ?";
```

```
ResultSet rs = JdbcHelper.query(sql, 500.0, 900.0);
```

```
String sql = "{CALL sp_BangDiem (?)}";
```

```
ResultSet rs = JdbcHelper.query(sql, 1000);
```

```
public static Object value(String sql, Object... args) {  
    try {  
        ResultSet rs = JdbcHelper.query(sql, args);  
        if(rs.next()) {  
            return rs.getObject(o);  
        }  
        rs.getStatement().getConnection().close();  
        return null;  
    } catch (Exception e) {  
        throw new RuntimeException(e);  
    }  
}
```

```
String sql = "SELECT count(*) FROM NhanVien";  
int count = (Integer)JdbcHelper.value(sql);
```

```
String sql = "SELECT HoTen FROM NhanVien WHERE  
MaNV=?";  
String name = (String)JdbcHelper.value(sql,  
"TeoNV");
```

```
public static int update(String sql, Object... args) {  
    try {  
        PreparedStatement stmt = JdbcHelper.getStmt(sql, args);  
        try {  
            return stmt.executeUpdate();  
        }  
        finally {  
            stmt.getConnection().close();  
        }  
    } catch (Exception e) {  
        throw new RuntimeException(e);  
    }  
}
```

String sql = "INSERT...?, ?, ?...";

String sql = "UPDATE...?, ?, ?...";

String sql = "DELETE...?, ?, ?...";

JdbcHelper.update(sql, a1, a2, a3...);

✓ Xây dựng Jdbc Helper

✓ JdbcHelper.**query**(String sql, Object...args) => ResultSet

✓ JdbcHelper.**update**(String sql, Object...args) => int

✓ JdbcHelper.**value**(String sql, Object...args) => Object





Cảm ơn