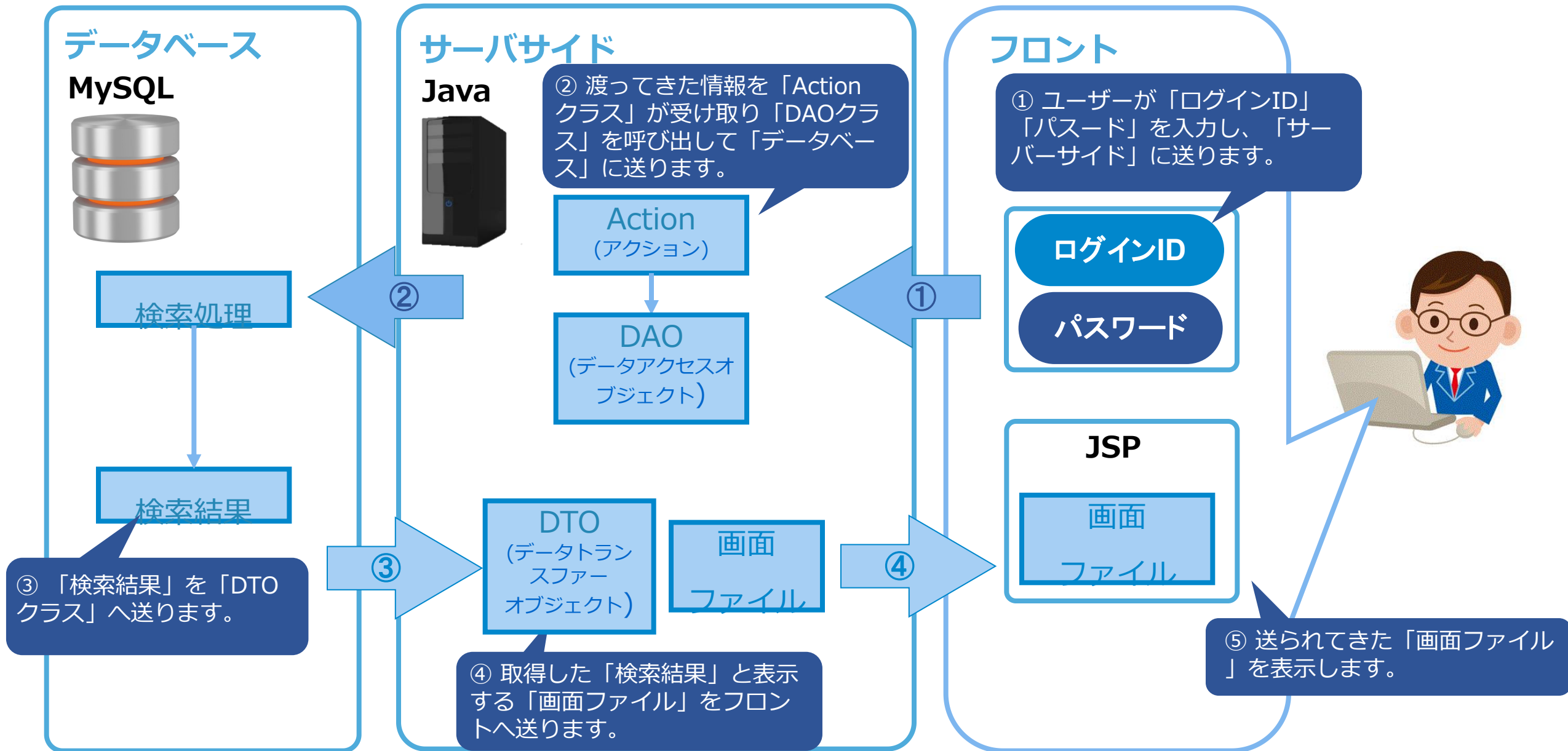

ログイン認証

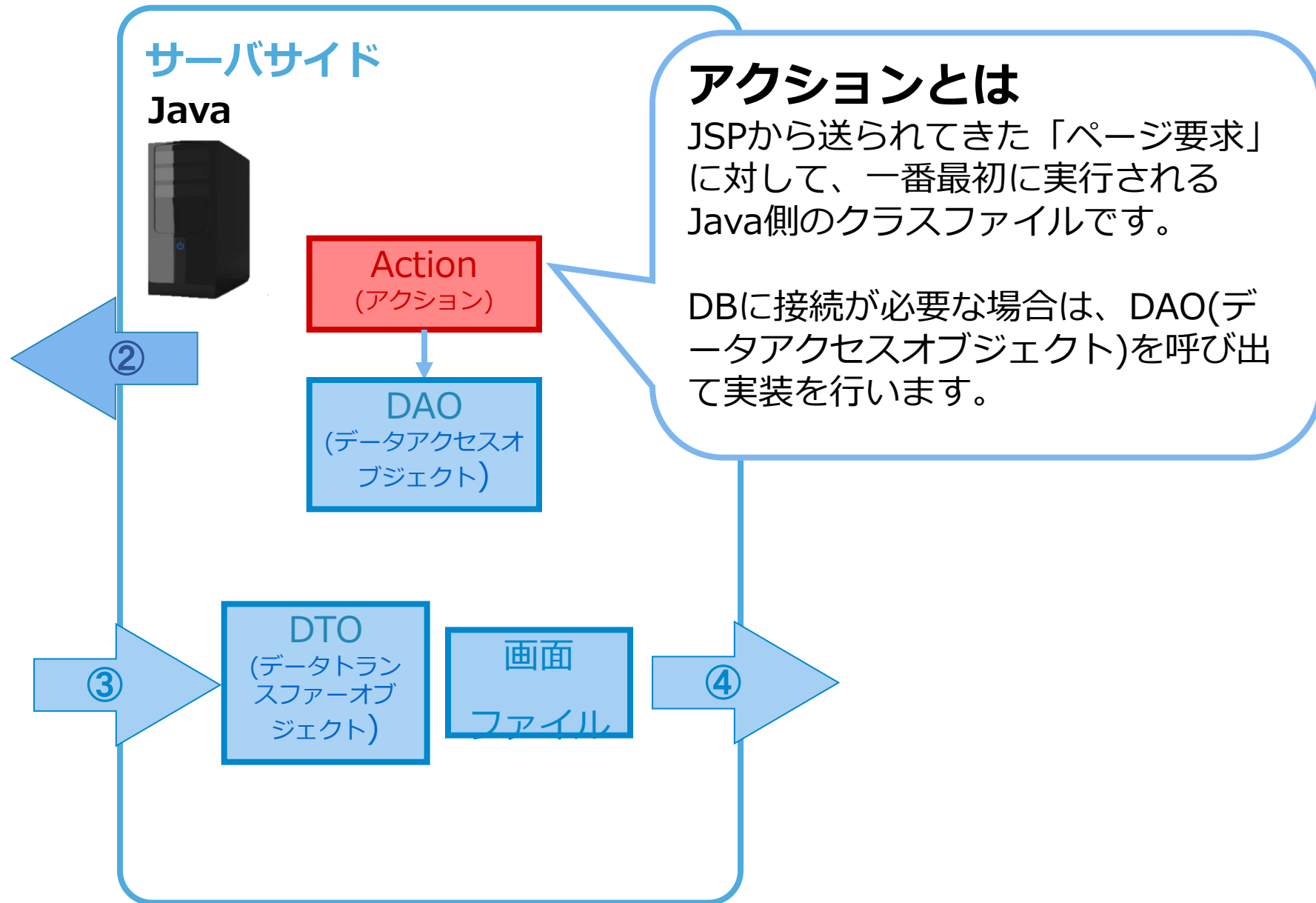
Action DAO DTOの作成

3 時間目

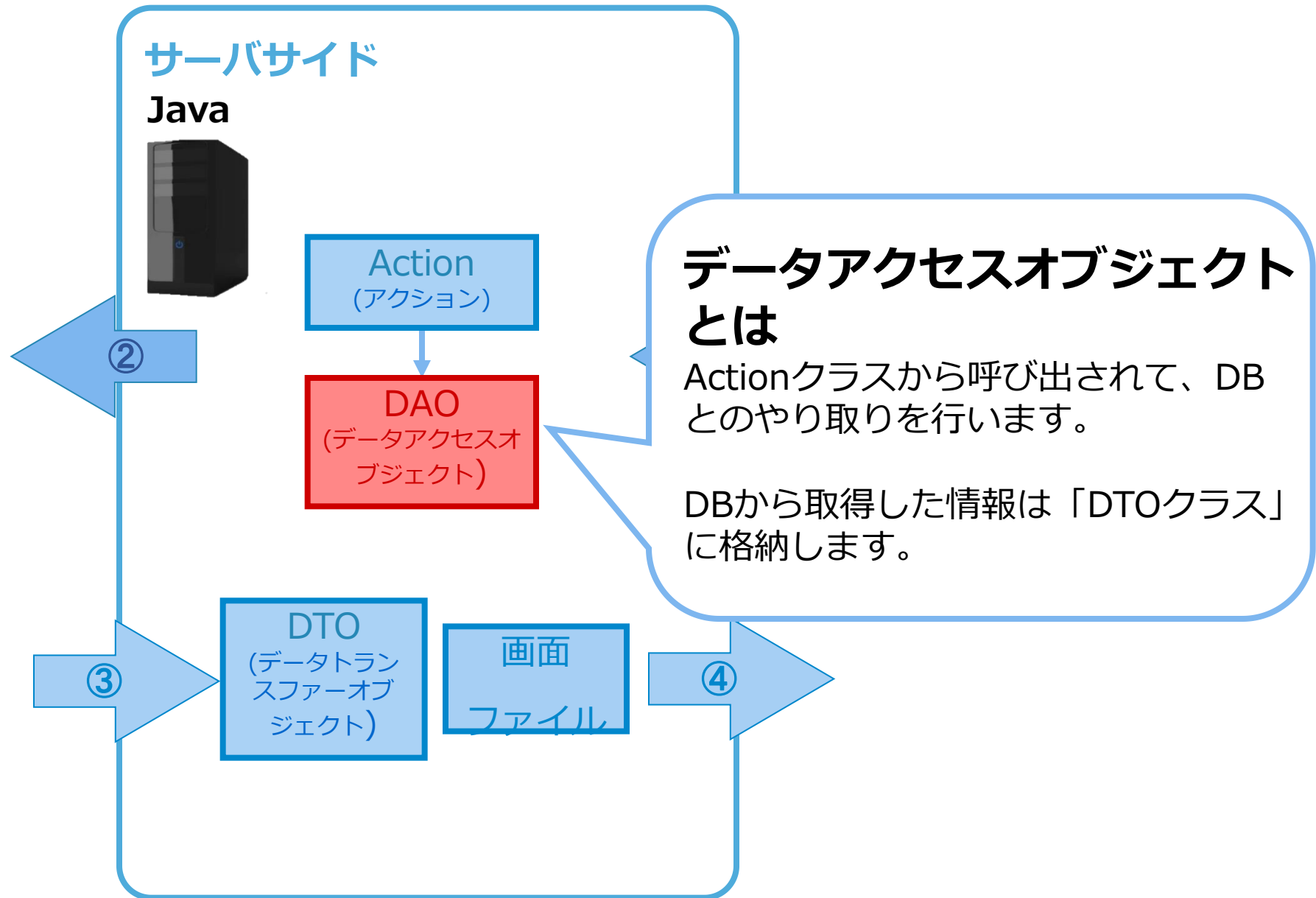
MVCモデルについて



MVCモデルについて



MVCモデルについて



MVCモデルについて

サーバサイド
Java

データトランスファーオブジェクトとは

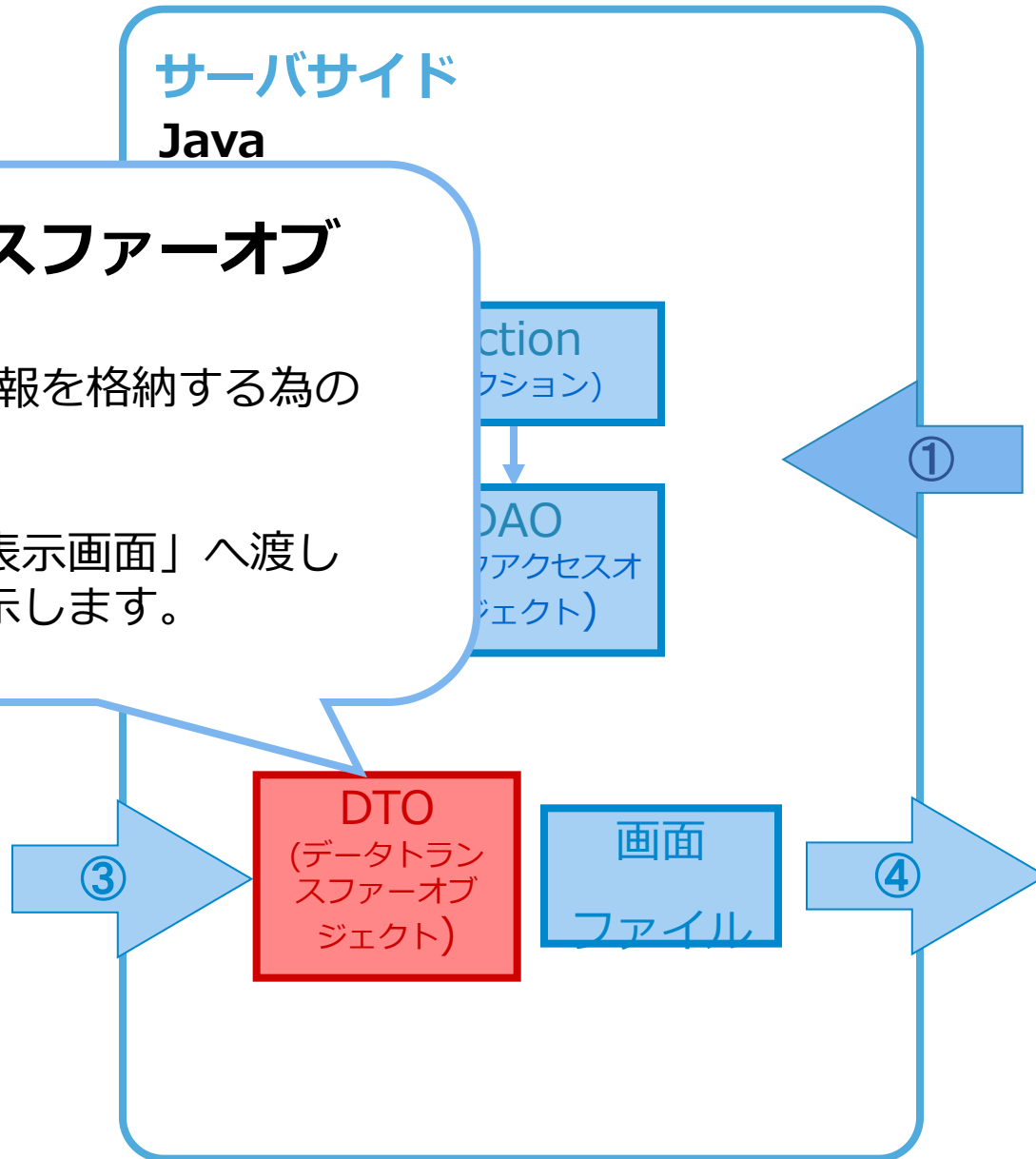
DBから取得した情報を格納する為のクラスです。

格納した情報を「表示画面」へ渡して、画面の中で表示します。

ction
クション)
↓
DAO
アクセスオ
ジェクト)

DTO
(データトランスファーオブジェクト)

画面
ファイル



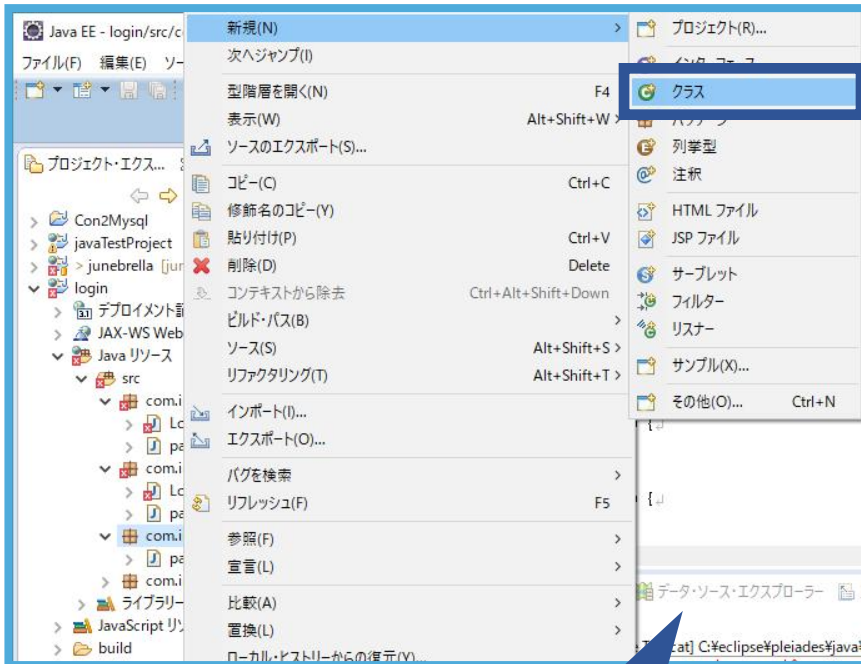
- 1) MVCモデルについて
- 2) クラスファイルの作成
 - 1 : Actionクラスの作成
 - 2 : DAOクラスの作成
 - 3 : DTOクラスの作成

Actionクラスの作成

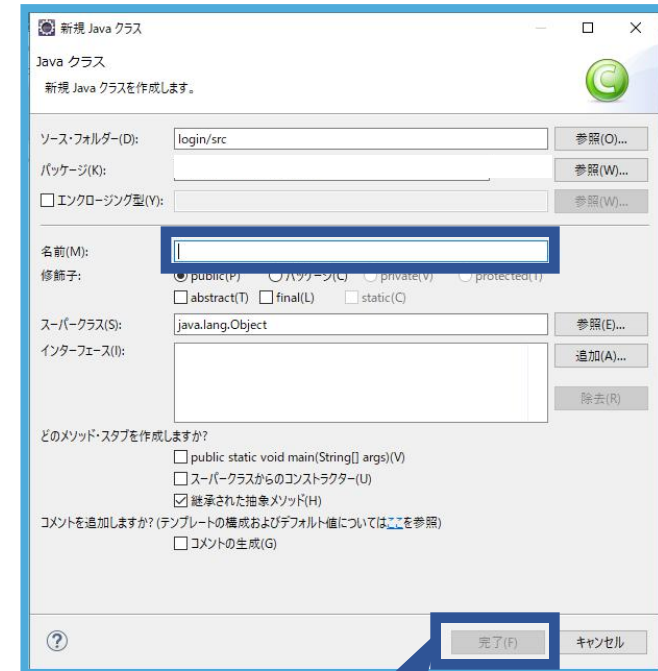
解説

Actionクラスでは、画面から送られてきたリクエストを取得します。
内部処理に応じてDAOやDTOクラスを呼び出し、最終的に次のJSPへ値を返します。

1 Actionクラス



① 「com.internousdev.login.action」を選択し、「右クリック」「新規」「クラス」を選択します。



② 「名前(M):」欄に「LoginAction」を入力し、完了ボタンをクリックします。

Actionクラスの作成

③ 以下の内容を写経します。

LoginAction(javaファイル)

```
package com.internousdev.action;

import java.sql.SQLException;
import java.util.Map;
import org.apache.struts2.interceptor.SessionAware;
import com.internousdev.dao.LoginDAO;
import com.internousdev.dto.LoginDTO;
import com.opensymphony.xwork2.ActionSupport;

public class LoginAction extends ActionSupport implements SessionAware {
    private String name;
    private String password;
    private Map<String, Object> session;
```

次へ続きます。

Actionクラスの作成

前の続きです。

```
public String execute() throws SQLException {  
    String ret = ERROR;  
    LoginDAO dao = new LoginDAO();  
    LoginDTO dto = new LoginDTO();  
  
    dto = dao.select(name,password);  
    if(name.equals(dto.getName())) {  
        if(password.equals(dto.getPassword())){  
            ret = SUCCESS ;  
        }  
    }  
    session.put("name", dto.getName());  
    return ret;  
}
```

メソッド名は「execute」にします。

「DAOクラス」を呼び出してDBに接続します。
今回はユーザーが入力した「ログインID」と「パスワード」に一致する情報がないのか検索します。

次へ続きます。

Actionクラスの作成

前の続きです。

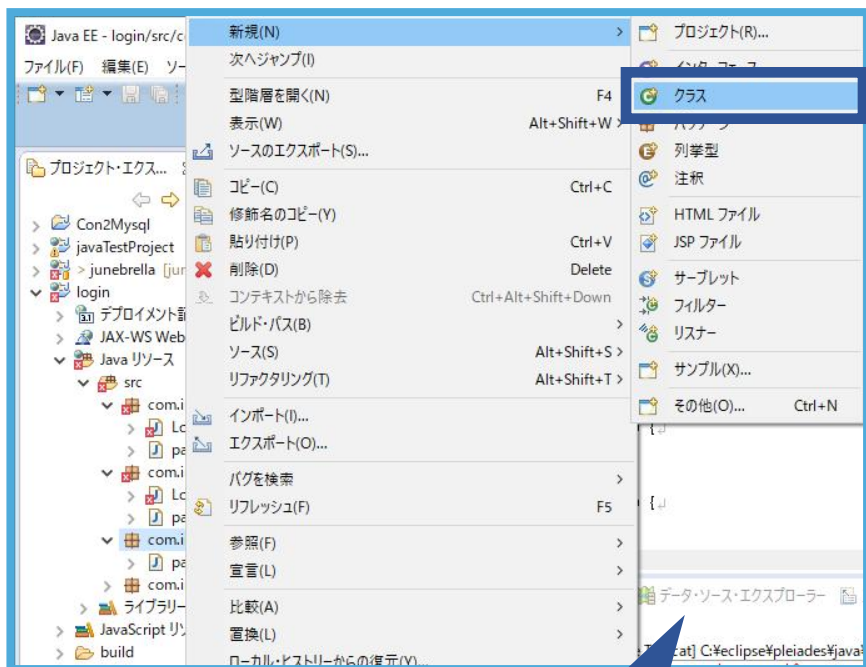
```
public String getName() {  
    return name;  
}  
public void setName(String name) {  
    this.name = name;  
}  
public String getPassword() {  
    return password;  
}  
public void setPassword(String password) {  
    this.password = password;  
}  
public Map<String, Object> getSession() {  
    return session;  
}  
public void setSession(Map<String, Object> session) {  
    this.session = session;  
}  
}
```

DAOクラスの作成

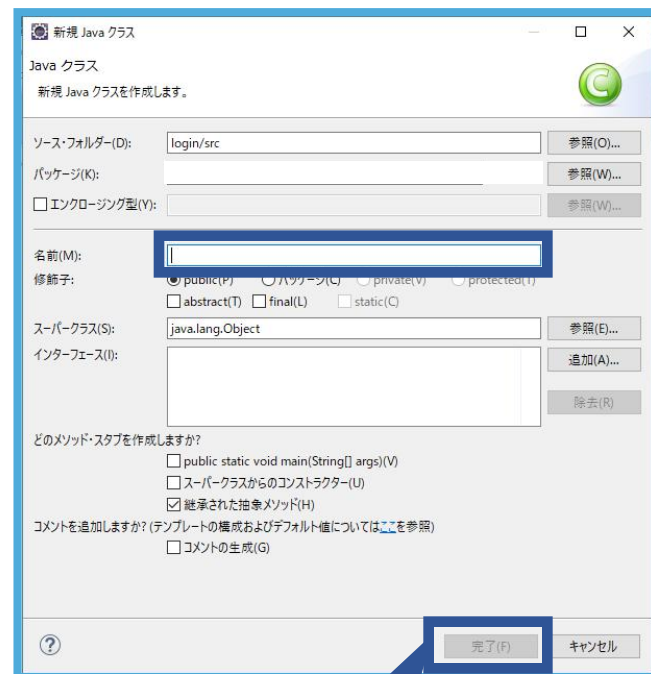
解説

DAOクラスでは、Actionから送られてきた情報を使ってDBへ問い合わせを行います。
問い合わせ取得した値情報をそのままDTOクラスに格納します。

2 DAOクラス



① 「com.internousdev.login.dao」を選択し、「右クリック」「新規」「クラス」を選択します。



② 「名前(M):」欄に「LoginDAO」を入力し、完了ボタンをクリックします。

DAOクラスの作成

LoginDAO(javaファイル)

③ 以下の内容を写経します。

```
package com.internousdev.dao;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import com.internousdev.dto.LoginDTO;
import com.internousdev.util.DBConnector;
```

メソッド名は処理内容を分かりやすくするために「select」にします。

```
public class LoginDAO {
    public LoginDTO select(String name,String password) throws SQLException{
        LoginDTO dto=new LoginDTO();
        DBConnector db = new DBConnector();
        Connection con = db.getConnection();
        String sql="select * from user where user_name=? and password=?";

        try {
            PreparedStatement ps = con.prepareStatement(sql);
            ps.setString(1, name);
            ps.setString(2, password);
```

次へ続きます。

DAOクラスの作成

前の続きです。

```
ResultSet rs=ps.executeQuery();  
  
if(rs.next()) {  
    dto.setName(rs.getString("user_name"));  
    dto.setPassword(rs.getString("password"));  
}  
  
} catch (SQLException e) {  
    e.printStackTrace();  
}  
  
} finally {  
    con.close();  
}  
  
return dto;  
  
}
```

SQL文を実行します。

Select文でDBから取得した情報を
DTOクラスに格納します。

処理中にSQL関連のエラーが発生した際
に実行する処理です。

必ず実行する処理です。
DB接続を切断します。

DAOクラスの実装

セキュリティ対策を考慮してJavaでは
PreparedStatementを利用します。

```
PreparedStatement ps = con.prepareStatement(sql);
```

```
ps.setString(1, name);  
ps.setString(2, password);
```

SQL文の「？」パラメータに指定した
値を挿入することができます。

```
String sql="select * from user where user_name=? and password=?";
```

↑
name

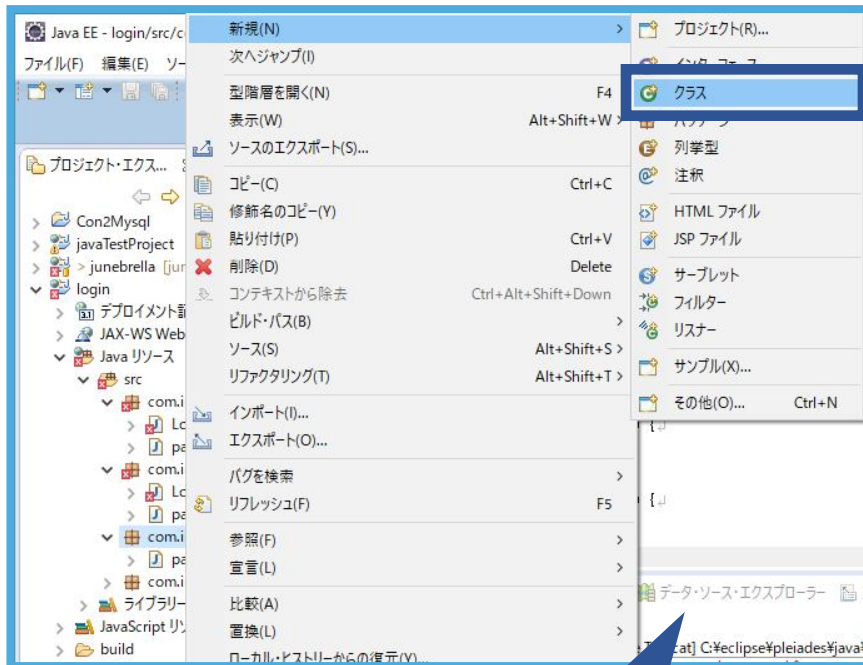
↑
password

DTOクラスの作成

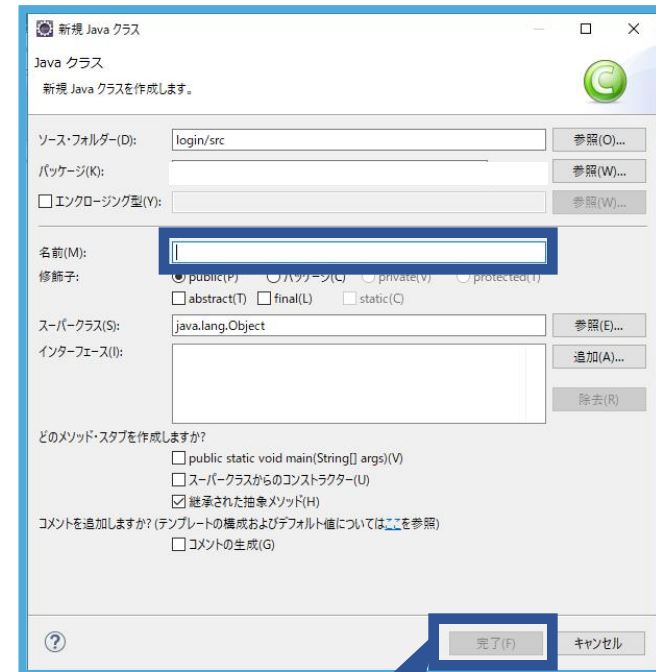
解説

DTOクラスでは、DAOがDBから取得した値Actionへ戻す時、DB値を格納するのに利用されます。
DTOクラス内では、テーブルカラム値に紐づいたフィールド変数とGetter Setterのみが定義されます

3 DTOクラス



① 「com.internousdev.login.dto」を選択し、「右クリック」「新規」「クラス」を選択します。



② 「名前(M):」欄に「LoginDTO」を入力し、完了ボタンをクリックします。

DTOクラスの作成

③ 以下の内容を写経します。

LoginDTO(javaファイル)

```
package com.internousdev.dto;  
public class LoginDTO {  
    private int id;  
    private String name;  
    private String password;
```

テーブルカラムに対応したフィールド変数を宣言します。

```
    public int getId() {  
        return id;  
    }
```

フィールド変数に対応したGetter Setterを定義します。

```
    public void setId(int id) {  
        this.id = id;  
    }
```

```
    public String getName() {  
        return name;  
    }
```

次へ続きます。

前の続きです。

```
public void setName(String name) {  
    this.name = name;  
}  
  
public String getPassword() {  
    return password;  
}  
  
public void setPassword(String password) {  
    this.password = password;  
}  
}
```