

## ① 機能

### ○ 認証機能

→ "ID" と "パスワード" を入力させる

→ 指定されたものに一致: 管理者

〃 不一致: 一般

	ID	パスワード
管理者	<small>kanrisyaId</small> kanrisya	<small>kanrisyaPass</small> 123
一般	<small>ippanId</small> ippan	<small>ippanPass</small> abc

※ 指定されたIDとパスワードは

プロパティファイルを使用

- キーと値がペアで格納されているファイル
- プログラムの設定値などを保存しておく
- コンパイルが不要である

○ 使う際は import が必要

→ `import java.util.Properties;`

○ `load X.Y.txt` で プロパティファイルを読み込み  
`getProperty X.Y.txt` で プロパティファイルの値を読み込む

○ 各行に「キー」と「値」をセットで記述

○ 行の始めに「#」の記述でコメント

○ 拡張子は「properties」である

## ⑥ クラス仕様

○ View (JSP 対応)

1) login.jsp → ログイン画面

2) main.jsp → トップページ画面

3) tippan.jsp → 一般画面

4) kanriya.jsp → 管理者画面

# 1) login-jsp

ログイン

ID  ①

パスワード  ②

③

① type="text" name="inputId"

② type="password" name="inputPw"

③ type="submit" value="ログイン"

## 2) main.jsp

### メインメニュー

①  
一般ページは こちら をクリック

②  
管理者ページは こちら をクリック

③  
ログアウト

① href="/ninsyo/kengen

② href="/ninsyo/kengen

③ href="/ninsyo/logout

### 3) ippan.jsp

一般

$x_1 > x_2 - 1$  ①

① href = "/ninsyo/main"

② href = "/ninsyo/logout"

## 4) kanr'sya-jsp

管理者

X1=X2-1 ①

① href = "/ninsyo/main"

② href = "/ninsy/logout"

## o model パッケージ

1) User.java

→ アカウント情報 (id, password) を持つクラス

2) LoginLogic.java

→ 入力されたアカウント情報と

プロパティファイルのアカウント情報を  
比較する



## o controller (ページ - ジ (Servlet))

### 1) Main.java

→ ログインに関するリクエストを処理

→ "LoginLogic.java" の結果が

$$\left\{ \begin{array}{l} \cdot \text{true} : \text{「main.jsp」へ} \\ \cdot \text{false} : \text{「login.jsp」に戻る} \end{array} \right.$$

### 2) Kengen.java

→ 社員権限に関するリクエストを処理

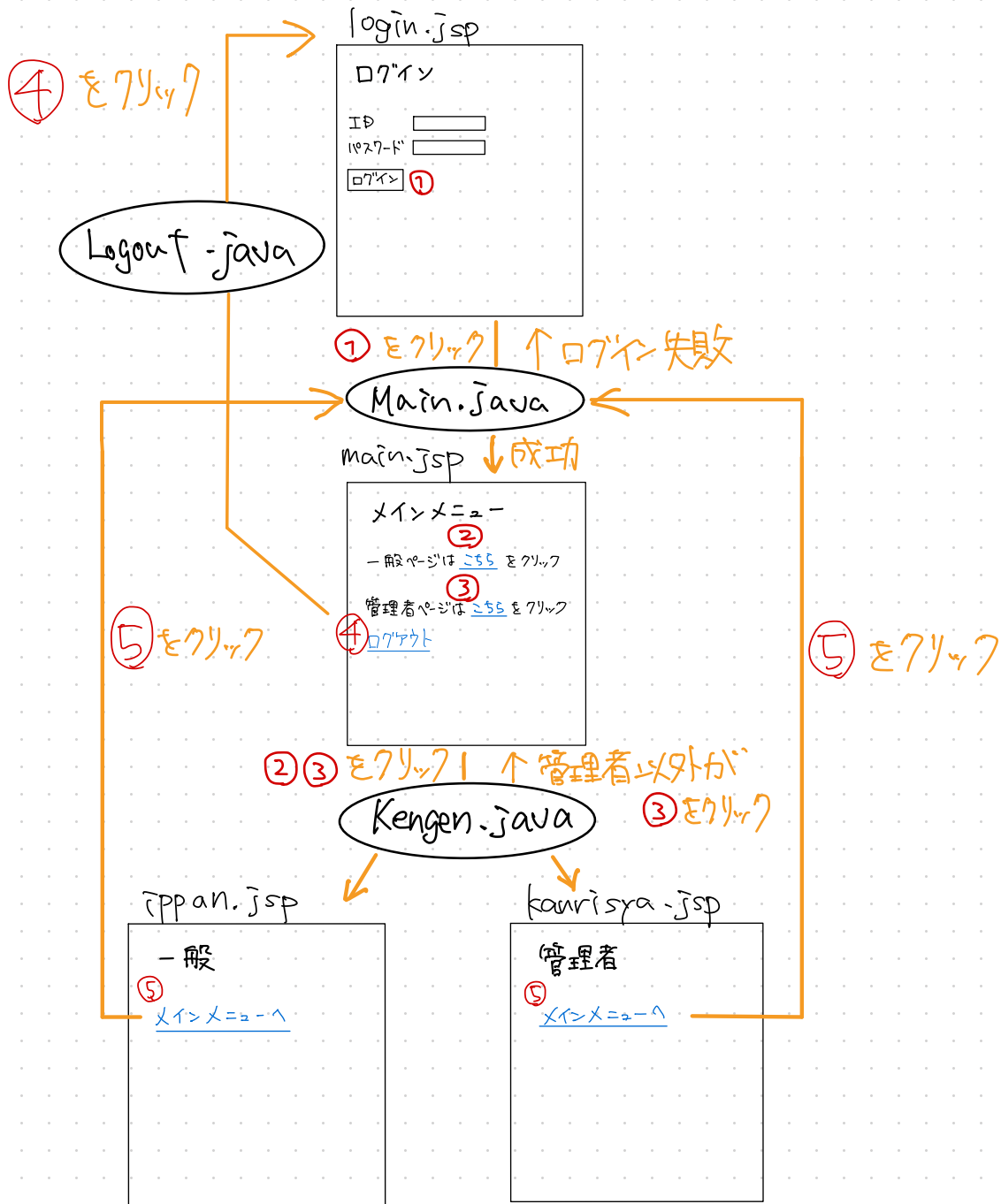
→  $\left\{ \begin{array}{l} \cdot \text{一般} : \text{「一般ページ」のみ} \\ \cdot \text{管理者} : \text{「一般」 + 「管理者」ページ} \end{array} \right.$

### 3) Logout.java

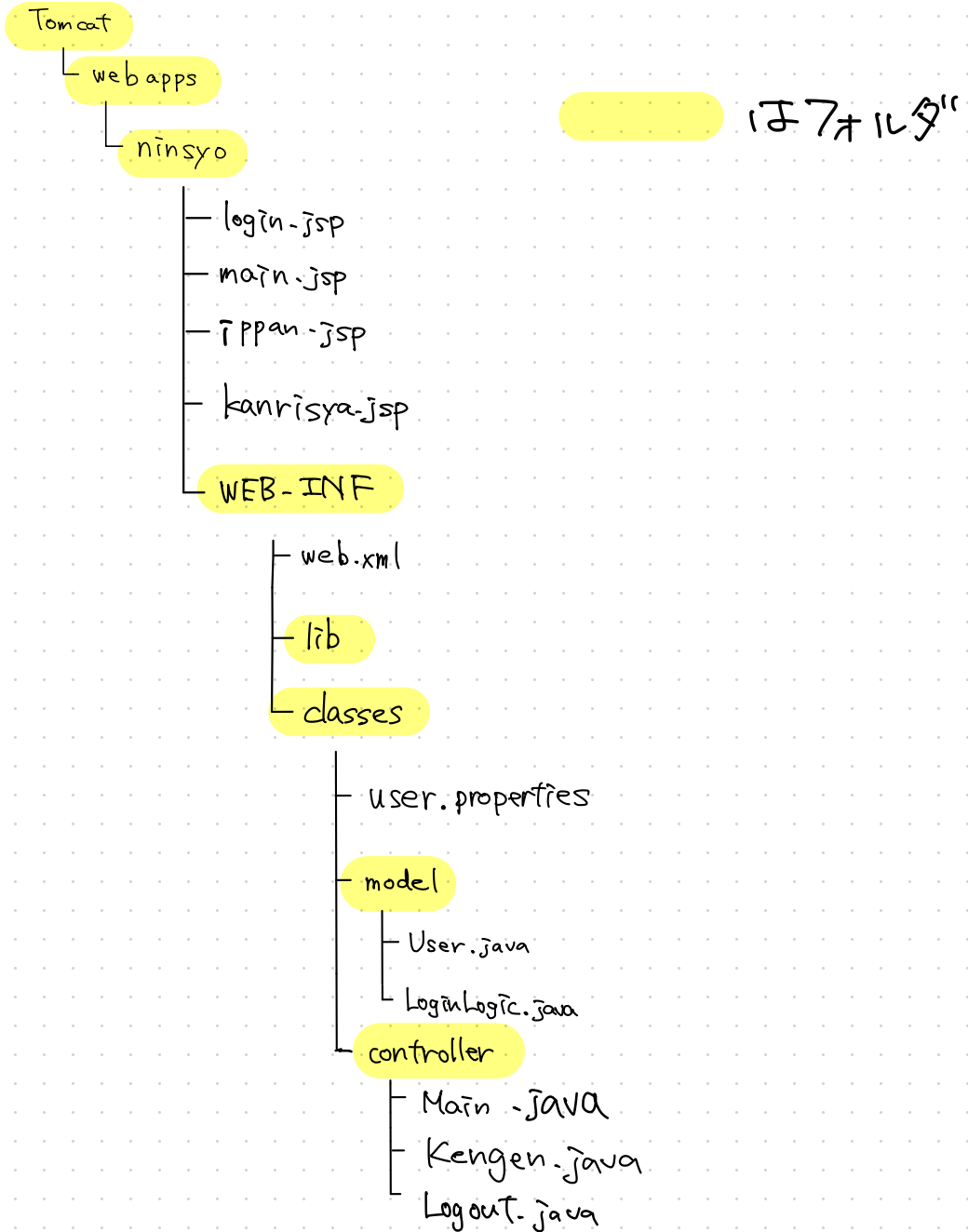
→ ログアウトに関するリクエストを処理

→ セッションを破棄

# ① 画面遷移図



# ② ディレクトリ構成



## ① 取り組みの流れ

- web.xml
  - URL の指定
- property ファイルの設定
  - ID, Pass をセット
- JSP ファイルの大枠作成
- model パッケージの作成
- controller パッケージの作成
- 画面遷移のチェック, 修正
- オートリロード機能追加
  - context.xml の  
<context>内には reloadable="true" を  
追加

○ Javadoc 作成

○ Filter に関する知識をインプットし.

コード作成するも完成せず

## ⑥ メモ

o Filter の メソッド 実行 タイミング

init() → フィルタがインスタンス化  
された直後

destroy() → フィルタのインスタンスが  
破棄される直前

★ doFilter() → 設定したサーブレットクラスを  
リクエストしたとき

## ① 疑問点

○ redirect と forward について

URL の指定の仕方は違う？

"ninsyo/login.jsp"

"/main.jsp"

○ session の受け取りについて

→ NullPointerException の対処