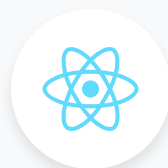


研修用アプリケーション概要

4つの研修プロジェクトの全体像



Jakarta EE



React



Struts



Python ETL

目次

1. 研修用アプリケーション概要

4つの研修プロジェクトの全体像と目的

2. 目次

全体構成と各セクションの概要（このページ）

3. berry-books（顧客向け）

Jakarta EE 10、JSF 4.0による書籍EC、レッスン1-5

4. berry-books-frontend（管理者向け）

React + TypeScriptによる管理画面SPA、レッスン6

5. struts-to-jsf-person（レガシー）

Struts 1.3.10からJSFへの移行、レッスン7

6. accounting_glue（ETL）

Python/PySparkによるデータ統合パイプライン、レッスン8

7. まとめ

プロジェクト比較と研修を通じての到達目標

研修プロジェクト概要



エンタープライズJava開発（JSF/JPA）



モダンフロントエンド
（React/TypeScript）



レガシーマイグレーション（Struts→JSF）



データ統合パイプライン（Python/ETL）

berry-books（顧客向けオンライン書店）

概要

Jakarta EE 10とJSF 4.0を使用したオンライン書店のWebアプリケーション（顧客向け画面）。書籍検索、ショッピングカート、注文処理などのEC機能を実装。管理者画面は別途実装、DBは共有。

採用技術

- フレームワーク: Jakarta EE 10（JSF 4.0, JPA 3.1, CDI 4.0）
- サーバー: Payara Server 6
- データベース: HSQLDB（berry-books-frontendと共有）
- テスト: JUnit 5 + Mockito

主な機能

- 書籍検索・閲覧（カテゴリ/キーワード検索）
- ショッピングカート（セッション管理）
- 注文処理（配送先入力、決済方法選択、在庫チェック）
- 注文履歴表示

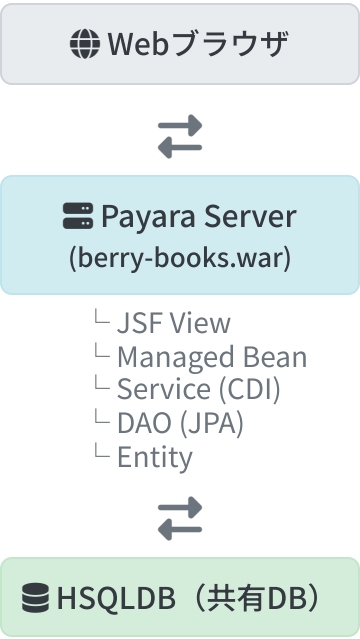
研修レッスン

- 1: 小規模改善（Enum化、メッセージリソース化）

2: アーキテクチャガイドライン準拠チェック
- 3: 不具合修正（バリデーション、ロック処理）

4: 機能拡張（在庫チェック機能追加）
- 5: 単体テスト生成（JUnit + Mockito）

システム構成



※管理者画面と同じDBを使用

berry-books-frontend（管理者向けオンライン書店）

概要

Berry Booksオンライン書店の**管理者画面**。React + TypeScriptによるSPAとして実装。顧客向け画面（berry-books）とは別のアプリケーションだが、**同じデータベースを共有**。バックエンドはREST API（berry-books-rest）で連携。

フロントエンド技術

フレームワーク: React 18 + TypeScript 5

ビルドツール: Vite 5

開発サーバー: HMR対応

バックエンド技術

フレームワーク: Jakarta EE 10（JAX-RS 3.1, JSON-B 3.0）

サーバー: Payara Server 6

データベース: HSQLDB（berry-booksと共有）

主な機能

顧客一覧表示（注文件数・購入冊数の統計情報含む）

顧客情報編集（インラインダイアログ、フォームバリデーション）

研修レッスン

レッスン6: 仕様書からのスクラッチ開発（React）

REST API設計・実装（JAX-RS）

SPA開発（React + TypeScript）

フロントエンド/バックエンド連携

システム構成

🌐 管理者（Webブラウザ）



⚡ Vite Dev Server
(React SPA)

REST API (JSON)



☰ Payara Server
(berry-books-rest.war)

└ JAX-RS Resource
└ Service (CDI)
└ DAO (JPA)



🗄️ HSQLDB（共有DB）

※berry-books（顧客向け）と同じDB

struts-to-jsf-person（レガシー移行）

概要

レガシーなStruts 1.3.10を使用した人材管理システム。フレームワークマイグレーション（Struts → JSF）の研修課題として使用。

採用技術

- フレームワーク: Apache Struts 1.3.10（レガシー）
- ビジネスロジック: EJB 3.2（Stateless Session Bean）
- データアクセス: JDBC（PreparedStatement）
- サーバー: Apache TomEE 8
- データベース: HSQLDB

主な機能


- Person一覧表示
- Person追加・編集（入力→確認→登録の3画面遷移）
- Person削除

研修レッスン


レッスン7: フレームワークのマイグレーション（Struts → JSF へのリライト）

- Strutsタグライブラリの理解
- レガシーコードの解析
- モダンフレームワークへの移行

システム構成

 Webブラウザ



 TomEE 8
(struts-person.war)

- └ JSP + Strutsタグ
- └ Action (Struts Controller)
- └ ActionForm
- └ EJB Service (@Stateless)
- └ DAO (JDBC)



 HSQLDB

マイグレーションポイント

- ⚠ Strutsの複雑な画面遷移をJSFのイベント駆動型モデルへ変換
- ⚠ ActionFormからBacking Beanへのデータモデル移行
- ⚠ JSPタグライブラリからFaceletへの変換

accounting_glue（ERP会計統合ETL）

概要

上流システム（売上・人事・在庫）から会計パッケージへのデータ統合を行うETL処理。ローカル（開発）とAWS Glue（本番）の両方で実行可能。

採用技術

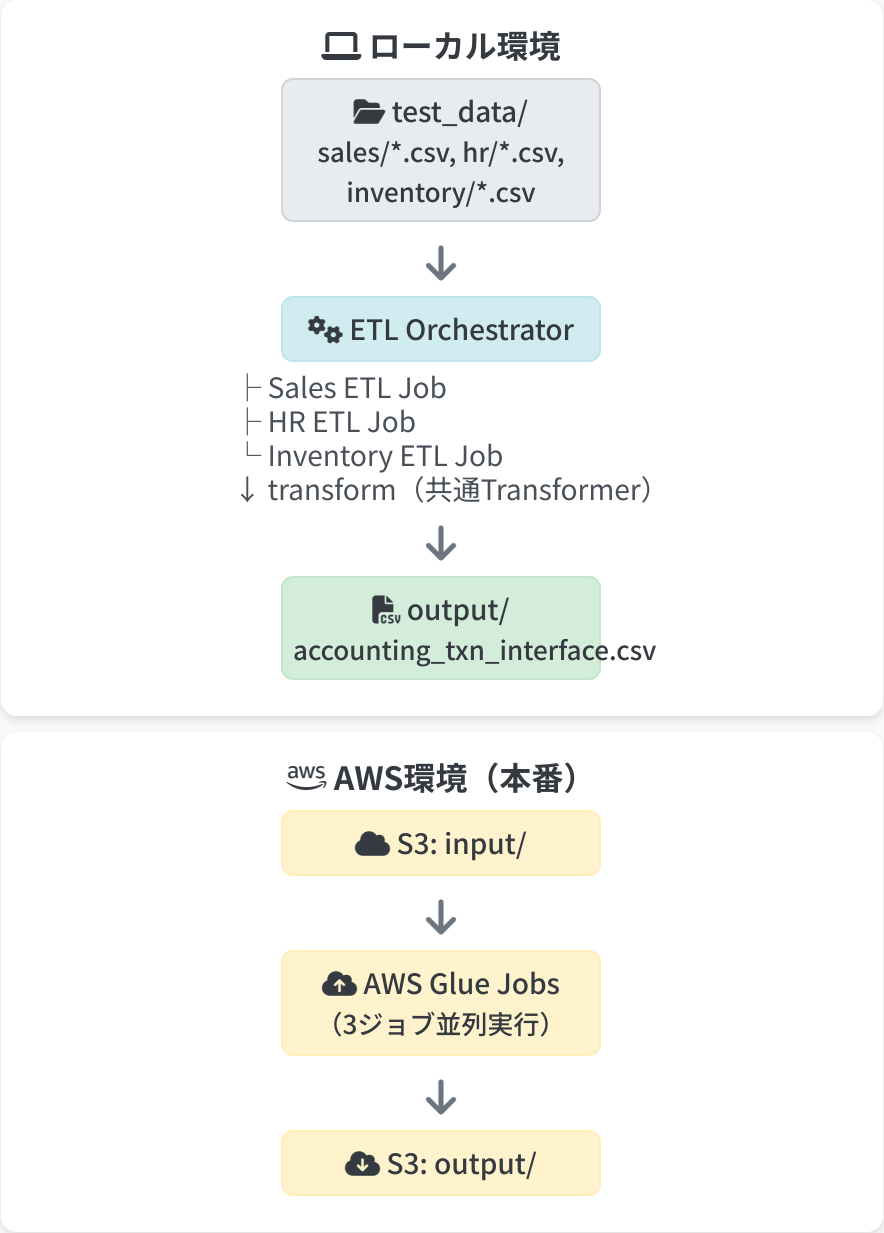
- 言語: Python 3.11
- 実行エンジン: PySpark（ローカル/AWS Glue）
- データソース: CSV（ファイルシステム/S3）
- テスト: pytest
- 並列処理: ThreadPoolExecutor / Spark

主な機能

- Sales ETL:** 売上データ → 売掛金/売上高の仕訳生成
- HR ETL:** 人事データ → 給料手当/未払費用の仕訳生成
- Inventory ETL:** 在庫データ → 商品/売上原価の仕訳生成
- 並列処理:** 3つのETLジョブを同時実行
- 統合出力:** 会計パッケージ取込用の単一CSVファイル生成

研修レッスン8: 仕様書からのスクラッチ開発

- ETL処理の設計・実装（データソース定義、変換ルール作成）
- データ変換ロジック（Transformer）の実装
- 単体テスト・統合テスト（pytest）
- クラウド対応（AWS Glue）と環境間の互換性確保



まとめ（比較と到達目標）

プロジェクト	技術スタック	画面種別	研修内容
berry-books	Jakarta EE 10 JSF 4.0 JPA 3.1 CDI 4.0 Payara Server 6 HSQLDB	顧客向けEC	レッスン1-5: 小規模改善、ガイドライン準拠、不具合修正、機能拡張、単体テスト
berry-books-frontend	React 18 TypeScript 5 Vite 5 JAX-RS 3.1 JSON-B 3.0	管理者SPA （DB共有）	レッスン6: 仕様書からのSPA開発（REST API設計・実装・連携）
struts-to-jsf-person	Struts 1.3.10 EJB 3.2 JDBC Apache TomEE 8 HSQLDB	レガシーシステム	レッスン7: フレームワークマイグレーション（Struts → JSF へのリライト）
accounting_glue	Python 3.11 PySpark AWS Glue CSV/S3 pytest	データ基盤 ETL	レッスン8: ETL開発（仕様書からのスクラッチ開発、テスト、クラウド対応）

共通ポイント

- 共有データベース活用（berry-books系）
- システム間連携（REST API, ETL）
- 自動テスト（JUnit, pytest）
- クラウド対応設計（AWS Glue）
- UI/UX設計（JSF, React）

研修到達目標

- 仕様書理解からコーディングまでの一貫したスキル習得
- レガシーコード解析と最新フレームワークへの移行手法
- 小規模改善から大規模リファクタリングまでの実践力
- フロントエンド/バックエンド/データ連携の総合的理解
- 品質担保のためのテスト実装・自動化スキル