はじめに読むもの

受注管理システムの歩き方

研修で3日間ずっと触るシステムです。何をする業務で、データがどう入っていて、どのファイルがどの役割なのか。ここを先に押さえておくと、演習で自分がどこを直しているのかを見失いません。Day1 の冒頭でも一緒に確認します。

どんな業務のシステムか

法人向けにIT機器を売っている会社の、受注を管理するシステムです。

営業が注文を受けたら受注として登録します。社内で内容を確認して確定させ、出荷して、届いたら完了。途中でお客様の都合により取り消しが入ることもあります。画面はありません。他のシステムから呼ばれる API だけの構成です。

扱うデータは3つです。顧客、受注、受注明細。受注1件に対して明細が複数ぶら下がります。たとえば「ネットワークスイッチ 10台」という受注の中身が、L3スイッチ6台とL2スイッチ4台に分かれている、という具合です。

テーブル	件数	持っているもの
customers	5	顧客コード、会社名、住所、電話、メール
orders	10	受注番号、顧客名、商品名、数量、単価、合計金額、ステータス、受注日、納品日
order_items	13	受注ID、商品コード、商品名、数量、単価、小計

金額は税込で持ちます。計算は 数量 × 単価 × 1.10 で、端数は切り捨てです。受注番号は `ORD-20260401-001` の形式で、日付と連番が入ります。

入っているデータ

受注10件は、ステータスも金額もばらけるように作ってあります。

ID	受注番号	顧客	商品	数量	合計（税込）	ステータス	受注日
1	ORD-20260401-001	東京電機工業	サーバーラック 42U	3	594,000	PENDING	04-01
2	ORD-20260401-002	大阪精密機械	ネットワークスイッチ 48ポート	10	935,000	CONFIRMED	04-01
3	ORD-20260402-003	名古屋システムサービス	UPS 3000VA	5	660,000	CONFIRMED	04-02
4	ORD-20260403-004	福岡ソリューションズ	デスクトップPC Core i7	20	3,190,000	SHIPPED	04-03
5	ORD-20260403-005	東京電機工業	モニター 27インチ 4K	20	990,000	PENDING	04-03
6	ORD-20260404-006	北海道テクノロジー	ノートPC 14インチ	15	2,772,000	DELIVERED	04-04
7	ORD-20260405-007	大阪精密機械	プリンター複合機 A3	2	770,000	PENDING	04-05
8	ORD-20260406-008	名古屋システムサービス	SSD 1TB NVMe	50	660,000	CONFIRMED	04-06
9	ORD-20260407-009	福岡ソリューションズ	キーボード メカニカル	30	280,500	CANCELLED	04-07
10	ORD-20260408-010	東京電機工業	Webカメラ フルHD	25	187,000	PENDING	04-08

受注日は 2026-04-01 から 2026-04-08 に分布しています。日付範囲で絞り込む演習では、この並びが効いてきます。ステータスは PENDING が4件、CONFIRMED が3件、SHIPPED・DELIVERED・CANCELLED が1件ずつです。

アプリはメモリ上のデータベース（H2）で動きます。止めるとデータは消え、起動し直すと必ずこの10件に戻ります。壊しても元に戻るので、気にせず触ってください。

ステータスの決まりごと

この5つの状態を行き来します。演習の合否はここを基準に判定します。

PENDING → **CONFIRMED** → **SHIPPED** → **DELIVERED**

PENDING または CONFIRMED → CANCELLED

SHIPPED 以降のキャンセルは業務上できません

受注は登録した時点で PENDING です。社内で内容を確認したら CONFIRMED、出荷したら SHIPPED、先方に届いたら DELIVERED。取り消しは、まだ出荷していない PENDING と CONFIRMED のときだけ認めます。出荷済みのものを取り消せてしまうと、モノは動いたのに帳簿だけ消える状態になるためです。

> 配布されたコードは、この決まりごとを完全には実装していません。 `CONFIRMED` から `CANCELLED` への変更が通らない、届け済みのものが取り消せてしまう、といった穴が残してあります。演習で埋めていきます。

どのファイルが何をしているか

4つの層に分かれています。上から下へ呼び出し、下は上を知りません。

controller 受け口	外から来たリクエストを受けて、下に渡して、結果を返します。業務の判断はここでは書きません。 controller/OrderController.java controller/CustomerController.java
service 判断 よく触る	ステータスを変えてよいか、金額をどう計算するか。業務の決まりごとはすべてここに集まります。演習でいちばん触る場所です。 service/OrderService.java（宣言だけ） service/OrderServiceImpl.java（中身）
repository 出し入れ	データベースから取ってくる、書き込む。メソッド名を決めるだけで問い合わせが組み立てられる仕組みを使っています。 repository/OrderRepository.java repository/CustomerRepository.java
model / dto 入れ物	model はテーブルの1行に対応する入れ物、dto は外へ返すための入れ物です。分けてあるのは、内部の持ち方をそのまま外に出さないためです。 model/Order.java・OrderItem.java・Customer.java dto/OrderDTO.java・OrderCreateRequest.java

ほかに、独自の例外を置いた `exception/` と、設定クラスを置いた `config/` があります。Java のファイルは全部で15本です。

> サービス層のインターフェース（ `OrderService.java` ）には、まだ中身のないメソッドが宣言だけ置いてあります。 `findByDateRange` がそれです。呼ぶと例外で止まります。これは仕様で、Day1 の演習で埋めます。

動かして中身を見る

起動して、ブラウザで受注一覧を開くところまでやってみてください。

起動する

VSCode で `handson` フォルダを開き、ターミナルで次を実行します。

```
.\mvnw.cmd spring-boot:run
```

`Started OrderApplication` と出れば起動しています。止めるときは `Ctrl+C` です。

受注一覧を見る

起動したまま、ブラウザのアドレス欄に次を入れてください。

```
http://localhost:8080/api/orders
```

受注10件ぶんのデータが、ひとかたまりの文字列で表示されます。これが API の返す生のデータです。

```
// こんな形で10件ぶんが続きます
[{"id":1,"orderNumber":"ORD-20260401-001","customerName":"東京電機工業株式会社",
"productName":"サーバーラック 42U","quantity":3,"unitPrice":180000,
"totalAmount":594000,"status":"PENDING","orderDate":"2026-04-01",
"deliveryDate":"2026-04-15"}, ...
```

読めなくはありませんが、10件でこれです。どのステータスが何件あるのか、合計はいくらか。ぱっと出てきません。

> この読みにくさは、Day1 の途中で解消します。受注を表で見られるダッシュボードを、Claude Code に作らせる演習があります。自分が見たい形の画面を、その場で用意できるようになります。

データベースを直接のぞく

SQL を打って中身を確かめたいときは、H2 コンソールが使えます。

```
http://localhost:8080/h2-console
```

JDBC URL に `jdbc:h2:mem:orderdb` を入れて接続します。ユーザー名は `sa`、パスワードは空です。SELECT * FROM ORDERS; と打てば、上の表と同じ10件が並びます。

3日間でどこを触るか

触る場所は日ごとにはっきり分かれています。

日	触るところ	やること
Day1	service / dto / static	足りない検索を実装し、明細を返せるようにし、ステータスの決まりごとを埋めます。途中でダッシュボードを1枚つくります
Day2	src/test/ と exercises/day2/	テストを書かせて足りない観点を補い、仕込まれたバグをテストで失敗させてから直します
Day3	exercises/day3/ と config/	異常終了と性能劣化の原因を突き止め、ヘルスチェックと処理時間ログを足し、最後にレビューします

配布したコードには、演習で使うための穴がいくつも残してあります。動かないのは不具合ではなく、そういう題材です。どこが穴なのかは演習の中で明かしていきます。