前端React 程式的程式功能說明：

1. **useMakeRequest.js**:

這是一個自定義的 React 鉤子 (hook)，用來向指定的 API 端點發送異步請求，並管理相關的狀態（如數據和錯誤）。此鉤子使用 useState 和 useEffect 來處理數據的狀態管理和異步請求的發起。

1. **Store.js**:

這是一個組件，用來顯示產品的總覽。它使用 useMakeRequest 鉤子來加載產品數據，並根據加載的結果顯示產品列表或顯示「無數據」信息。

1. **Card.js**:

這個功能用於顯示產品的視圖。它使用 Bootstrap 的 Card 組件來顯示每個產品的圖片、標題和價格。

1. **ComponentSession.js**:

這個功能使用 React 的 useState 和 useEffect 來加載和顯示數據，並且這些數據會根據用戶的操作動態更新。

1. **PagingCard.js**:

此功能實現了產品分頁。它允許用戶在頁面之間導航以查看不同的產品資訊。

1. **CheckoutForm.js**:
   * 這是一個用於處理結帳流程的功能。它顯示了用戶欲購買的產品列表，並允許用戶導航至不同的產品頁面。
2. **Login.js**:
   * 登入表單組件，它允許用戶輸入用戶名和密碼，並處理登入邏輯，包括向伺服器發送登入請求。
3. **ShowProduct.js**:
   * 此功能用於展示單項產品的詳細信息。當產品數據從伺服器加載完成後，用戶可以查看產品詳情並有選項購買產品。
4. **Cart.js**:
   * 購物車組件，顯示用戶已加入購物車的產品列表。用戶可以更改產品數量並處理結帳。
5. **LoginProductMain.js**:
   * 這是應用的菜單功能，它設置了路由和導航，管理登入狀態，並連接所有其他組件，如登入頁面、產品商店、產品詳情、購物車和結帳表單。

這些功能構成了一個完整的購物應用的前端，實作了用戶界面和功能的模組化。

Java應用程式結構：

1. **LoginProductCart0331Application.java**:
   * 在 Spring Boot 應用程式中，這個類別通常包含 main 方法，用於啟動應用程式。
2. **LoginUser.java**:
   * 這是一個模型類別（Model），代表應用程式中的「用戶」實體。包含用戶的屬性，如用戶名、密碼和其他個人資訊。
3. **LoginUserDAO.java**:
   * DAO（Data Access Object）是一個用於封裝資料庫訪問的類別，特別是與 LoginUser 相關的操作，如查詢和更新資料庫中的用戶資料。
4. **ProductDAO.java**:
   * 這是另一個 DAO 類別，用於管理與產品相關的資料庫操作。其包括獲取產品列表、查找特定產品等。
5. **ProductService.java**:
   * 這是一個服務類別，它封裝了與產品相關的業務邏輯。它可能會使用 ProductDAO 來執行其數據存取需求。
6. **ProductWithImage.java**:
   * 這個模型類別擴展了產品的信息，可能包括與產品相關的圖像數據。這是為了處理產品的圖像和其他多媒體內容。
7. **RatingWithImage.java**:
   * 這個類別可能用於表示包含圖像的評價信息。它包括用戶的評分、評論和關聯的圖片。
8. **LoginController.java**:
   * 這是一個控制器類別，用於處理有關用戶登入的 HTTP 請求。它包含方法來處理登入表單的提交和用戶身份驗證。
9. **ProductController.java**:
   * 另一個控制器類別，用於處理產品相關的 HTTP 請求，如顯示產品列表、產品詳情等。
10. **application.properties**:
    * 這個文件包含應用程式的配置設定，如資料庫連接信息、應用端口設定等。

**JDK 和 Spring Boot 版本**

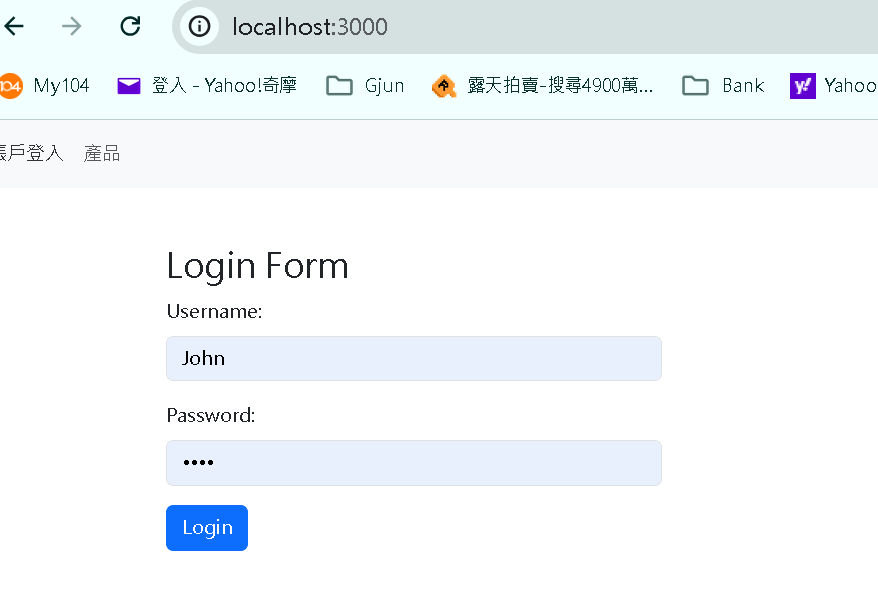
1. **JDK 版本**：
   * 應用程式使用的 Java 版本是 **Java 17**。
2. **Spring Boot 版本**：
   * 應用程式使用的 Spring Boot 版本是 **3.2.4**。

**資料庫連接**

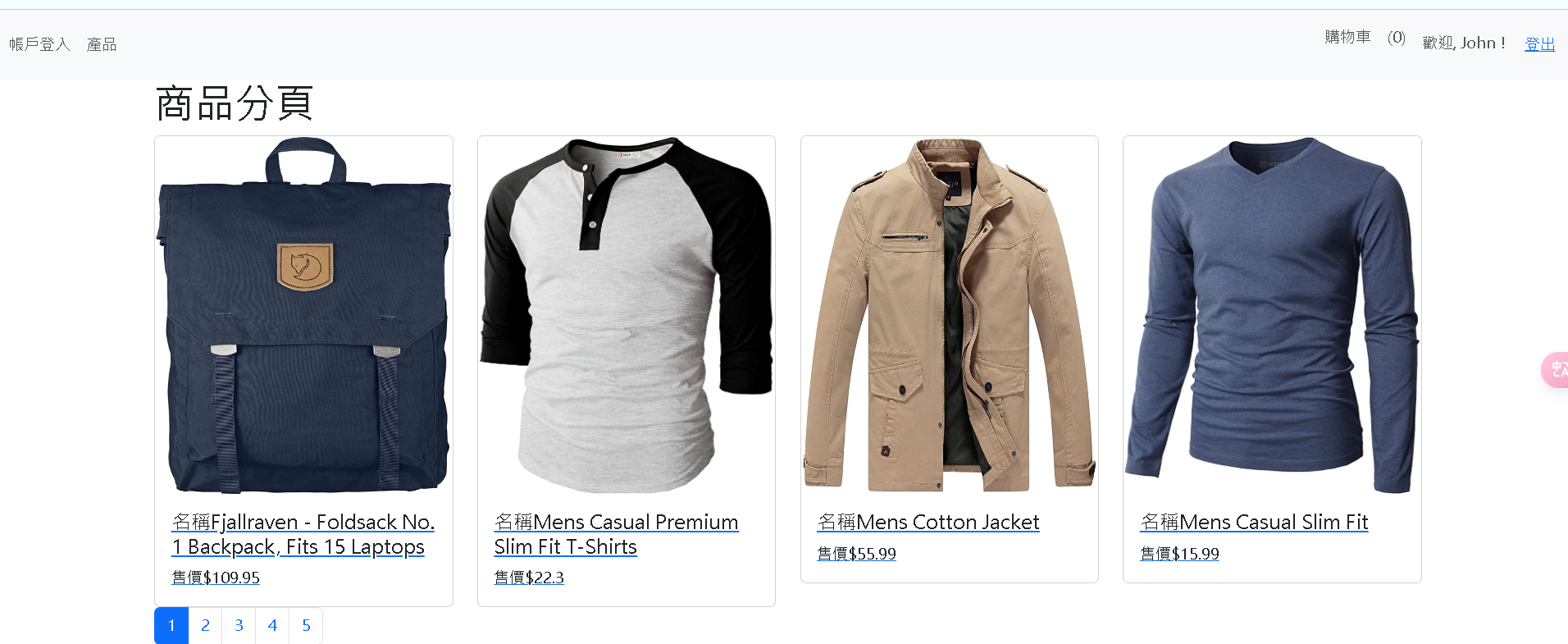
* **資料庫類型**：
  + 應用程式連接到的資料庫是 **MySQL**。
* **資料庫連接驅動**：
  + 使用的是 mysql-connector-java，這是 MySQL 的官方 JDBC 驅動程式。

操作說明:

### 1. 登入

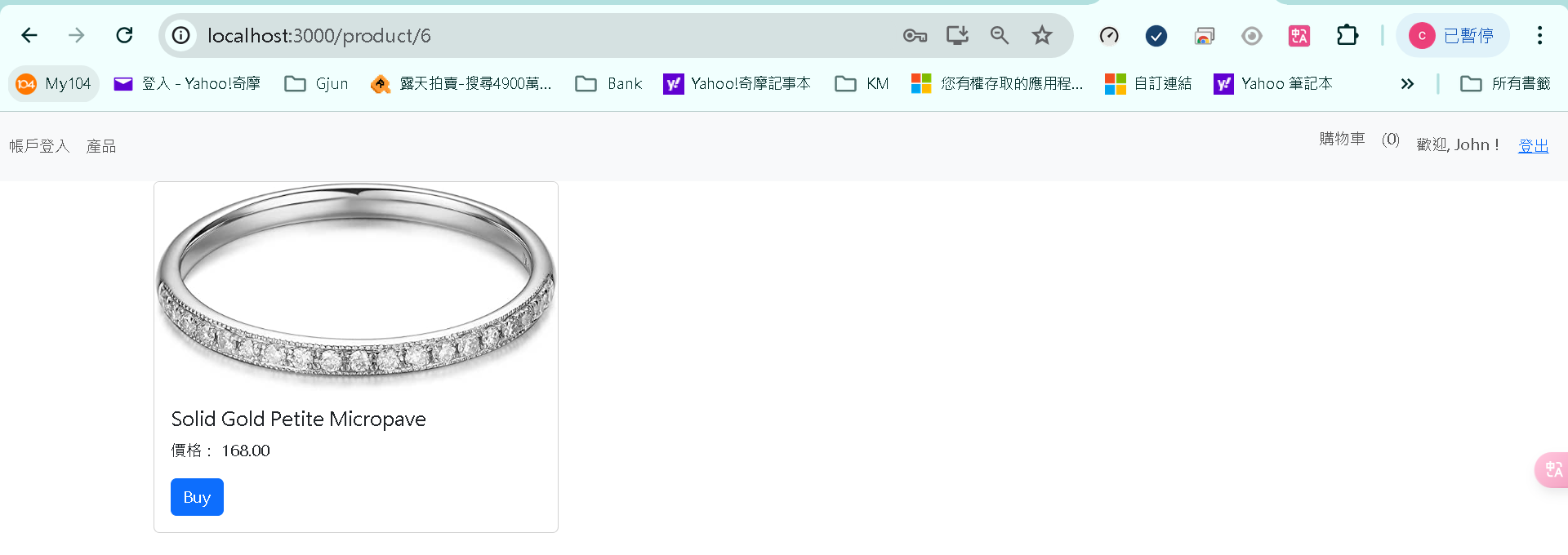
* **進入登入頁面**：在瀏覽器中輸入網址或通過導航到達登入頁面。
* **填寫用戶名和密碼**：
  + 在「Username」欄位輸入 John。
  + 在「Password」欄位輸入 j456。
* **提交登入信息**：點擊「Login」按鈕進行登入。
* 登入成功後，你會被導向到產品清單頁面。
* 

### 2. 瀏覽產品清單

* **查看產品**：登入後的產品清單頁面會展示多個產品。
* **選擇產品**：你可以瀏覽不同的產品，每個產品通常會顯示產品圖片、名稱、價格和一個「Buy」按鈕。
* 

### 3. 購買產品

* **選擇購買**：當你找到想要購買的產品後，點擊該產品的「Buy」按鈕。

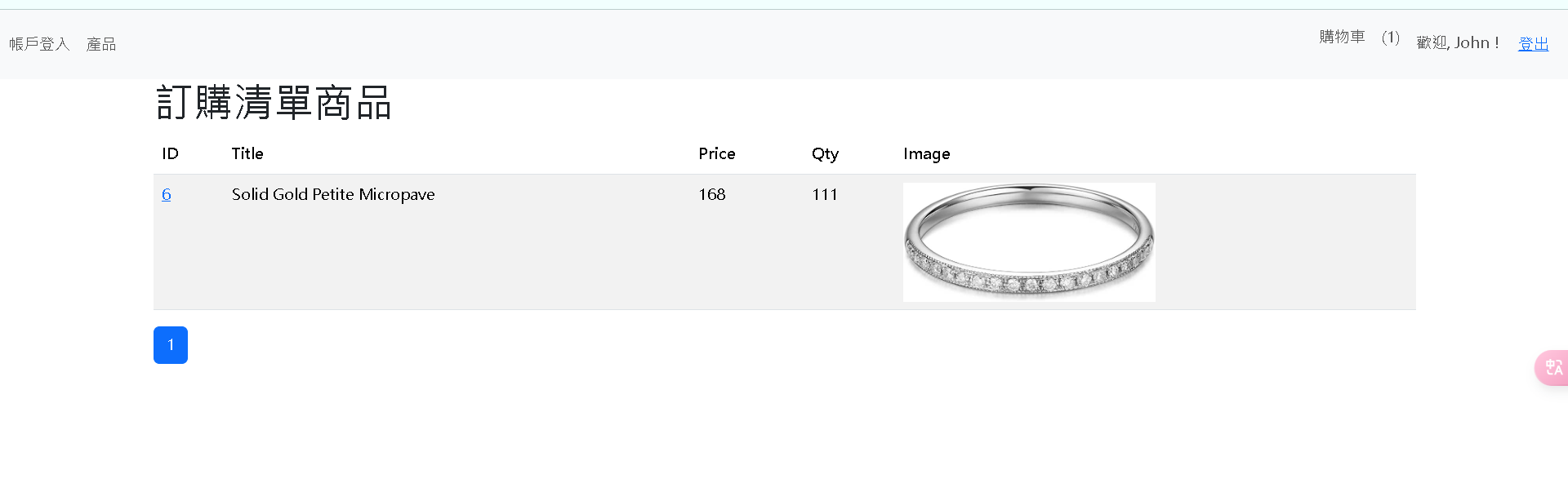


* **確認購買**：系統可能會要求你確認購買數量和詳細信息。
* **添加到購物車**：確認無誤後，產品將被添加到你的購物車。

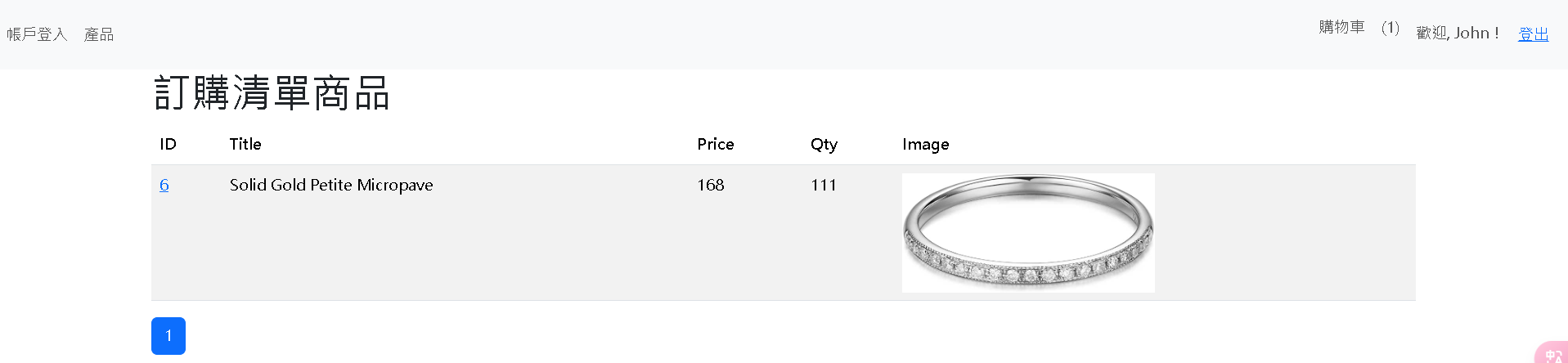
### 4. 查看和管理購物車

* **訪問購物車**：通過頁面上的購物車圖標或連結訪問你的購物車。
* **調整數量**：在購物車頁面，你可以修改產品的購買數量或刪除不想購買的產品。

### 5. 結帳

* **進行結帳**：檢查購物車後，點擊「Checkout」按鈕開始結帳流程。
* **提交訂單**：確認所有信息後，提交訂單完成購買。
* 

### 6. 完成購買

* 完成以上步驟後，你將收到一個訂購商品清單確認，此範例目前包括訂單Title、Price、Qty及購買品項照片等信息。
* 

以上程式說明與操作流程為基本的線上購物操作，是用來驗證前端React與後端Sprint BootＡＰＩ串接與呈現,而實際功能可以依照客戶需求再進行新增。