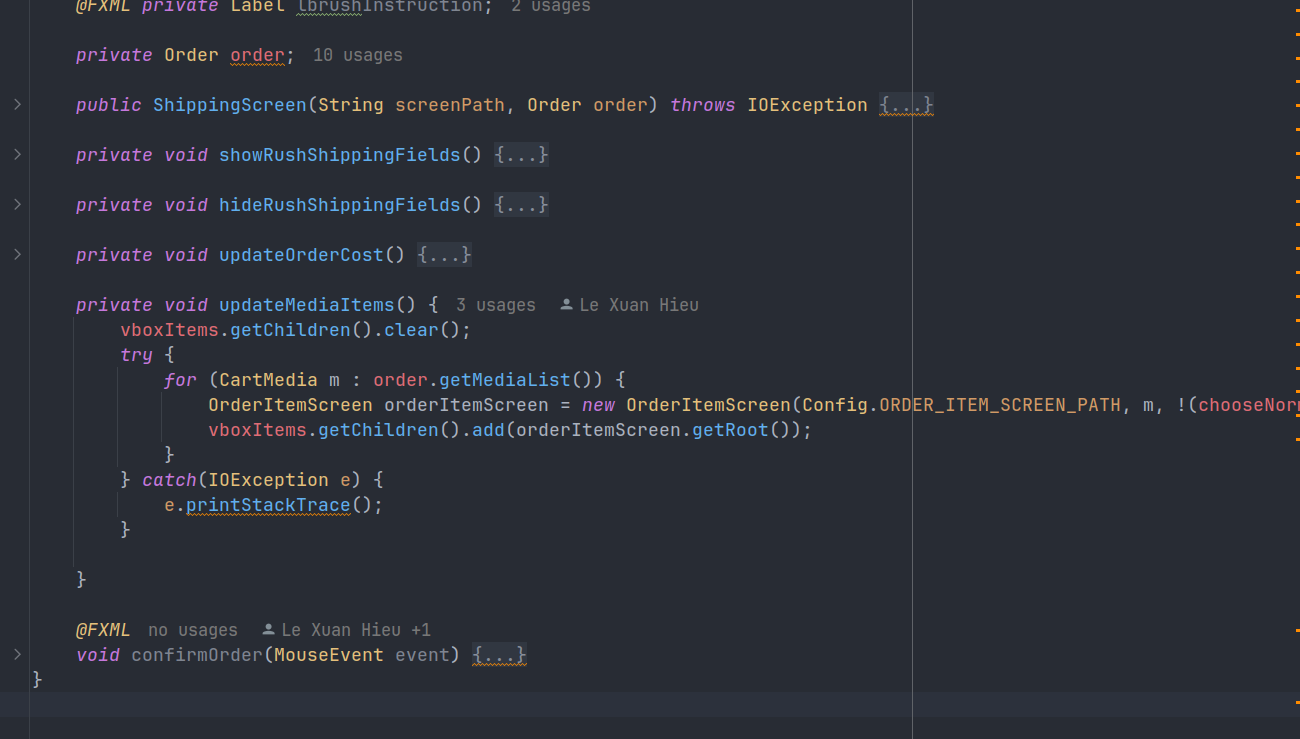
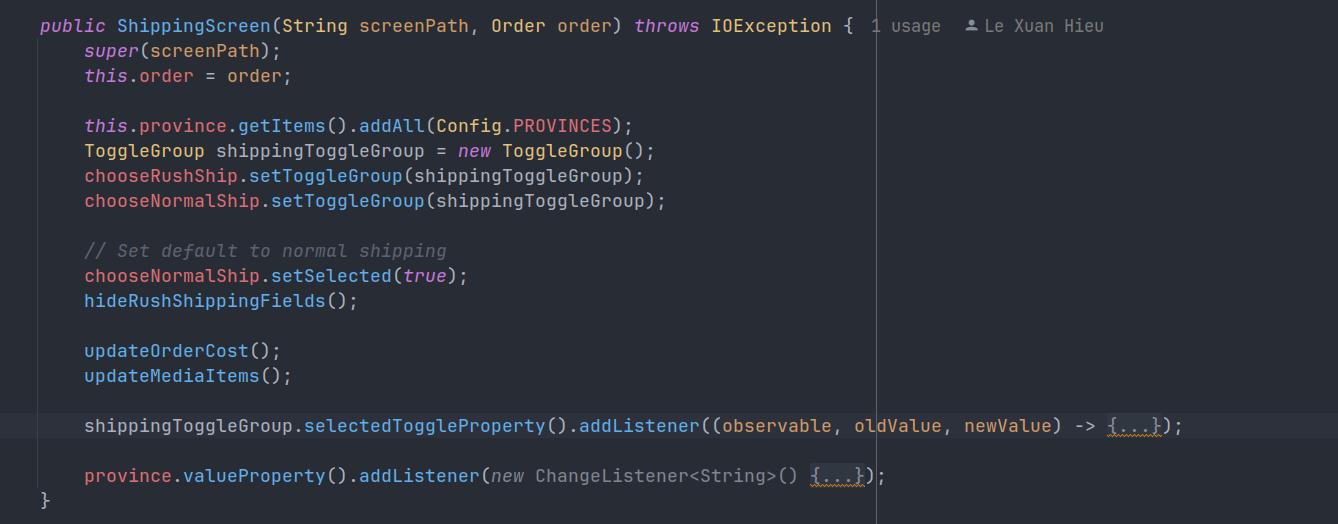
**Cohesion in Shipping and Invoice**

1. ShippingScreen





- methods ***updateOrderCost*** and ***updateMediaItems*** handle specific tasks related to order cost and media items, respectively.

- method ***confirmOrder*** handles multiple tasks (validation, order confirmation, invoice request).

=> **Procedure Cohesion,** methods are arranged in a logical order, following the shipping process flow.

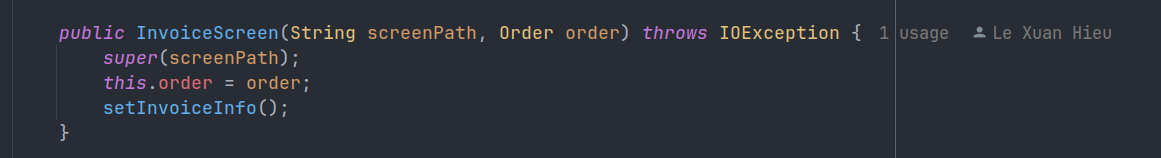
2. OrderItemSreen



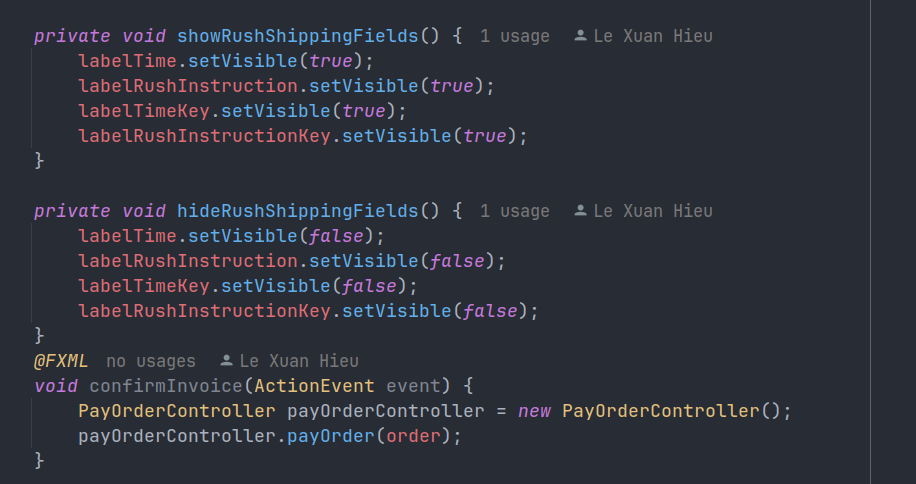
- This clear procedural flow avoids unrelated tasks within the class and focuses on the specific steps needed to display a single order item. => **Procedure Cohesion**

Clear names enhance code readability and maintainability. The suggested improvement in naming would further strengthen communication. => **Communicational Cohesion**

3. InvoiceScreen







While methods perform distinct tasks, they all contribute to the overall goal of displaying and managing the invoice. => **Procedure Cohesion.**

The class maintains a high level of informational cohesion by focusing solely on invoice data. There's no unrelated information processing or manipulation within the code. => **Information Cohesion**

4. InvoiceItemScreen



This clear procedural flow avoids unrelated tasks within the class and focuses on the specific steps needed to display a single invoice item.

=> **Procedure Cohesion**

The class avoids unrelated information processing, keeping the focus on the invoice item data. => **Information Cohesion**

Clear names enhance code readability and maintainability. The suggested improvement in naming would further strengthen communication.

=> **Communicational Cohesion**