SPOT Expt. No. 4c. Types of Inheritance and Different Accessibility of Packages

An E-Commerce Platform has multiple modules for products, users, and payments. The Product class is in the `ecommerce.product` package with attributes: `productId` (private), `productName` (public), `price` (protected), `stock` (default). The Order class is in the `ecommerce.order` package and extends `Product`. The Cart class is in the `ecommerce.cart` package but does not extend `Product`. The SpecialOrder class is in `ecommerce.special` package and extends `Order`.

Apply hierarchical inheritance where `Order` and `SpecialOrder` inherit from `Product`. Create an interface `Discountable with methods like `applyDiscount()`. Let `Order` and `SpecialOrder` implement it. → Demonstrates multiple inheritance via interface. Introduce a hierarchical structure where both `Order` and `Cart` extend `Product`. Build a multilevel inheritance chain: `SpecialOrder` extends `Order`, which extends `Product`. Apply hybrid inheritance by having a `FlashSaleProduct` class that extends `Product` and implements two interfaces `Discountable` and `Sharable`.

Students must analyze **accessibility of attributes** (`productId`, `productName`, `price`, `stock`) in the following contexts: Inside `Product` itself, Inside `Order` (different package subclass), Inside `Cart` (different package non-subclass), Inside `SpecialOrder` (multilevel inheritance across packages), Students should simulate object creation in a `Main` class and show which members can be accessed directly, and which are restricted due to **access specifiers and package rules.