You are tasked with designing a basic **e-commerce system** for an online store. The system should demonstrate the principles of **encapsulation**, **inheritance**, and **abstraction**.

1. **Encapsulation**: Implement a Product class that securely stores information about a product, such as ProductID, Name, Price, and StockQuantity. Ensure that:
   * Data is only accessible and modifiable through getter and setter methods.
   * Add validation in setters to prevent invalid data (e.g., negative price or stock).
2. **Inheritance**: Extend the Product class to create specific types of products:
   * ElectronicProduct: Includes additional properties like WarrantyPeriod and Brand.
   * ClothingProduct: Includes additional properties like Size and Material.
3. **Abstraction**: Design an abstract class Order that represents a generic order. This class should:
   * Have properties such as OrderID, CustomerName, and OrderDate.
   * Include an abstract method CalculateTotal() that must be implemented by subclasses.
4. Create a concrete class OnlineOrder that inherits from Order and implements the CalculateTotal() method by summing up the prices of the products in the order.

**Tasks:**

1. **Encapsulation**:
   * Write a Product class with private fields and public getters/setters. Include validation logic.
2. **Inheritance**:
   * Create the ElectronicProduct and ClothingProduct subclasses, each adding unique properties.
3. **Abstraction**:
   * Implement the abstract Order class and its concrete subclass OnlineOrder.
4. Write a main program that:
   * Creates an OnlineOrder with a mix of ElectronicProduct and ClothingProduct.
   * Displays the order details and calculates the total price.

**Expected Output Example:**

* Products in the order:
  + **Laptop**: $1000 (Brand: Dell, Warranty: 2 years)
  + **T-Shirt**: $20 (Size: M, Material: Cotton)
* **Order Total**: $1020