**🎯 Summary — Which Pattern is Best?**

| **Scenario** | **Best Pattern** | **Why?** |
| --- | --- | --- |
| Reusable validators for XML DTOs | ✅ **Strategy Pattern** | Clean, OOP-style reusable validators |
| Complex rule combinations (AND/OR logic) | ✅ **Specification Pattern** | Good for DDD or business rule validation |
| Simple inline validations (functional style) | ✅ **Consumer Pattern** | Lightweight, concise, ideal for DTO validation |
| Sequential, fail-fast validation | ✅ **Chain of Responsibility** | Good if you want to stop at first failure |
| HTML form field validation (browser) | ❌ **Use JS/HTML5 Validation** | Not a Java pattern – use required, pattern, etc. |