GitHub Copilot for Salesforce - Professional Guide

Optimized Prompting Strategies, Standardized Reusable Patterns, and Best Practices

1. Standardized Prompt Formula

[Comment in Code] + [Specific Task] + [Salesforce Context] + [Constraints/Best Practices]
Example:
# Apex class to query active Accounts with Industry = 'Energy',
# return List<Account> sorted by Name, bulkified for trigger context

2. Predefined Prompt Templates for Reuse

# Apex class for CRUD operations on __OBJECT__, bulkified, with unit tests
# Apex trigger on __OBJECT__ to enforce business rule: __RULE__
# Apex test class for __CLASSNAME__ covering positive/negative paths
// LWC component for __FEATURE__, wired to Apex, with error handling
// Aura component to __FUNCTION__, leveraging Lightning Data Service

3. Salesforce-Specific Usage Examples

Apex Class: # Apex class to retrieve all Contacts for given Account Ids, bulkified, no SOQL in loops
Apex Trigger: # Trigger on Opportunity to create a Task when StageName = 'Closed Won'
Apex Test Class: # Test class for OpportunityTriggerHandler with >95% coverage
LWC: // LWC to search Accounts by Name, display in datatable, handle server errors
Aura: // Aura component to list related Contacts for a selected Account

GitHub Copilot for Salesforce - Limitations & Best Practices

4. Known Limitations

- May generate non-bulkified Apex code; requires manual review
- Might not reflect latest Salesforce API versions or features
- Generated LWC/Aura code may lack necessary imports or decorators
- Test classes may not meet organization-specific coverage requirements
- Cannot query or interact with actual Salesforce org data

5. Recommended Best Practices for Optimal Usage

- Use descriptive class, method, and variable names to provide better context
- Break large requirements into smaller, logically grouped comments
- Generate code incrementally; validate after each section
- Review for governor limits, CRUD/FLS compliance, and bulkification
- Explicitly specify Salesforce objects, fields, and logic in prompts

This guide is intended for internal enablement and aligns with Salesforce secure coding guidelines.