

/sp-finishing-a-development-branch

Source: `~/ .claude/skills/sp-finishing-a-development-branch/SKILL.md`

name: finishing-a-development-branch
description: Use when implementation is complete, all tests pass, and you need to decide how to integrate the work - guides completion of development work by presenting structured options for merge, PR, or cleanup

Finishing a Development Branch

Overview

Guide completion of development work by presenting clear options and handling chosen workflow.

Core principle: Verify tests → Present options → Execute choice → Clean up.

Announce at start: "I'm using the finishing-a-development-branch skill to complete this work."

The Process

Step 1: Verify Tests

Before presenting options, verify tests pass:

```
# Run project's test suite
npm test / cargo test / pytest / go test ./...
```

If tests fail:

```
Tests failing (<N> failures). Must fix before completing:
```

```
[Show failures]
```

```
Cannot proceed with merge/PR until tests pass.
```

Stop. Don't proceed to Step 2.

If tests pass: Continue to Step 2.

Step 2: Determine Base Branch

```
# Try common base branches
git merge-base HEAD main 2>/dev/null || git merge-base HEAD master 2>/dev/null
```

Or ask: "This branch split from main - is that correct?"

Step 3: Present Options

Present exactly these 4 options:

```
Implementation complete. What would you like to do?
```

1. Merge back to <base-branch> locally
2. Push and create a Pull Request
3. Keep the branch as-is (I'll handle it later)
4. Discard this work

Which option?

Don't add explanation - keep options concise.

Step 4: Execute Choice

Option 1: Merge Locally

```
# Switch to base branch
git checkout <base-branch>

# Pull latest
git pull

# Merge feature branch
git merge <feature-branch>

# Verify tests on merged result
<test command>

# If tests pass
git branch -d <feature-branch>
```

Then: Cleanup worktree (Step 5)

Option 2: Push and Create PR

```
# Push branch
git push -u origin <feature-branch>

# Create PR
gh pr create --title "<title>" --body "$(cat <<'EOF'
## Summary
<2-3 bullets of what changed>

## Test Plan
```

```
- [ ] <verification steps>
EOF
)"
```

Then: Cleanup worktree (Step 5)

Option 3: Keep As-Is

Report: "Keeping branch . Worktree preserved at ."

Don't cleanup worktree.

Option 4: Discard

Confirm first:

```
This will permanently delete:
- Branch <name>
- All commits: <commit-list>
- Worktree at <path>

Type 'discard' to confirm.
```

Wait for exact confirmation.

If confirmed:

```
git checkout <base-branch>
git branch -D <feature-branch>
```

Then: Cleanup worktree (Step 5)

Step 5: Cleanup Worktree

For Options 1, 2, 4:

Check if in worktree:

```
git worktree list | grep $(git branch --show-current)
```

If yes:

```
git worktree remove <worktree-path>
```

For Option 3: Keep worktree.

Quick Reference

Option	Merge	Push	Keep Worktree	Cleanup Branch
1. Merge locally	✓	-	-	✓
2. Create PR	-	✓	✓	-
3. Keep as-is	-	-	✓	-
4. Discard	-	-	-	✓ (force)

Common Mistakes

Skipping test verification

- **Problem:** Merge broken code, create failing PR
- **Fix:** Always verify tests before offering options

Open-ended questions

- **Problem:** "What should I do next?" → ambiguous
- **Fix:** Present exactly 4 structured options

Automatic worktree cleanup

- **Problem:** Remove worktree when might need it (Option 2, 3)
- **Fix:** Only cleanup for Options 1 and 4

No confirmation for discard

- **Problem:** Accidentally delete work
- **Fix:** Require typed "discard" confirmation

Red Flags

Never:

- Proceed with failing tests
- Merge without verifying tests on result
- Delete work without confirmation
- Force-push without explicit request

Always:

- Verify tests before offering options
- Present exactly 4 options
- Get typed confirmation for Option 4
- Clean up worktree for Options 1 & 4 only

Integration

Called by:

- **subagent-driven-development** (Step 7) - After all tasks complete
- **executing-plans** (Step 5) - After all batches complete

Pairs with:

- **using-git-worktrees** - Cleans up worktree created by that skill

Revision #5

Created 2026-02-18 08:39:55 UTC by John

Updated 2026-06-21 20:00:57 UTC by John