

# IF, AND, OR Practice Workbook Guide

Four Exercises · Healthcare IT Scenarios ·  
TechFitFlow.com

Each tab contains a structured exercise, the correct formula, a common mistake version, and a copy behaviour comparison. Work through each tab in order before moving to the next.

## TAB 1 — IF BASICS: UAT PASS/FAIL LABEL

**Context:** A UAT test log has 40 rows. Column C = Status (Pass/Fail/Blocked/Not Run). Column D should show a label based on status. Column E should show blank when status is Pass, and "Review" for anything else.

### Exercise: D2 — Status label

✓ Correct	✗ Common mistake
<code>=IFS(C2="Pass", "✓ Passed", C2="Fail", "✗ Failed", C2="Blocked", "■ Blocked", TRUE, "■ Not Run")</code>	<code>=IF(C2="Pass", "✓ Passed", IF(C2="Fail", "✗ Failed", "Other"))</code>

*Why it matters:* The mistake version does not handle Blocked or Not Run — all non-Pass non-Fail rows return "Other".

### Exercise: E2 — Review flag

✓ Correct	✗ Common mistake
<code>=IF(C2="Pass", "", "Review")</code>	<code>=IF(C2="Pass", "Passed")</code>

*Why it matters:* Omitting value\_if\_false returns the word FALSE (not blank) when status is not Pass.

## TAB 2 — AND/OR: MULTI-CONDITION CLAIM FLAGS

**Context:** A claims export has Status (col C), Payer (col D), Amount (col E). Three derived columns needed: Review Required (denied/pending/on hold), Priority Review (denied AND amount>1000), Escalate (denied AND specific payers AND amount>1000).

### Exercise: Review Required

✓ Correct	✗ Common mistake
<code>=IF(OR(C2="Denied", C2="Pending", C2="On Hold"), "Review", "")</code>	<code>=IF(C2="Denied", "Review", "")</code>

*Why it matters:* The mistake version only flags denied — misses Pending and On Hold.

### Exercise: Priority Review

✓ Correct	✗ Common mistake
<code>=IF(AND(C2="Denied", E2&gt;1000), "Priority", "")</code>	<code>=IF(C2="Denied", IF(E2&gt;1000, "Priority", ""), "")</code>

*Why it matters:* The mistake uses nested IF for what AND handles in one level — same result, more nesting.

### Exercise: Escalate

✓ Correct	✗ Common mistake
<code>=IF(AND(C2="Denied",OR(D2="BCBS",D2="UHC",D2="Aetna"),E2&gt;DenialThreshold),"Escalate", "")</code>	<code>=IF(AND(C2="Denied",D2="BCBS",E2&gt;DenialThreshold),"Escalate", "")</code>

Why it matters: The mistake only flags BCBS — misses UHC and Aetna. OR handles multiple payer alternatives.

### TAB 3 — NESTED IF VS IFS: DENIAL CATEGORY

**Context:** A denial code column (D) must be categorised into readable groups. Six categories: Prior Auth (codes 15, 197), Coding Error (CO-4, CO-5), COB (CO-22), Missing Info (CO-16, CO-96), Not Denied (any non-denied claim), Other Denial.

#### Exercise: Nested IF version

✓ Correct	✗ Common mistake
<code>=IF(OR(D2="15",D2="197"),"Prior Auth", IF(OR(D2="CO-4",D2="CO-5"),"Coding Error", IF(D2="CO-22","COB", IF(OR(D2="CO-16",D2="CO-96"),"Missing Info", IF(C2&lt;&gt;"Denied","Not Denied","Other")))))</code>	Correct — but 5 levels deep. Adding a new category requires editing the nesting structure.

Why it matters: Works but each new condition adds a bracket pair. Error risk grows with each addition.

#### Exercise: IFS version (preferred)

✓ Correct	✗ Common mistake
<code>=IFS( OR(D2="15",D2="197"), "Prior Auth", OR(D2="CO-4",D2="CO-5"), "Coding Error", D2="CO-22", "COB", OR(D2="CO-16",D2="CO-96"),"Missing Info", C2&lt;&gt;"Denied", "Not Denied", TRUE, "Other Denial")</code>	Same logic, linear structure. Adding a new category = add one condition-result pair.

Why it matters: No catch-all TRUE: IFS returns #N/A for any code not in the list.

### TAB 4 — IFERROR VS IFNA: LOOKUP ERROR HANDLING

**Context:** An XLOOKUP retrieves denial descriptions from a reference table. Some codes in the data are not in the reference table. Three versions: no error handling (shows #N/A), IFERROR (hides all errors), IFNA (handles not-found only).

#### Exercise: No error handling

✓ Correct	Alternative version
<code>=XLOOKUP(D2,CodeTable[Code],CodeTable[Description])</code>	(See context above)

Why it matters: Returns #N/A when code is not in table. Disrupts downstream formulas that reference this cell.

#### Exercise: IFERROR (too broad)

✓ Correct	Alternative version

```
=IFERROR(XLOOKUP(D2,CodeTable[Code],CodeTable[Description]),"Code not found")
```

(See context above)

*Why it matters: Hides #REF! and #VALUE! errors too — structural problems invisible. Use only when all errors are expected.*

### Exercise: IFNA (correct)

✓ Correct

Alternative version

```
=IFNA(XLOOKUP(D2,CodeTable[Code],CodeTable[Description]),"Code not in reference table")
```

(See context above)

*Why it matters: Handles only #N/A (not found). #REF! and #VALUE! still visible — structural errors surface immediately.*