

PhiCode Template Library: Reusable Instruction Prompts

- **Modular Design:** Symbols and patterns are interchangeable
- **Pattern Recognition:** LLMs excel at following symbolic structures
- **Consistent Results:** Same patterns → Similar reasoning quality
- **Minimal Customization:** Change a few symbols, get domain-specific behavior

Core Template Structure

```

 $\forall$  [input_type]  $\rightarrow$  [processing_approach]  $\Rightarrow$  {
  phase.1: [initial_analysis]  $\rightarrow$  {components},
  phase.2: [deeper_analysis]  $\rightarrow$  {insights},
  phase.3: [synthesis]  $\rightarrow$  {conclusions},
  phase.4: [output_generation]  $\rightarrow$  {deliverables},
  domain.[type]: focus.on  $\rightarrow$  [priorities],
  uncertainty:  $\forall$  finding  $\rightarrow$  confidence.score  $\oplus$  evidence.level,
  flags: {🔍 inferred, ⚠️ uncertain, ? missing, 📊 quantified}
}
```

Ready-to-Use Templates

1. Data Analysis Template

```

 $\forall$  dataset  $\rightarrow$  analysis.systematic  $\Rightarrow$  {
  phase.1: data.exploration  $\rightarrow$  {structure, quality, patterns},
  phase.2: statistical.analysis  $\rightarrow$  {descriptive, inferential, significance},
  phase.3: insights.extraction  $\rightarrow$  {trends, anomalies, correlations},
  phase.4: recommendations  $\rightarrow$  {actionable, prioritized, measurable},
  uncertainty:  $\forall$  finding  $\rightarrow$  confidence.score  $\oplus$  evidence.level,
  flags: {📊 quantified, 🔍 inferred, ⚠️ uncertain, ? missing}
}
```

2. Code Review Template

```

 $\forall$  code.submission  $\rightarrow$  review.comprehensive  $\Rightarrow$  {
  phase.1: architecture.analysis  $\rightarrow$  {design.patterns, scalability,
maintainability},
  phase.2: security.assessment  $\rightarrow$  {vulnerabilities, authentication,
data.protection},
  phase.3: performance.evaluation  $\rightarrow$  {efficiency, optimization, bottlenecks},
  phase.4: quality.metrics  $\rightarrow$  {readability, testing, documentation},
}
```

```

    domain.technical: focus.on → [functionality ⊕ reliability ⊕ best.practices],
    flags: {🔧 refactor.needed, ⚡ performance.critical, 🛡 security.concern}
  }

```

3. Business Analysis Template

```

∀ business.proposal → evaluation.strategic ⇒ {
  phase.1: market.analysis → {opportunity, competition, trends},
  phase.2: financial.assessment → {revenue.potential, costs, ROI},
  phase.3: risk.evaluation → {technical, market, operational},
  phase.4: recommendation → {go.no.go, modifications, timeline},
  domain.business: metrics → [growth.potential ⊕ profitability ⊕ feasibility],
  flags: {💰 financial.impact, 📈 growth.opportunity, ⚖ risk.factor}
}

```

4. Research Review Template

```

∀ research.paper → review.scientific ⇒ {
  phase.1: methodology.evaluation → {design, sample, validity},
  phase.2: results.analysis → {statistical.significance, effect.size,
reproducibility},
  phase.3: conclusions.assessment → {supported.by.data, limitations,
implications},
  phase.4: contribution.evaluation → {novelty, significance, impact},
  domain.scientific: standards → [rigor ⊕ validity ⊕ reliability],
  flags: {📊 data.driven, 🧪 experimental.design, 📐 statistical.significance}
}

```

5. Creative Content Template

```

∀ creative.work → analysis.aesthetic ⇒ {
  phase.1: style.analysis → {technique, composition, originality},
  phase.2: content.evaluation → {message, emotion, impact},
  phase.3: technical.assessment → {execution, craftsmanship, innovation},
  phase.4: audience.consideration → {accessibility, engagement, effectiveness},
  domain.creative: focus.on → [expression ⊕ innovation ⊕ impact],
  flags: {🎨 artistic.merit, 💡 innovative.approach, 🎯 audience.appropriate}
}

```

6. Document Summarization Template

```

∀ document → summarization.intelligent ⇒ {
  phase.1: content.mapping → {main.topics, key.points, structure},
  phase.2: importance.ranking → {critical, supporting, supplementary},

```

```
    phase.3: synthesis.creation → {coherent.narrative, logical.flow},
    phase.4: output.formatting → {executive.summary, detailed.points,
    action.items},
    uncertainty: ∀ inference → mark.interpretation 🔍,
    flags: {📋 factual, 🔍 interpreted, ⚠️ unclear, 📊 quantified}
  }
```

🔧 Customization Guide

Minimal Effort Customization (5-10 changes):

```
# Change domain
domain.technical → domain.medical
domain.business → domain.legal

# Update flags
{🔧 refactor.needed} → {💊 dosage.concern}
{💰 financial.impact} → {⚖️ legal.compliance}
```

Moderate Customization (15-20 changes):

```
# Add specialized phases
phase.5: regulatory.compliance → {FDA, safety, documentation}

# Domain-specific focus areas
focus.on → [patient.safety ⊕ efficacy ⊕ side.effects]
```

Template Combinations:

```
# Hybrid analysis
∀ tech.startup.proposal → evaluation.hybrid ⇒ {
  technical.assessment: [USE code.review.template.phases],
  business.evaluation: [USE business.analysis.template.phases],
  synthesis: combine.technical ⊕ business.insights
}
```

⚡ Quick Start Examples

Email Analysis Template

```
∀ email → analysis.communication ⇒ {
  phase.1: intent.detection → {request, information, complaint, praise},
```

```

phase.2: urgency.assessment → {immediate, routine, low.priority},
phase.3: action.identification → {respond, delegate, archive, escalate},
phase.4: response.drafting → {tone, content, timeline},
flags: {🚨 urgent, 📞 call.required, ✉️ email.sufficient}
}

```

Meeting Notes Template

```

∀ meeting.transcript → structuring.systematic ⇒ {
  phase.1: topic.extraction → {agenda.items, discussions, decisions},
  phase.2: action.identification → {tasks, owners, deadlines},
  phase.3: follow.up.mapping → {next.meetings, dependencies, blockers},
  phase.4: summary.generation → {key.decisions, action.items, next.steps},
  flags: {☑️ decided, ⏳ pending, ⌚ deadline.critical}
}

```

Product Review Template

```

∀ product.review → evaluation.comprehensive ⇒ {
  phase.1: feature.analysis → {functionality, usability, performance},
  phase.2: comparison.study → {competitors, alternatives, market.position},
  phase.3: user.experience → {satisfaction, pain.points, recommendations},
  phase.4: purchase.recommendation → {pros, cons, best.use.cases},
  flags: {★ standout.feature, ⚠️ potential.issue, 💰 value.proposition}
}

```

🌀 Benefits of PhiCode Templates

Consistency

- Same symbolic patterns → Predictable reasoning quality
- Standardized output formats across different users
- Reliable performance across time and domains

Adaptability

- Easy domain switching with minimal changes
- Modular components can be mixed and matched
- Conditional logic handles edge cases automatically

Scalability

- Templates can be extended with additional phases
- Multiple templates can be combined for complex analysis
- New domains can be added using existing patterns

Quality

- Enforces systematic thinking through structured phases
 - Built-in uncertainty handling prevents overconfidence
 - Domain-specific flags ensure appropriate analysis depth
-

Pro Tips for Template Creation

1. **Start with the core structure** and modify incrementally
2. **Use meaningful phase names** that guide reasoning
3. **Include appropriate domain flags** for visual clarity
4. **Test with diverse inputs** to validate adaptability
5. **Combine templates** for multi-domain analysis
6. **Maintain symbol consistency** for pattern recognition