**Summary of Steps to Answer G1-C1**

1. **Connect Core Classes**:
   * Use suitableFor to link Feedback (core class) to Context (core class).
2. **Create Subclasses**:
   * Define TimeSensitiveContext under TemporalContext.
   * Define feedback subclasses (VisualFeedback, AuditoryFeedback, etc.).
3. **Add Data Property**:
   * Attach hasResponseTime (domain=Feedback, range=xsd:integer) to track response times.
4. **Create Instances**:
   * Feedback instances: VisualAlert (250ms), AuditoryBeep (350ms).
   * Context instance: EmergencyDashboardContext (type=TimeSensitiveContext).
5. **Link Feedback to Context**:
   * Assert :VisualAlert :suitableFor :EmergencyDashboardContext.
6. **Add Restriction**:
   * Use Protégé’s class expression editor to enforce:  
     *"All feedback mechanisms linked to TimeSensitiveContext must have hasResponseTime ≤300."*
7. **Run Reasoner**:
   * Start HermiT/Pellet to validate consistency.
   * **Result**: AuditoryBeep (350ms) is flagged as inconsistent with TimeSensitiveContext.
8. **SPARQL Query**:

sparql

Copy

Download

SELECT ?feedback ?time

WHERE {

?feedback :hasResponseTime ?time ;

:suitableFor/:a :TimeSensitiveContext .

FILTER (?time <= 300)

}

**Result**: Only VisualAlert (250ms) is returned.