Education & Learning / Quiz Generation

Tag and categorize questions by topic, difficulty, and cognitive level — content taxonomy for efficient quiz assembly.
Difficulty: Intermediate
Model: GPT-4 / Claude / Gemini
Use Case: Content Management, Quiz Assembly
Updated: June 2026
Why This Prompt Exists
Disorganized question banks are unusable. Without tagging, teachers can’t find questions by topic, difficulty, or type. Most question collections are flat lists — no search, no filtering.

You get:

- questions buried in long lists (can’t find what you need)
- no way to filter by difficulty or topic
- duplicate questions across teachers (no sharing system)
- inconsistent quality and standards
- time wasted searching instead of teaching

But organized question banks have structure:

- topic tags: subject, unit, chapter, concept
- difficulty level: easy, medium, hard
- cognitive level: remember, understand, apply, analyze, evaluate, create

- question type: multiple choice, short answer, essay, matching
- usage tracking: used on which assessments, performance data

Without organization, question banks are unusable.
This prompt organizes questions into searchable, tagged databases.

The Prompt

```
Assume the role of a curriculum database designer who organizes
question banks.

Your task is to tag and categorize questions for easy retrieval.

Generate:

1. QUESTION BANK STRUCTURE

**Topic Hierarchy**
- Domain: [broad subject area]
  - Unit: [major topic]
    - Chapter: [specific topic]
      - Concept: [individual learning objective]

2. QUESTION METADATA TEMPLATE

{
  "id": "QZ-001",
  "topic_hierarchy": {
    "domain": "",
    "unit": "",
```

```
    "chapter": "",
    "concept": ""
  },
  "difficulty": "easy|medium|hard",
  "cognitive_level":
"remember|understand|apply|analyze|evaluate|create",
  "question_type":
"multiple_choice|short_answer|essay|matching|fill_in",
  "estimated_time": "30s|60s|90s|2min|5min",
  "tags": [],
  "usage_count": 0,
  "performance_data": {
    "p_value": null,
    "discrimination": null
  }
}
```

3. TOPIC TAG INVENTORY

Topic Level	Tag Name	Parent Tag	Sub-tags
Domain	[name]	None	[units]
Unit	[name]	[domain]	[chapters]
Chapter	[name]	[unit]	[concepts]
Concept	[name]	[chapter]	None

4. DIFFICULTY CALIBRATION

| Difficulty | Success Rate Target | Time Estimate | Question Features

|
|------------|---------------------|--------------|------------------
-|
| Easy | 80-90% | 30-60s | Direct recall, one step |
| Medium | 60-80% | 60-90s | Two steps, simple application |
| Hard | 40-60% | 90-120s | Multiple steps, analysis required |
| Challenge | 20-40% | 2-5min | Synthesis, evaluation, creation |

5. QUESTION BANK INVENTORY

| QID | Question | Topic | Difficulty | Cognitive Level | Type | Usage
|
|-----|----------|-------|------------|----------------|------|-------
|
| [ID] | [first 50 chars] | [tag] | [E/M/H] | [level] | [type] |
[count] |

6. RETRIEVAL QUERY PATTERNS

Query Type	Example	Use Case
Topic filter	"topic:photosynthesis"	Find all photosynthesis
questions
Difficulty filter	"difficulty:hard"	Challenge students
Cognitive level	"level:analyze"	Build higher-order thinking
Combined	"topic:algebra AND difficulty:hard AND level:apply"
Targeted assessment |

7. QUESTION BANK QUALITY METRICS

Metric	Description	Target
P-value	Proportion of students answering correctly	0.4-0.8 for medium questions
Discrimination	Difference between high and low performers	>0.3
Distractor effectiveness	Each distractor chosen by some students	>5% per distractor
Usage frequency	How often question is used	Distributed evenly

8. COMMON ORGANIZATION MISTAKES

Mistake	Why It Fails	Correct Approach
Flat list structure	Can't find questions	Hierarchical tagging
No difficulty calibration	Unbalanced assessments	Calibrate with student data
Inconsistent tagging	Can't filter reliably	Standardized tag list
No usage tracking	Unknown question quality	Log performance data
Duplicate questions	Wasted effort	Central question bank

INPUTS:

Subject area:
[PASTE SUBJECT AREA]

Existing questions (if any):
[PASTE QUESTIONS OR "NEW BANK"]

```
Number of questions to organize:
[PASTE NUMBER]

Tagging preferences (optional):
[PASTE PREFERENCES]

RULES:
- Use hierarchical topic tags (domain → unit → chapter → concept)
- Tag every question with difficulty (easy/medium/hard) based on
success rate targets
- Tag every question with cognitive level (Bloom's taxonomy)
- Track usage and performance data for quality improvement
- Avoid duplicate questions (check before adding)
- Calibrate difficulty with real student data when available
- Review question bank annually (retire poor questions, add new ones)
```

How To Use It

- Use hierarchical topic tags — domain → unit → chapter → concept for granular searching.
- Tag every question with difficulty (easy/medium/hard) — based on success rate targets, not intuition.
- Tag every question with cognitive level — Bloom’s taxonomy for balanced assessment.
- Track usage and performance data — P-values and discrimination indices reveal question quality.
- Avoid duplicate questions — check the question bank before adding new questions.
- Calibrate difficulty with real student data when available — adjust tags based on actual performance.
- Review the question bank annually — retire poor questions, add new ones, update tags.

Example Input

Subject area:
“High School Biology”

Existing questions:
“10 questions about photosynthesis, 5 about cellular respiration, 3 about DNA structure”

Number of questions to organize:
“18”

Tagging preferences:
“Add domain, unit, chapter, concept tags; include difficulty and cognitive level”

Why It Works
Flat question lists are unusable at scale. Without tagging, teachers waste time searching instead of teaching.
This framework improves outcomes by forcing: hierarchical topic tagging, difficulty calibration, cognitive level classification, usage tracking, and quality metrics.
Failure modes this prevents: Buried questions, no filtering, duplicates, inconsistent quality, time wasted searching.
This improves on: Flat question lists. Organized question banks enable efficient quiz assembly.
Related to: QZ-01 (Bloom’s) for cognitive tags; QZ-02 (Distractor) for multiple-choice quality.

Build Better AI Systems

Subscribe for advanced prompt engineering, AI coding tools, debugging frameworks, and practical strategies for developers and engineers.

Carefully engineered prompts for people doing real work.