## **Coding Assessment for KK Course Bundle System**

### **Problem Domain**

KK, a platform providing education resources, wants to create a **course bundle recommendation system** that returns bundle quotes based on teacher requests.

**Scenario**:  
Teachers can request different levels of content coverage for various course topics. For example, a teacher may want **20 resources on reading**, **50 on math**, and **30 on science**. This request specifies the level of content needed for each topic to create a customized bundle for their classroom.

Based on the teacher’s request, quotes are provided by a set of resource providers. Each provider can offer different rates for bundles depending on the topics they cover. Here is the pricing approach:

### **What to Build**

Build an application that generates resource bundle quotes based on teacher requests. Here’s an example of a teacher’s request in JSON format:

json

Copy code

{

"topics": {

"reading": 20,

"math": 50,

"science": 30,

"history": 15,

"art": 10

}

}

The JSON request specifies the amount of content the teacher is requesting for five possible topics.

The application should use a static configuration file (also in JSON) listing each provider’s available topics:

json

Copy code

{

"provider\_topics": {

"provider\_a": "math+science",

"provider\_b": "reading+science",

"provider\_c": "history+math"

}

}

The application should generate quotes from each provider based on the topics requested. The pricing model is as follows:

1. **If 2 topics match** with a provider’s offering, the quote is **10%** of the combined requested content level for those topics.
2. **If only 1 topic matches**, the quote depends on the importance of the topic:
   * **20%** for the highest requested topic.
   * **25%** for the second-highest topic.
   * **30%** for the third-highest topic.

The application should consider only the **top 3 topics** requested by the teacher, discarding the other two.

The system should not return a quote if the calculated value is zero (0).

### **Example**

Given this request:

json

Copy code

{

"topics": {

"reading": 20,

"math": 50,

"science": 30,

"history": 15,

"art": 10

}

}

The application will select the top 3 requested topics and ignore the others:

* **math**: 50
* **science**: 30
* **reading**: 20

The application will then calculate the following quotes:

* **provider\_a**: 8 (10% of 80 for two matches on math and science)
* **provider\_b**: 5 (25% of 20 for one match on reading, the 3rd biggest topic)
* **provider\_c**: 12.5 (25% of 50 for one match on math, the 2nd biggest topic)

### **Assessment Criteria**

* **Code Structure and Readability**: How well-organized and readable is the code? Are the classes and methods structured logically?
* **Object-Oriented Design**: How well does the candidate apply object-oriented principles (e.g., encapsulation, inheritance)?
* **Testing**: Encourage the candidate to write simple unit tests to validate their methods.

### **Notes for the Candidate**

* You may use the latest version of the language and framework.
* Focus on clear, maintainable code rather than implementing a fully comprehensive solution.
* Feel free to ask any clarifying questions as if you had access to a project owner.

Link to repository:

Questions:

* Are all five topics (reading, math, science, history, art) fixed, or will the topics list expand in the future?
* Can there be additional attributes in the teacher’s request?
* Can a provider offer topics beyond the five listed in the example?
* How should mismatched topics (not requested by the teacher) be treated in the calculation?
* Why **provider\_c**: 12.5 25% of 50 for one match on math if math is 1st match? => 20%/50
* Should the application validate the provider topics, or can it assume all input is correct?
* What should the app do if:
* No providers match any of the teacher’s requested topics?
* The provider configuration is missing or invalid?