

Scenario #1: A convention is coming to town, and local residents want to make some money by renting out all or part of their homes. In true do-it-yourself mode, they decide to build an online listings system from scratch. You've been hired to plan the data structures for the listings.

Each listing contains the rental price for the weekend, the number of beds the unit has, the street address, and an optional photo. The site also stores the email address of the person who posted each listing, but the email addresses won't be displayed on the site.

When a renter visits the site, they can search on either or both of the number of beds or the price range. The site displays those listings that match the search requirements.

Listings get added to or removed from the site throughout the time that the site is up.

Question 1: Which problem actions occur frequently, and thus need an efficient run-time?

Question 2: What data structure(s) do you propose for the listings? Be sure to include details such as the type of elements in lists, the types of keys/values in hashmaps, etc.

Justify your data structure choice, in light of Question 1

Scenario #2: With all the online work going on these days, you've decided to create a web-based document editor (like a simplified Google Docs. Users of your tool will edit documents through a web client, with the actual document contents stored on a server. To keep the project manageable, you'll focus on four operations: adding text, deleting text, undoing edits, and searching for words in the document. The editor should allow documents to have basic styling, such as boldface words/phrases and section headings.

In particular, searching for words needs to be fast (because you imagine people writing large documents in your tool once it becomes popular).

Question 1: What data do you need to manage for this problem?

Question 2: What data structure(s) do you propose? Be sure to include details such as the type of elements in lists, the types of keys/values in hashmaps, whether objects are mutable or immutable, etc.

Justify your data structure choices