Big Depot

A depot has a number of storage areas (at most 20) organized in 4 floors (1, 2, 3, 4), with the front desk at floor 1 (ground floor).
Each storage area has a unique 4 symbols alphanumeric identification, and contains a number of shelves (at most 10).
Each shelf has a numeric identification, and can accommodate a limited number of items (between 20 and 40).
Items are identified by their code (13-digit unique number), brand, name, and manufacturing date (4-digit year, 2-digit month, 2-digit day). There are items which are lent, and thus they are not on any shelf. When an item is returned, it should be placed on the shelf where it was located before lending.

Two static methods are provided, in the package **`depot`**:

`generateItemInfo:` which generates information about a new item, and returns an array of strings with code (13-digit unique number), brand, name, and manufacturing date (4-digit year, 2-digit month, 2-digit day) in this order, and has no arguments.

generateStorageAreaInfo: which generates info for a storage area, and returns and array of strings with identification and number of shelves in this order

Develop a stand-alone application which supports the following operations:

1. Populate the depot with items
2. Lend a number of items (at random)
3. Lend a specific item (identified by its code)
4. List all the items of a given brand, complete with location and number of samples
5. List all the items located in a given storage area, on a certain shelf
6. List all the items which are lent
7. Insert an item in the first available slot (remember there are lent items), as close as possible to the other items of the same brand
8. Find an item in the depot indicating the storage area and shelf (note that it might be lent)