Autocomplete for form validation

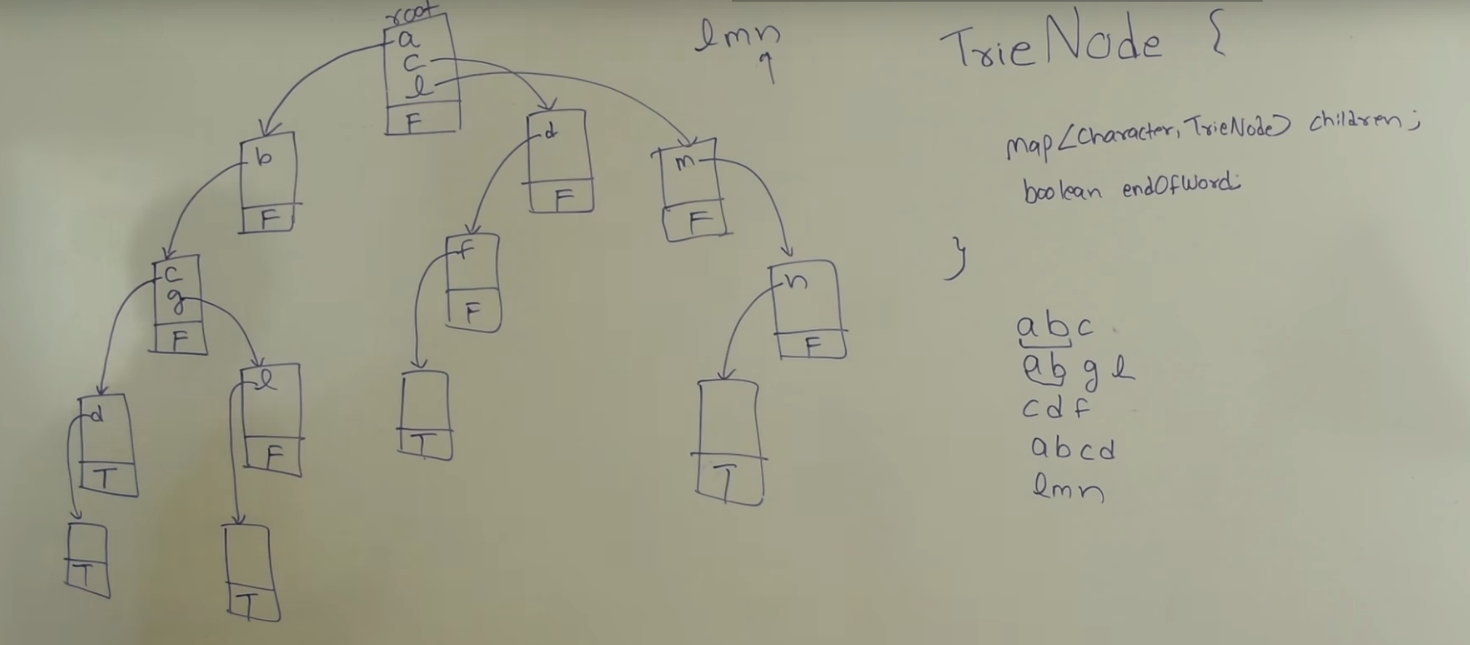
# Trie Based approach

**Why Trie?**

It is an efficient information retrieval data structure.

Using trie, search complexities can be brought to optimal limit (key length). If we store keys in binary search tree, a well balanced BST will need time proportional to **M \* log N**, where M is maximum string length and N is number of keys in tree. Using trie, we can search the key in O(M) time. **Since lookups are more in autocomplete Trie is suitable data structure considering little trade off in creation time**.

However penalty is on trie storage requirements. Time complexity for Trie creation is O(N\*L) where N is number of word and L is average length of word



Typical Trie Data Structure looks as above which has abc, abgl, cdf, abcd, lmn stored in it. T and F are Booleans which represent end of word

**Logic of Trie implementation in our Prefix Based auto suggestions**

* For every prefix user enter, we find the matched Trie node by traversing the prefix
* Get all the children of the matched node recursively which are complete word and populate auto complete suggestion (can be improved further if number of suggestions are limited)
* If user further enter a character then the search is further narrowed deep in the trie structure which in return improves efficiency

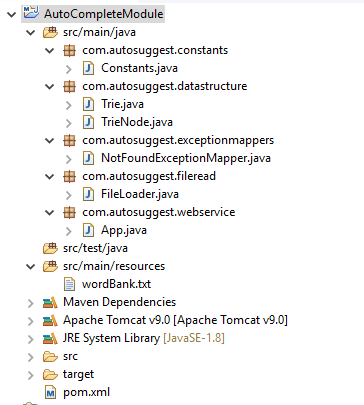
**Assumptions**

* No limit on number of suggestions, currently showing all filtered autocomplete suggestions. If limit imposed, it will improve further efficiency of the algorithm and usability
* Special characters are not neglected in auto suggestions
* JSON response format from rest service is assumed since UI validations are mostly done using Java Script which understands JSON easily than XML

**References:**

1. <https://www.cs.purdue.edu/homes/ayg/CS251/slides/chap11.pdf>
2. <https://www.toptal.com/java/the-trie-a-neglected-data-structure>
3. <http://sujitpal.blogspot.in/2007/02/three-autocomplete-implementations.html>

**Project Structure**

****

**GIT Link:** [**https://github.com/ncsbalaji/AutoCompleteModule/**](https://github.com/ncsbalaji/AutoCompleteModule/)

**Rest URL Format and inputs:**

[**http://localhost:8080/AutoSuggest/suggest/query?prefix={prefix}&isEnd={boolean}**](http://localhost:8080/AutoSuggest/suggest/query?prefix=%7bprefix%7d&isEnd=%7bboolean%7d)

1. To get all suggestions pass any prefix in {prefix} and isEnd = false
2. To validate entry from user ie., if user press Enter Key pass isEnd = true
3. Input data source is at location: \src\main\resources\wordBank.txt

**Sample Output**

