# Trie

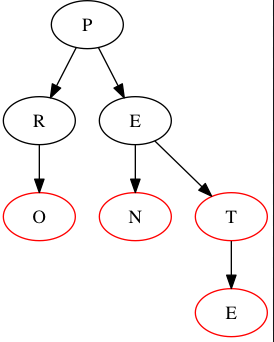
## What is trie?

This is similar to search tree used to store and search the strings in time and space efficient manner

Node can store non-repetitive multiple character

Every node contains link to next node

Every node keeps track of end of the string



## Why Trie?

Trie is used in several string based operations:

1. Spelling checker
2. Search auto-complete

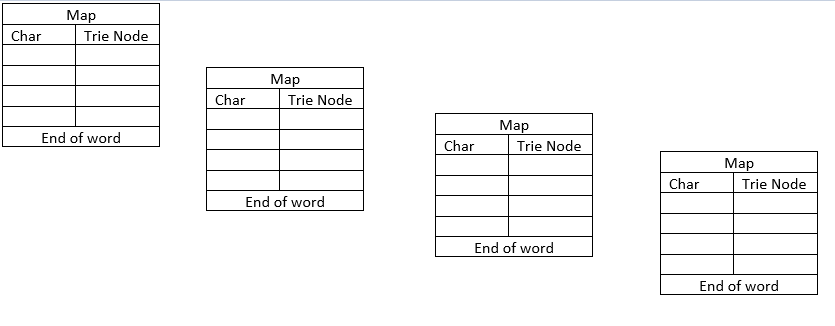
# Creation of Trie

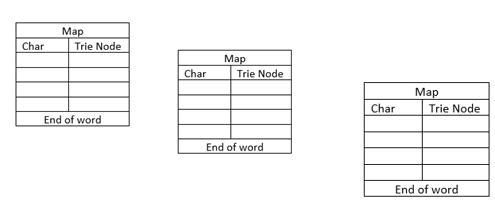
Case 1: Trie is empty (set)

Case 2: new character/string is common to another string pre-fix (Send)

Case 3: New string is already available as complete string(Sen)

Case 4: String to be inserted is already in Trie (Seth)



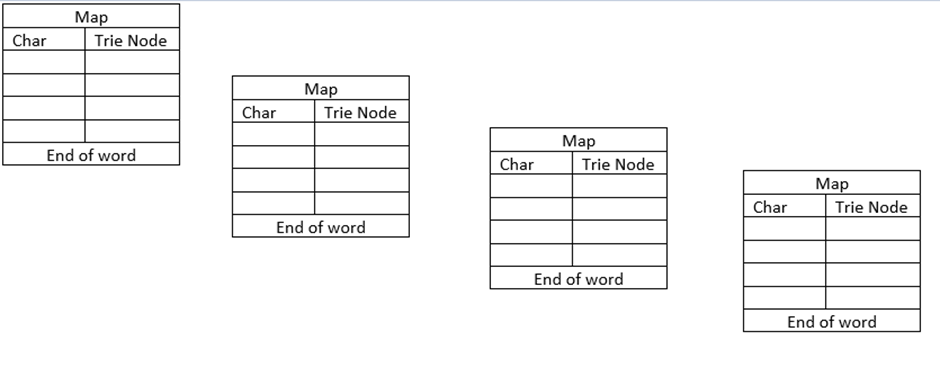


# Search of Trie

Case 1: Trie is not available

Case 2: Trie is available in the

Case 3; Trie available is the prefix of the search value



# Deletion of Trie

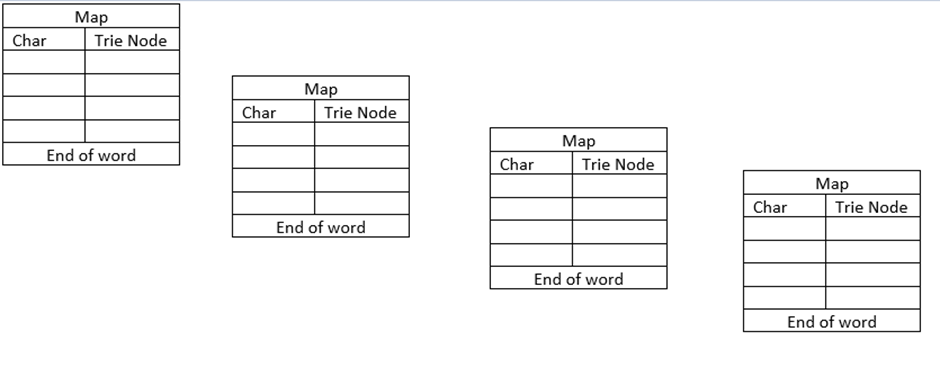
Deletion happens from leaf to root

Case 1: If the word deleted belongs to another word.

Case 2: Current word is prefix of another word

Case 3: Some other word is prefix of this word

Case 4: No dependency



# Practical uses of Trie

Autocomplete

Spelling checker