# 208. Implement Trie (Prefix Tree)

## SOLUTION IN JAVA

class TrieNode {

public TrieNode[] children = new TrieNode[26];

public boolean isWord = false;

}

class Trie {

public void insert(String word) {

TrieNode node = root;

for (final char c : word.toCharArray()) {

final int i = c - 'a';

if (node.children[i] == null)

node.children[i] = new TrieNode();

node = node.children[i];

}

node.isWord = true;

}

public boolean search(String word) {

TrieNode node = find(word);

return node != null && node.isWord;

}

public boolean startsWith(String prefix) {

return find(prefix) != null;

}

private TrieNode root = new TrieNode();

private TrieNode find(String prefix) {

TrieNode node = root;

for (final char c : prefix.toCharArray()) {

final int i = c - 'a';

if (node.children[i] == null)

return null;

node = node.children[i];

}

return node;

}

}