Name: Samuel Baah

Index number: 3381122

Ref number: 20907123

**CODE:**

// Trie Node definition

class TrieNode {

constructor() {

this.children = {}; // Map from character to TrieNode

this.isEndOfWord = false; // True if this node marks the end of a word

}

}

// Trie class implementation

class Trie {

constructor() {

this.root = new TrieNode(); // Root node of the Trie

}

/\*\*

\* Inserts a word into the trie.

\* @param {string} word

\*/

insert(word) {

let node = this.root;

for (const char of word) {

if (!(char in node.children)) {

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

}

node = node.children[char];

}

node.isEndOfWord = true;

}

/\*\*

\* Returns true if the word is in the trie.

\* @param {string} word

\* @return {boolean}

\*/

search(word) {

let node = this.root;

for (const char of word) {

if (!(char in node.children)) {

return false;

}

node = node.children[char];

}

return node.isEndOfWord;

}

/\*\*

\* Returns true if there is any word in the trie that starts with the given prefix.

\* @param {string} prefix

\* @return {boolean}

\*/

startsWith(prefix) {

let node = this.root;

for (const char of prefix) {

if (!(char in node.children)) {

return false;

}

node = node.children[char];

}

return true;

}

}

**Example Test Run**

const trie = new Trie();

trie.insert("apple");

console.log(trie.search("apple")); // true

console.log(trie.search("app")); // false

console.log(trie.startsWith("app")); // true

trie.insert("app");

console.log(trie.search("app")); // true

**Output**

true

false

true

true