Problem 677: Map Sum Pairs

Problem Information

Difficulty: Medium
Acceptance Rate: 57.00%
Paid Only: No
Tags: Hash Table, String, Design, Trie

Problem Description

Design a map that allows you to do the following:

* Maps a string key to a given value. * Returns the sum of the values that have a key with a prefix equal to a given string.

Implement the `MapSum` class:

* `MapSum()` Initializes the `MapSum` object. * `void insert(String key, int val)` Inserts the `key-val` pair into the map. If the `key` already existed, the original `key-value` pair will be overridden to the new one. * `int sum(string prefix)` Returns the sum of all the pairs' value whose `key` starts with the `prefix`.

Example 1:

Input ["MapSum", "insert", "sum", "insert", "sum"] [[], ["apple", 3], ["ap"], ["app", 2], ["ap"]] **Output** [null, null, 3, null, 5] **Explanation** MapSum mapSum = new MapSum(); mapSum.insert("apple", 3); mapSum.sum("ap"); // return 3 (_ap_ ple = 3) mapSum.insert("app", 2); mapSum.sum("ap"); // return 5 (_ap_ ple + _ap_ p = 3 + 2 = 5)

Constraints:

* `1 <= key.length, prefix.length <= 50` * `key` and `prefix` consist of only lowercase English letters. * `1 <= val <= 1000` * At most `50` calls will be made to `insert` and `sum`.

Code Snippets

C++:

```
class MapSum {
public:
MapSum() {

}

void insert(string key, int val) {

}

int sum(string prefix) {

}
};

/**
* Your MapSum object will be instantiated and called as such:
* MapSum* obj = new MapSum();
* obj->insert(key,val);
* int param_2 = obj->sum(prefix);
*/
```

Java:

```
class MapSum {

public MapSum() {

}

public void insert(String key, int val) {

}

public int sum(String prefix) {

}
}
```

```
/**
* Your MapSum object will be instantiated and called as such:
* MapSum obj = new MapSum();
* obj.insert(key,val);
* int param_2 = obj.sum(prefix);
*/
```

Python3:

```
class MapSum:

def __init__(self):


def insert(self, key: str, val: int) -> None:


def sum(self, prefix: str) -> int:



# Your MapSum object will be instantiated and called as such:
# obj = MapSum()
# obj.insert(key,val)
# param_2 = obj.sum(prefix)
```