5430. Design Browser History

My Submissions (/contest/weekly-contest-192/problems/design-browser-history/submissions/)

Back to Contest (/contest/weekly-contest-192/)

User Accepted:	1
User Tried:	1
Total Accepted:	1
Total Submissions:	1
Difficulty:	Medium

You have a **browser** of one tab where you start on the `homepage` and you can visit another `url`, get back in the history number of `steps` or move forward in the history number of `steps`.

Implement the `BrowserHistory` class:

- `BrowserHistory(string homepage)` Initializes the object with the `homepage` of the browser.
- `void visit(string url)` visits `url` from the current page. It clears up all the forward history.
- `string back(int steps)` Move `steps` back in history. If you can only return `x` steps in the history and `steps > x`, you will return only `x` steps. Return the current `url` after moving back in history **at most** `steps`.
- `string forward(int steps)` Move `steps` forward in history. If you can only forward `x` steps in the history and `steps > x`, you will forward only `x` steps. Return the current `url` after forwarding in history **at most** `steps`.

Example:

```
Input:
["BrowserHistory","visit","visit","visit","back","back","forward","visit","forward","back'

[["leetcode.com"],["google.com"],["facebook.com"],["youtube.com"],[1],[1],[1],["linkedin.c
Output:
[null,null,null,null,"facebook.com","google.com","facebook.com",null,"linkedin.com","googl

Explanation:
BrowserHistory browserHistory = new BrowserHistory("leetcode.com");
browserHistory.visit("google.com");       // You are in "leetcode.com". Visit "google.com'
browserHistory.visit("facebook.com");     // You are in "google.com". Visit "facebook.com'
browserHistory.visit("youtube.com");      // You are in "facebook.com". Visit "youtube.con
browserHistory.back(1);                   // You are in "youtube.com", move back to "facet
browserHistory.back(1);                   // You are in "facebook.com", move back to "goog
browserHistory.forward(1);                // You are in "google.com", move forward to "fac
browserHistory.visit("linkedin.com");     // You are in "facebook.com". Visit "linkedin.cc
browserHistory.forward(2);                // You are in "linkedin.com", you cannot move fc
browserHistory.back(2);                   // You are in "linkedin.com", move back two step
browserHistory.back(7);                   // You are in "google.com", you can move back or
```

Constraints:

- `1 <= homepage.length <= 20`
- `1 <= url.length <= 20`
- `1 <= steps <= 100`
- `homepage` and `url` consist of '.' or lower case English letters.
- At most `5000` calls will be made to `visit`, `back`, and `forward`.

JavaScript

```
/**
 * @param {string} homepage
 */
var BrowserHistory = function(homepage) {

};

/**
 * @param {string} url
 * @return {void}
 */
BrowserHistory.prototype.visit = function(url) {

};

/**
 * @param {number} steps
 * @return {string}
 */
BrowserHistory.prototype.back = function(steps) {

};

/**
 * @param {number} steps
 * @return {string}
 */
BrowserHistory.prototype.forward = function(steps) {

};

/**
 * Your BrowserHistory object will be instantiated and called as such:
 * var obj = new BrowserHistory(homepage)
 * obj.visit(url)
 * var param_2 = obj.back(steps)
 * var param_3 = obj.forward(steps)
 */
```

☐ **Custom Testcase** Use Example Testcases