**Remove Duplicates**

[string](http://www.practice.geeksforgeeks.org/tag-page.php?tag=string&isCmp=0)[Microsoft](http://www.practice.geeksforgeeks.org/tag-page.php?tag=Microsoft&isCmp=1)

Given a string, remove duplicates from it. Note that original order of characters must be kept same.  Expected time complexity O(n) where n is length of input string and extra space O(1) under the assumption that there are total 256 possible characters in a string.

**Input:**First line of the input is the number of test cases T. And first line of test case contains a string.  
  
**Output:**Modified string without duplicates and same order of characters.  
  
**Constraints:**  Input String Length <= 1000  
  
**Example:**  
**Input:**  
2  
geeksforgeeks  
geeks for geeks

**Output:**  
geksfor  
geks for

\*\*For More Examples Use Expected Output\*\*

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=462>

**import** java.util.\*;

**import** java.lang.\*;

**import** java.io.\*;

**class** GFG {

**public** **static** **void** main(String[] args) {

*// TODO code application logic here*

        Scanner sc = **new** Scanner(System.in);

**int** t= Integer.parseInt(sc.nextLine());

**while**(t-- > 0) {

            String s = sc.nextLine();

            String ans = "";

**for**(**int** i =0; i<s.length(); i++) {

**if**(!ans.contains(String.valueOf(s.charAt(i)) )){

                    ans += s.charAt(i);

                }

            }

            System.out.println(ans);

        }

    }

}