Chapter-1 Arrays And Strings

Implement an algorithm to determine if a string has all unique characters.

Approach 1 :   
1. Create a HashSet object.  
2. Scan the whole string, and add each character one by one to the HashSet object  
3. If the add object  returns true then continue  
    else return false

Approach 2 : Brute Force technique  
Run 2 loops with variable i and j. Compare str[i] and str[j]. If they become equal at any point, return false.  
Time Complexity: O(n2)

Approach 3 : Sorting

Using sorting based on ASCII values of characters  
Time Complexity: O(n log n)

Approach 4 :

1.  For this method we need to know about  two inbuilt functions in java , indexOf() which returns the index of first occurence of the character in the string , while second function lastIndexOf() returns the index of last occurence of the character in the given string.  
2. First , we convert the given inputstring into characterarray by using toCharArray() function.  
3. Calculate the indexOf() and lastIndexOf() for each character in the given inputstring  
4. If both are equal then continue and make result= trueelse set flag result = false  
5. Return result

Approach 5: Without Extra Data Structure

we iterate over the string, we find the int value of the character with respect to ‘a’ with the statement int bitAtIndex = str.charAt(i)-‘a’;   
Then the bit at that int value is set to 1 with the statement 1 << bitAtIndex .  
Now, if this bit is already set in the checker, the bit AND operation would make checker > 0. Return false in this case.  
Else Update checker to make the bit 1 at that index with the statement checker = checker | (1 <<bitAtIndex);

**Time Complexity**: O(n)