Our Solution(s)   Run Code

Solution 1

```java
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

import java.util.*;

class Program {
  // O(n) time | O(min(n, a)) space
  public static String longestSubstringWithoutDuplication(String str)
    Map<Character, Integer> lastSeen = new HashMap<Character, Integer
    int[] longest = {0, 1};
    int startIdx = 0;
    for (int i = 0; i < str.length(); i++) {
      char c = str.charAt(i);
      if (lastSeen.containsKey(c)) {
        startIdx = Math.max(startIdx, lastSeen.get(c) + 1);
      }
      if (longest[1] - longest[0] < i + 1 - startIdx) {
        longest = new int[] {startIdx, i + 1};
      }
      lastSeen.put(c, i);
    }
    String result = str.substring(longest[0], longest[1]);
    return result;
  }
}
```

Your Solutions   Run Code

Solution 1   Solution 2   Solution 3

```java
class Program {
  public static String longestSubstringWithoutDuplication(String str)
    // Write your code here
    return "";
  }
}
```

Our Tests

Custom Output   Submit Code

Run or submit code when you're ready.