Longest Non-Repeating Substring

Background: The challenge is to write a function that finds and returns the longest portion of a string that has no repeating characters.

Challenge: LeetCode - 3. Longest Substring Without Repeating Characters

Resources: MDN Web Docs and w3Schools

Notes: The code below is intended to be informational and is not meant to be the only way to write a function that determines the longest non-repeating substring. In addition to the function below, I employed other functions and methods to set conditions, extract data and facilitate user interaction.

```javascript
function lengthOfLongestSubstring(str) {

    // create a set to store the longest substring
        //Why? JS set cannot contain duplicate elements
    let longestStrLength = 0;
    let newArray = [];

    // check if the str is blank, challenge condition
    if (str != ""){
        // if not blank, loop through the provided string
        for (let i = 0; i < str.length; i++){
            // Use a set to store the string created from the current point
            let stringSet = new Set();

            // Use an internal loop to evaluate the letters starting at the current
              //point
            if (stringSet.has(str[x])) {
              // If it does, stop because it already exists in the set
              break;
            } else {
                // the letter is added to the set because it's not found
                stringSet = stringSet.add(str[x]);
            };
        }
        // assign longestStrLength to the greater numerical value(longestStrLength
        // or stringSet)
        longestStrLength = Math.max(
          longestStrLength,
          stringSet.size
        );
        // use the spread operator to copy/convert the set to the new array
```

```javascript
            newArray = [...stringSet]
            return longestStrLength + " and " + newArray.join('');
        }
    }else {
        // the string is blank
        return 0;
    }
};

// TESTING THE FUNCTION
console.log(lengthOfLongestSubstring("abcabcbb"));  // 3 and abc
console.log(lengthOfLongestSubstring("bbbbb"));  //  1 and b
console.log(lengthOfLongestSubstring("pwwkew"));  // 2 and pw
console.log(lengthOfLongestSubstring("lokmmmhy"));  // 4 and lokm
console.log(lengthOfLongestSubstring("fudrrweqtn"));  // 4 and fudr
console.log(lengthOfLongestSubstring(""));  //  0
```