জাভাস্ক্রিপ্টে স্ট্রিং (String) হচ্ছে কিছু অক্ষরের সমষ্টি। যেমন "Hello Webcoachbd" বা 'How many tutorials you have' এগুলি হচ্ছে স্ট্রিং। ডাবল বা সিঙ্গেল কোটেশনের ভিতরে কিছু রাখলেই সেটা স্ট্রিং হিসেবে গন্য হয়। এমনকি '15' এটাও একটা স্ট্রিং। স্ট্রিং এ সিঙ্গেল না ডাবল কোটেশন দিবেন এটা নিজের ইচ্ছা। দুটি স্ট্রিং

ব্যাকস্ল্যাশ (\) দিয়ে কোটেশন এসকেপ করতে হয়। শুধু কোটেশনের সামনে ব্যাকস্ল্যাশ দিলেই হয়।

\*\* যদি পুরো স্ট্রিং সিঙ্গেল কোটেশনের ভিতর রাখেন তাহলে ভিতরে ডাবল কোটেশন রাখতে পারেন, কাজ হবে তবে সিঙ্গেল কোটেশনের ভিতর সিঙ্গেল কোটেশন দিলে এসকেপ করতে হবে। তদ্রুপ ডাবল কোটেশনের ভিতর সিঙ্গেল কোটেশন দিলে এসকেপ করতে হবেনা কিন্তু ডাবল কোটেশন দিলে এসকেপ করতে হবে যেমন আমি শেষ লাইনে করেছি।

যাইহোক স্ট্রিং নিয়ে কাজ করার জন্য জাভাস্ক্রিপ্টে প্রচুর ফাংশন এবং প্রোপার্টিজ আছে, যেমন .length দিয়ে স্ট্রিং এর দৈর্ঘ্য বের করা যায়।

1.var x = 'Hello Webcoachbd';

2.document.write(x.length);

**কিছু মেথড**

=============

[**charAt(index)**](http://www.webcoachbd.com/javascript-tutorials/javascript-functions) **:** স্ট্রিং থেকে নির্দিষ্ট একটা অক্ষর নেয়া যায়। index প্যারামিটারে যে সংখ্যা দিবেন সেই সংখ্যা যে অক্ষরের ইনডেক্স সেটা রিটার্ন করবে। ০ থেকে প্রথম অক্ষরের ইনডেক্স শুরু।

[**indexOf(string, start)**](http://www.webcoachbd.com/javascript-tutorials/javascript-indexof) **:** স্ট্রিং খোজার জন্য। string প্যারামিটারে যে শব্দ বা স্ট্রিংটির ইনডেক্স দিতে চান সেটা দিলে পুরো স্ট্রিংয়ে ঐ শব্দটির অবস্থান কত নম্বর ইনডেক্সে সেটা বের করবে আর start প্যারামিটারটি ঐচ্ছিক। যদি এটা দেন (পূর্নসংখ্যা) তাহলে ঐ সংখ্যা যার ইনডেক্স হবে সেখান থেকে খোজা শুরু করবে। যেমন ৫ দিলে ৫ নম্বর ইনডেক্সধারী এলিমেন্ট থেকে খোজা শুরু হবে।

[**lastIndexOf(string, start)**](http://www.webcoachbd.com/javascript-tutorials/javascript-indexof) **:** পিছন দিক থেকে স্ট্রিং খোজার জন্য। ধরুন একটা স্ট্রিংয়ে ১০ টি "e" আছে এখন আপনি চান সর্বশেষ "e" টা কত নম্বর ইনডেক্সে আছে সেটা দেখবেন তখন এই মেথড ব্যবহার করতে পারেন। string প্যারামিটারে যেটা খুজবেন সেটা এবং start (এটা না দিলে ০ ইনডেক্সধারী এলিমেন্ট থেকে খোজা শুরু করবে) প্যারামিটারে কত নম্বর ইনডেক্স থেকে খোজা শুরু করবেন সেটা।

[**split(delimiter, limit)**](http://www.webcoachbd.com/javascript-tutorials/javascript-string-split) **:** স্ট্রিং থেকে অ্যারে বানানোর জন্য। dilimiter হচ্ছে পুরো স্ট্রিংয়ের শব্দগুলি কি দিয়ে জুড়ানো যেমন কমা (,) দিয়ে হতে পারে, সাধারনত স্পেস দিয়ে থাকে। তো dilimiter এ জায়গায় হতে পারে ' ' আর limit (ঐচ্ছিক) দিয়ে ঠিক করা যায় যে কতগুলি শব্দ নিবেন।

[**substr(start, length)**](http://www.webcoachbd.com/javascript-tutorials/javascript-functions) **:** স্ট্রিং থেকে একটা অংশ/সাবস্ট্রিং বের করার জন্য। start হল কত নম্বর স্ট্রিং পজিশন থেকে খোজা শুরু করবে আর length হচ্ছে কয়টি ক্যারেক্টার/অক্ষর নিবে।

[**substring(start, end)**](http://www.webcoachbd.com/javascript-tutorials/javascript-functions) **:** স্ট্রিং থেকে একটা অংশ/সাবস্ট্রিং বের করার জন্য। substr() এর সাথে পার্থক্য হচ্ছে ২য় প্যারামিটার নিয়ে। এই মেথড start (পূর্নসংখ্যা) থেকে শুরু করে end (পূর্নসংখ্যা) এর আগ পর্যন্ত ক্যারেক্টার নিয়ে সেটা রিটার্ন করবে।

**slice(start, end) :** স্ট্রিং থেকে সাবস্ট্রিং বের করার জন্য। start এ দিতে হবে পূর্নসংখ্যা যেটা দিয়ে ঠিক হবে কোথা থেকে শুরু হবে আর end দিতে হবে আরেকটি পূর্ন সংখ্যা কোথা পর্যন্ত নিবে।

[**replace(regex, replacetext)**](http://www.webcoachbd.com/javascript-tutorials/javascript-replace) **:** স্ট্রিং এর কোন অংশ রিপ্লেস বা প্রতিস্থাপন করার জন্য। কোন অংশ রিপ্লেস করবেন সেটার রেগুলার একসপ্রেশন regex প্যারামিটারে আর replacetext এ দিবেন যে স্ট্রিংয়ে রিপ্লেসমেন্ট করতে চান।

[**search(regex)**](http://www.webcoachbd.com/javascript-tutorials/javascript-stringsearch) **:** স্ট্রিংয়ে খোজ করার জন্য। কোন অংশ খুজবেন সেটা ঠিক করতে হবে regex তথা রেগুলার এক্সপ্রেশন দিয়ে। খুজে পেলে পজিশন রিটার্ন করবে (প্রথমটির) না পেলে -1 রিটার্ন করে। regex এর জায়াগায় কোন নির্দিষ্ট স্ট্রিং দিতে পারেন।

[**match(regex)**](http://www.webcoachbd.com/javascript-tutorials/javascript-stringsearch) **:** এটাও স্ট্রিং খোজার জন্য তবে এখানে শুধু রেগুলার এক্সপ্রেশন দেয়া হয়। যদি মিল পায় তাহলে মানগুলি অ্যারে অবজেক্ট আকারে রিটার্ন করবে আর না পেলে null রিটার্ন করবে।

# জাভাস্ক্রিপ্ট স্ট্রিং মেথড

[« পূর্ববর্তী](https://www.sattacademy.com/js/js_strings.php)

[পরবর্তী »](https://www.sattacademy.com/js/js_arrays.php)

স্ট্রিং মেথড আপনাকে স্ট্রিং নিয়ে কাজ করতে সাহায্য করবে।

## স্ট্রিং মেথড এবং প্রোপার্টি

প্রিমিটিভ(Primitive) ভ্যালুর, যেমন "Tamjid Hasan", কোন প্রোপার্টি বা মেথড থাকতে পারে না(কারন তারা অবজেক্ট নয়)।

## স্ট্রিং লেন্থ

**length** প্রোপার্টি স্ট্রিংটি কয়টি ক্যারেক্টার নিয়ে গঠিত তা রিটার্ন করেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<p id="test"></p>

<script>

var text = "WelcomeToSattAcademy";

document.getElementById("test").innerHTML = text.length;

</script>

</body>

</html>

Copy

### ফলাফল

## স্ট্রিংয়ের মধ্যে স্ট্রিং খোঁজা

**indexOf()** মেথডটি একটি স্ট্রিংয়ের মধ্যে একটি নির্দিষ্ট টেক্সটের **প্রথম** অবস্থান রিটার্ন করেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4 id="para">Please locate where 'locate' occurs!</h4>

<button onclick="myFunc()"> নিজে চেষ্টা করি</button>

<p id="test"></p>

<script>

function myFunc() {

var str = document.getElementById("para").innerHTML;

var position = str.indexOf("locate");

document.getElementById("test").innerHTML = position;

}

</script>

</body>

</html>

Copy

### ফলাফল

**lastIndexOf()** মেথডটি একটি স্ট্রিংয়ের মধ্যে একটি নির্দিষ্ট টেক্সটের **শেষ** অবস্থান রিটার্ন করেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4 id="para">Please locate where 'locate' occurs!</h4>

<button onclick="myFunc()"> নিজে চেষ্টা করি</button>

<p id="test"></p>

<script>

function myFunc() {

var str = document.getElementById("para").innerHTML;

var position = str.lastIndexOf("locate");

document.getElementById("test").innerHTML = position;

}

</script>

</body>

</html>

Copy

### ফলাফল

যদি টেক্সটটি না পাওয়া যায় indexOf() এবং lastIndexOf() উভয় মেথডই -1 রিটার্ন করে।

জাভাস্ক্রিপ্টে স্ট্রিংয়ের অবস্থান শূন্য থেকে গণনা করা হয়।  
স্ট্রিং-এ ০(শূন্য) হচ্ছে প্রথম অবস্থান, ১ হচ্ছে দ্বিতীয় অবস্থান আর ২ হচ্ছে তৃতীয় অবস্থান এবং এইভাবে চলতে থাকে।

উভয় মেথডই দ্বিতীয় একটি প্যারামিটার গ্রহণ করে।

## স্ট্রিংয়ের মধ্যে একটি স্ট্রিং অনুসন্ধান

**search()** মেথডটি একটি স্ট্রিংয়ের মধ্যে একটি নির্দিষ্ট ভ্যালু অনুসন্ধান করে এবং অনুসন্ধানকৃত ভ্যালুর অবস্থান রিটার্ন করেনঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4 id="para">Please locate where 'locate' occurs!</h4>

<button onclick="myFunc()"> নিজে চেষ্টা করি</button>

<p id="test"></p>

<script>

function myFunc() {

var str = document.getElementById("para").innerHTML;

var position = str.search("locate");

document.getElementById("test").innerHTML = position;

}

</script>

</body>

</html>

Copy

### ফলাফল

## আপনি লক্ষ্য করেছিলেন কি?

indexOf() এবং search() উভয় মেথড একই।

তারা উভয়ই একই আর্গুমেন্ট(arguments) বা প্যারামিটার ভ্যালু গ্রহন করে এবং একই মান রিটার্ন করে।

দুটি মেথড একই, কিন্তু search() মেথডটি অনেক বেশি শক্তিশালী।

আপনি অনুসন্ধান সম্পর্কে রেগুলার এক্সপ্রেশন(regular expressions) পরিচ্ছেদে আরো জানবেন।

## স্ট্রিংকে বিভক্ত করা

স্ট্রিংয়ের একটি অংশকে বিভক্ত করার জন্য ৩টি মেথড রয়েছেঃ

* slice(start, end)
* substring(start, end)
* substr(start, length)

## slice() মেথড

**slice()** মেথডটি স্ট্রিংয়ের একটি অংশকে ভাগ করে এবং বিভক্ত অংশটিকে নতুন একটি স্ট্রিংয়ে রিটার্ন করে।

মেথডটি দুইটি প্যারামিটার গ্রহন করেঃ শুরুর অবস্থান(start index) এবং শেষের অবস্থান(end index)।

এই উদাহরণে একটি স্ট্রিংয়ের ৭ থেকে ১৩ অবস্থানের অংশটুকু ভাগ করে নেয়ঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4> slice() মেথড স্ট্রিং-এর একটি অংশকে এক্সট্রাক্ট করে এবং এক্সট্রাক্ট করা অংশকে নতুন স্ট্রিং-এ রিটার্ন করে। </h4>

<p id="test"></p>

<script>

var animals = "Tiger, Lion, Elephant";

document.getElementById("test").innerHTML = animals.slice(7,11);

</script>

</body>

</html>

Copy

### ফলাফল

যদি কোন প্যারামিটারের মান ঋনাত্মক হয় তবে স্ট্রিংয়ের অবস্থান গণনা শেষ থেকে শুরু হবে।

এই উদাহরণে স্ট্রিংয়ের -১২ থেকে -৬ অংশটি ভাগ করা হবে। যেহেতু প্যারামিটারের মান ঋনাত্মক এটি স্ট্রিংয়ের শেষ থেকে গণনা শুরু করবে।

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4>slice() মেথড স্ট্রিং-এর একটি অংশকে এক্সট্রাক্ট করে এবং এক্সট্রাক্ট করা অংশকে নতুন স্ট্রিং-এ রিটার্ন করে। </h4>

<p id="test"></p>

<script>

var animals = "Tiger, Lion, Elephant";

document.getElementById("test").innerHTML = animals.slice(-15, -10);

</script>

</body>

</html>

Copy

### ফলাফল

আপনি যদি ২য় প্যারামিটারটি উল্লেখ না করেন তাহলে এটি প্রথম অবস্থান থেকে বাকী সম্পূর্ন অংশ ভাগ করবেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4> slice() মেথড স্ট্রিং-এর একটি অংশকে এক্সট্রাক্ট করে এবং এক্সট্রাক্ট করা অংশকে নতুন স্ট্রিং-এ রিটার্ন করে। </h4>

<p id="test"></p>

<script>

var animals = "Tiger, Lion, Elephant";

document.getElementById("test").innerHTML = animals.slice(13);

</script>

</body>

</html>

Copy

### ফলাফল

অথবা এটি শেষ থেকে গণনা করবেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4> slice() মেথড স্ট্রিং-এর একটি অংশকে এক্সট্রাক্ট করে এবং এক্সট্রাক্ট করা অংশকে নতুন স্ট্রিং-এ রিটার্ন করে। </h4>

<p id="test"></p>

<script>

var animals = "Tiger, Lion, Elephant";

document.getElementById("test").innerHTML = animals.slice(-15);

</script>

</body>

</html>

Copy

### ফলাফল

ইন্টারনেট এক্সপ্লোরার ৮ এবং এর আগের ব্রাউজারগুলোতে প্যারামিটারের ঋনাত্মক মান কাজ করে না।

## substring() মেথড

**substring()** মেথডটি slice() মেথডের মতই।

পার্থক্য হচ্ছে substring() মেথড ঋনাত্মক ইন্ডেক্স ভ্যালু গ্রহণ করে না।

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4> substring() মেথড স্ট্রিং-এর একটি অংশকে এক্সট্রাক্ট করে এবং এক্সট্রাক্ট করা অংশকে নতুন স্ট্রিং-এ রিটার্ন করে। </h4>

<p id="test"></p>

<script>

var animals = "Tiger, Lion, Elephant";

document.getElementById("test").innerHTML = animals.substring(7,11);

</script>

</body>

</html>

Copy

### ফলাফল

আপনি যদি ২য় প্যারামিটারটি উল্লেখ না করেন তাহলে substring() মেথড প্রথম অবস্থান থেকে বাকী সম্পূর্ন অংশ ভাগ করবে।

## substr() মেথড

**substr()** মেথডটি slice() মেথডের মতই।

পার্থক্য হচ্ছে দ্বিতীয় প্যারামিটারটিতে কতটি ক্যারেক্টার ভাগ করতে চান তা উল্লেখ করতে হবে।

<

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4> substr() মেথড স্ট্রিং-এর একটি অংশকে এক্সট্রাক্ট করে এবং এক্সট্রাক্ট করা অংশকে নতুন স্ট্রিং-এ রিটার্ন করে। </h4>

<p id="test"></p>

<script>

var animals = "Tiger, Lion, Elephant";

document.getElementById("test").innerHTML = animals.substr(7,4);

</script>

</body>

</html>

Copy

### ফলাফল

যদি প্রথম প্যারামিটারের মান ঋনাত্মক হয় তাহলে স্ট্রিংয়ের শেষ প্রান্ত থেকে গণনা শুরু হয়।

২য় প্যারামিটারের মান কথনোই ঋনাত্মক হতে পারবে না কারন এটি দৈর্ঘ্য নির্দেশ করে।

আপনি যদি ২য় প্যারামিটারটি উল্লেখ না করেন তাহলে substr() মেথড বাকী সম্পূর্ন অংশ ভাগ করবে।

## স্ট্রিংয়ের কন্টেন্ট প্রতিস্থাপন

**replace()** মেথডটি স্ট্রিংয়ের নির্দিষ্ট একটি মানকে অন্য একটি ভ্যালু দ্বারা প্রতিস্থাপন করেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4> নিচের প্যারাগ্রাফে "Codeacademy" বদলে "sattacademy" প্রতিস্থাপন করুন </h4>

<button onclick="myFunc()">এটি নিজে চেষ্টা করি </button>

<p id="test"> দয়া করে Codeacademy পরিদর্শন করুন!</p>

<script>

function myFunc() {

var str = document.getElementById("test").innerHTML;

var text = str.replace("Codeacademy","sattacademy");

document.getElementById("test").innerHTML = text;

}

</script>

</body>

</html>

Copy

### ফলাফল

replace() মেথড অনুসন্ধান ভ্যালু হিসাবে রেগুলার এক্সপ্রেশনও নিতে পারে।

স্বাভাবিকভাবে replace() মেথডটি অনুসন্ধানকৃত অংশকে প্রথম যেখানে পাবে তাকেই প্রতিস্থাপন করবে। যদি সবগুলোকে প্রতিস্থাপন করতে চান সেক্ষেত্রে রেগুলার এক্সপ্রেশনের g ফ্ল্যাগ(গ্লোবাল সার্চের জন্য) ব্যবহার করতে হবেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4> নিচের প্যারাগ্রাফে "Codeacademy" বদলে "sattacademy" প্রতিস্থাপন করুন </h4>

<button onclick="myFunc()">এটি নিজে চেষ্টা করি </button>

<p id="test"> দয়া করে Codeacademy এবং Codeacademy পরিদর্শন করুন!</p>

<script>

function myFunc() {

var str = document.getElementById("test").innerHTML;

var text = str.replace(/Codeacademy/g,"sattacademy");

document.getElementById("test").innerHTML = text;

}

</script>

</body>

</html>

Copy

### ফলাফল

replace() মেথডটি স্ট্রিংকে পরিবর্তন করে না।এটা নতুন একটি স্ট্রিং রিটার্ন করে।

## বড়হাতের অক্ষর(Upper Case) এবং ছোটহাতের অক্ষরে(Lower Case) রূপান্তর

একটি স্ট্রিংকে **toUpperCase()** মেথডের মাধ্যমে বড় হাতের অক্ষরে রূপান্তর করা হয়েছেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4>স্ট্রিং-কে বড় হাতের অক্ষরে রুপান্তরঃ </h4>

<button onclick="myFunc()">নিজে চেষ্টা করি</button>

<p id="test">Welcome To Satt Academy!</p>

<script>

function myFunc() {

var text = document.getElementById("test").innerHTML;

document.getElementById("test").innerHTML = text.toUpperCase();

}

</script>

</body>

</html>

Copy

### ফলাফল

একটি স্ট্রিংকে **toLowerCase()** মেথডের মাধ্যমে ছোটহাতের অক্ষরে রূপান্তর করা হয়েছেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4>স্ট্রিং-কে ছোট হাতের অক্ষরে রুপান্তরঃ </h4>

<button onclick="myFunc()">নিজে চেষ্টা করি</button>

<p id="test">Welcome To Satt Academy!</p>

<script>

function myFunc() {

var text = document.getElementById("test").innerHTML;

document.getElementById("test").innerHTML = text.toLowerCase();

}

</script>

</body>

</html>

Copy

### ফলাফল

## concat() মেথড

স্ট্রিং যোগ করার জন্য **concat()** মেথড ব্যবহার করা হয়ঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4>concat() মেথড দুই বা তার বেশি স্ট্রিং-কে সংযুক্ত করে </h4>

<p id="test"></p>

<script>

var word1 = "Satt";

var word2 = "Academy!";

document.getElementById("test").innerHTML = word1.concat(" ",word2);

</script>

</body>

</html>

Copy

### ফলাফল

**concat()** মেথডটি প্লাস(+) অপারেটরের পরিবর্তে ব্যবহার করা যেতে পারে। দুইটি লাইন একই কাজ করেঃ

var word = "Satt" + " " + "Academy!";

var word = "Satt".concat(" ","Academy!");

Copy

সকল স্ট্রিং মেথডই একটি নতুন স্ট্রিং রিটার্ন করে। তারা মূল স্ট্রিংকে পরিবর্তন করে না।  
স্ট্রিং অপরিবর্তনীয়ঃ স্ট্রিংগুলো পরিবর্তন করা যাবে না শুধুমাত্র প্রতিস্থাপন করা যায়।

## স্ট্রিংয়ের ক্যারেক্টার ভাগ

স্ট্রিং ক্যারেক্টারকে ভাগ করার দুইটি পদ্ধতি রয়েছেঃ

* charAt(position)
* charCodeAt(position)

## charAt() মেথড

**charAt()** মেথডটি একটি স্ট্রিংয়ের একটি উল্লেখিত অবস্থানের(index) ক্যারেক্টারকে রিটার্ন করেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4> charAt () মেথড স্ট্রিং-এর দেওয়া কারেক্টার অনুযায়ী অবস্থান রিটার্ন করেঃ </h4>

<p id="test"></p>

<script>

var sentence = "SATT ACADEMY";

document.getElementById("test").innerHTML = sentence.charAt(0);

</script>

</body>

</html>

Copy

### ফলাফল

## charCodeAt() মেথড

**charCodeAt()** মেথডটি একটি স্ট্রিংয়ের একটি উল্লেখিত অবস্থানের(index) ক্যারেক্টারের ইউনিকোড রিটার্ন করেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4> charCodeAt () মেথড স্ট্রিং-এর দেওয়া কারেক্টারের ইউনিকোড অনুযায়ী অবস্থান রিটার্ন করেঃ </h4>

<p id="test"></p>

<script>

var sentence = "SATT ACADEMY";

document.getElementById("test").innerHTML = sentence.charCodeAt(1);

</script>

</body>

</html>

Copy

### ফলাফল

## স্ট্রিংকে অ্যারের মত করে এক্সেস করা নিরাপদ নয়

আপনি হয়তো স্ট্রিংকে অ্যারের মত করে এক্সেস করা দেখতে পারেনঃ

var sentence = "SATT ACADEMY";

sentence[1]; // A রিটার্ন করে

Copy

ইহা বিপজ্জনক এবং অনিশ্চিত ফলাফল দেখাতে পারেঃ

* এটা সকল ব্রাউজারে কাজ করে না(IE5, IE6, IE7 এ কাজ করে না)
* এটা স্ট্রিংকে অ্যারের মত করে তোলে(কিন্তু তারা অ্যারে না)
* str[0] = "H" কোন ভুল দেখায় না(কিন্তু কাজও করে না)

যদি আপনি একটি স্ট্রিংকে অ্যারে আকারে পড়তে চান, প্রথমে একে অ্যারেতে রূপান্তর করুন।

## স্ট্রিংকে অ্যারেতে রূপান্তর

**split()** মেথড ব্যবহার করে একটি স্ট্রিংকে অ্যারেতে রূপান্তরিত করা যেতে পারেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4>প্রথম অ্যারে এলিমেন্ট প্রদর্শনের জন্য "নিজে চেষ্টা করি" ক্লিক করুনঃ </h4>

<button onclick="myFunc()">নিজে চেষ্টা করি</button>

<p id="test"></p>

<script>

function myFunc() {

var char = "m,s,f,e,h,k";

var arr = char.split(",");

document.getElementById("test").innerHTML = arr[0];

}

</script>

</body>

</html>

Copy

### ফলাফল

যদি বিভাজক বাদ দেওয়া হয়, তাহলে সম্পূর্ন স্ট্রিংটি অ্যারের ০ ইনডেক্সে রাখা হবে।

বিভাজক যদি "" হয়, তবে অ্যারের মধ্যে প্রতিটি ক্যারেক্টারকে ভিন্ন ইনডেক্সে রাখা হবেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4>প্রথম দুটি অ্যারে এলিমেন্ট প্রদর্শনের জন্য "নিজে চেষ্টা করি" ক্লিক করুনঃ </h4>

<button onclick="myFunc()">নিজে চেষ্টা করি</button>

<p id="test"></p>

<script>

function myFunc() {

var word = "Satt";

var arr = word.split("");

document.getElementById("test").innerHTML = arr[0] + "<br>" + arr[1];

}

</script>

</body>

</html>

Copy

### ফলাফল

## সম্পূর্ণ স্ট্রিং রেফারেন্স

সম্পূর্ণ রেফারেন্স জন্য আমাদের [সম্পূর্ণ জাভাস্ক্রিপ্ট স্ট্রিং রেফারেন্সে](https://www.sattacademy.com/jsref/jsref_obj_string.php) যান।

রেফারেন্সে সকল স্ট্রিং প্রোপার্টি এবং মেথডের উদাহরণ রয়েছে।