**জাভাস্ক্রিপ্ট এইচটিএমএল কন্টেন্ট পরিবর্তন করতে পারে**

জাভাস্ক্রিপ্টে বহুবিধ মেথড রয়েছে তন্মধ্যে **getElementById()** মেথডটি অন্যতম।

এই উদাহরনে **getElementById()** মেথডটি id="test" যুক্ত এইচটিএমএল এলিমেন্টকে খুঁজে বের করার জন্য ব্যবহার করা হয়েছে এবং এলিমেন্টের কন্টেন্ট "হ্যালো জাভাস্ক্রিপ্ট" দ্বারা পরিবর্তন করা হয়েছে।

**উদাহরণ**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h3>জাভাস্ক্রিপ্ট উদাহরণ</h3>

<p id="test">জাভাস্ক্রিপ্ট এইচটিএলএল কন্টেন্ট পরিবর্তন করতে পারে।</p>

<button type="button"

onclick="document.getElementById('test').innerHTML = 'হ্যালো জাভাস্ক্রিপ্ট!'">ক্লিক করুন</button>

</body>

</html>

Copy

**ফলাফল**

### জাভাস্ক্রিপ্ট উদাহরণ

হ্যালো জাভাস্ক্রিপ্ট!

**জাভাস্ক্রিপ্ট এইচটিএমএল এট্রিবিউট পরিবর্তন করতে পারে**

এই উদাহরণটিতে **<input>** ট্যাগের **type** এট্রিবিউট পরিবর্তন করা হয়েছেঃ

**উদাহরণ**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h3>জাভাস্ক্রিপ্ট উদাহরণ</h3>

<button onclick="document.getElementById('test').type='text">পাসওয়ার্ড শো</button><br><br>

<input id="test" type="password" value="123456"><br><br>

<button onclick="document.getElementById('test').type='password'">পাসওয়ার্ড হাইড</button>

</body>

</html>

Copy

**ফলাফল**

**জাভাস্ক্রিপ্ট এইচটিএমএল স্টাইল পরিবর্তন করতে পারে**

এইচটিএমএল এলিমেন্টের এট্রিবিউট পরিবর্তন করা থেকে এইচটিএমএল স্টাইল পরিবর্তন করা একটু ভিন্নঃ

**উদাহরণ**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h3>জাভাস্ক্রিপ্ট উদাহরণ</h3>

<p id="test">জাভাস্ক্রিপ্ট এইচটিএমএল এলিমেন্টের স্টাইল পরিবর্তন করতে পারে।

<button type="button" onclick="document.getElementById('test').style.color='green'">ক্লিক করি!</button>

</body>

</html>

# জাভাস্ক্রিপ্ট কোথায় রাখবেন?

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

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

এইচটিএমএল পেজের মধ্যে <head> অথবা <body> অথবা উভয় ট্যাগ এর মধ্যে জাভাস্ক্রিপ্টকে রাখা যায়।

<script> ট্যাগ এর মধ্যে জাভাস্ক্রিপ্ট কোড লিখতে হয়।

## ****<script>**** ট্যাগ

আপনি যদি এইচটিএমএল এর মধ্যে জাভাস্ক্রিপ্ট কোড ব্যবহার করতে চান তাহলে জাভাস্ক্রিপ্ট কোডকে অবশ্যই <script> এবং </script> ট্যাগের মধ্যে লিখতে হবে।

**উদাহরণ**

<script>

document.getElementById("test").innerHTML = "আমার প্রথম জাভাস্ক্রিপ্ট";

</script>

Copy

জাভাস্ক্রিপ্ট এর পুরাতন ভার্সন সমূহে type এট্রিবিউট ব্যবহার করা হত। যেমনঃ <script type="text/javascript">।  
কিন্তু বর্তমানে আর এটার প্রয়োজন নাই। কারণ জাভাস্ক্রিপ্ট এইচটিএমএলের পূর্বনির্ধারিত স্ক্রিপ্টিং ল্যাংগুয়েজ।

## জাভাস্ক্রিপ্ট ফাংশন এবং ইভেন্ট

জাভাস্ক্রিপ্ট **ফাংশন** হচ্ছে একগুচ্ছ জাভাস্ক্রিপ্ট কোড যাকে আপনি **"কল"** করলে সম্পাদিত হবে।

উদাহরণস্বরূপঃ কোন **ইভেন্ট**(ঘটনা) ঘটলেই একটি ফাংশন সম্পাদিত হবে। যেমনঃ ব্যবহারকারী যদি মাউস ক্লিক করে।

পরবর্তীতে [ফাংশন](https://www.sattacademy.com/js/js_functions.php) এবং [ইভেন্ট](https://www.sattacademy.com/js/js_events.php) সম্পর্কে বিস্তারিত আলোচনা করা হয়েছে।

## ****<head>**** অথবা ****<body>**** এর মধ্যে জাভাস্ক্রিপ্ট

এইচটিএমএল ডকুমেন্টের মধ্যে আপনি যত খুশি জাভাস্ক্রিপ্ট কোড লিখতে/রাখতে পারবেন।

জাভাস্ক্রিপ্টকে এইচটিএমএলের <body> অথবা <head> অথবা উভয়ের মধ্যেই রাখা যেতে পারে।

|  |  |
| --- | --- |
| **Note** | সবচেয়ে ভাল অভ্যাস হচ্ছে সকল কোড একই স্থানে রাখা। |

## ****<head>****এলিমেন্টে জাভাস্ক্রিপ্ট

এই উদাহরণে জাভাস্ক্রিপ্ট ফাংশনকে এইচটিএমএল পেজের **<head>** অংশে রাখা হয়েছে।

বাটনে ক্লিক করলে ফাংশনটি আপনার ডাকে সাড়া দিবেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="bn">

<head>

<meta charset="utf-8">

<title> Head সেকশনে জাভাস্ক্রিপ্ট এর ব্যবহার</title>

<script>

function myFunc() {

document.getElementById("test").innerHTML = "প্যারাগ্রাফটি পরিবর্তিত হয়েছে ";

}

</script>

</head>

<body>

<h4>হেড সেকশনে জাভাস্ক্রিপ্ট কোড লেখা হয়েছে</h4>

<p id="test">সাধারণ প্যারাগ্রাফ </p>

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

</body>

</html>

Copy

### ফলাফল

## <body> এর মধ্যে জাভাস্ক্রিপ্ট

এই উদাহরণে জাভাস্ক্রিপ্ট ফাংশনকে এইচটিএমএল পেজের **<body>** এলিমেন্টে রাখা হয়েছে

বাটনে ক্লিক করলে ফাংশনটি আপনার ডাকে সাড়া দিবেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="bn">

<head>

<meta charset="utf-8">

<title> Head সেকশনে জাভাস্ক্রিপ্ট এর ব্যবহার</title>

</head>

<body>

<p id="test">সাধারণ প্যারাগ্রাফ </p>

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

<script>

function myFunc() {

document.getElementById("test").innerHTML = "প্যারাগ্রাফটি পরিবর্তিত হয়েছে ";

}

</script>

</body>

</html>

Copy

### ফলাফল

|  |  |
| --- | --- |
| **Note** | জাভাস্ক্রিপ্টকে <body> এলিমেন্টের নিচের দিকে রাখা সবচেয়ে ভাল অভ্যাস। এতে আপনার পেজ দ্রুত লোড হয়। কারণ জাভাস্ক্রিপ্ট কম্পাইলার আপনার পেজকে ধীরগতি সম্পন্ন করে দিতে পারে। |

## বাহ্যিক জাভাস্ক্রিপ্ট

জাভাস্ক্রিপ্টকে বাহ্যিক(external) ফাইলের মধ্যেও রাখা যায়ঃ

### script.js

function myFunc() {

document.getElementById("test"

একই ধরণের স্ক্রিপ্ট কোড যখন একাধিক ওয়েব পেজে ব্যবহার করার প্রয়োজন হয় তখন এক্সটার্নাল জাভাস্ক্রিপ্ট ব্যবহার করতে হয়।

জাভাস্ক্রিপ্ট ফাইলের ফাইল এক্সটেনশন হচ্ছে **.js**।

এক্সটার্নাল জাভাস্ক্রিপ্ট ব্যবহার করার জন্য জাভাস্ক্রিপ্ট ফাইলটি <script> ট্যাগের src এট্রিবিউটের মধ্যে রাখুনঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="bn">

<head>

<meta charset="utf-8">

<title>বাহ্যিক জাভাস্ক্রিপ্ট এর উদাহরণ</title>

</head>

<body>

<h3>এক্সটার্নাল জাভাস্ক্রিপ্ট</h3>

<p id="test">সাধারণ প্যারাগ্রাফ</p>

<button type="button" onclick="myFunc()">ক্লিক করি</button>

<p><strong>নোট:</strong> "Script.js" এক্সটার্নাল ফাইলের মধ্যে myFunc জমা আছে।</p>

<script src="Script.js"></script>

</body>

</html>

Copy

### ফলাফল

এক্সটার্নাল জাভাস্ক্রিপ্ট রেফারেন্স/ফাইলকে আপনি আপনার ইচ্ছামত <head> অথবা <body> এর মধ্যে রাখতে পারেন।

রেফারেন্স কোডগুলো এমন আচরণ করে যেন এগুলোকে <script> ট্যাগের মধ্যেই রাখা হয়েছে।

|  |  |
| --- | --- |
| **Note** | এক্সটার্নাল স্ক্রিপ্টের মধ্যে <script> ট্যাগ ব্যবহার করা যাবে না। |

## এক্সটার্নাল জাভাস্ক্রিপ্টের সুবিধা

জাভাস্ক্রিপ্টকে এক্সটার্নাল ফাইলের মধ্যে রাখলে কিছু বিশেষ বিশেষ সুবিধা পাওয়া যায়ঃ

* এটা এইচটিএমএল এবং জাভাস্ক্রিপ্ট কোডকে পৃথক করে।
* এইচটিএমএল এবং জাভাস্ক্রিপ্ট কোড অধিক পাঠযোগ্য হয় এবং খুব সহজেই রক্ষণাবেক্ষণ করা যায়।
* ক্যাশড(cached) জাভাস্ক্রিপ্ট ফাইল পেজ লোডের গতি বাড়িয়ে দেয়।

## জাভাস্ক্রিপ্ট প্রোগ্রাম

একটি **কম্পিউটার প্রোগ্রাম** একগুচ্ছ "ইন্সট্রাকশন/নির্দেশনার" তালিকা যা কম্পিউটারের মাধ্যমে সম্পাদিত হয়।

প্রোগ্রামিং-এ এই নির্দেশনাগুলোকে **স্টেটমেন্ট** বলা হয়।

জাভাস্ক্রিপ্ট একটি **প্রোগ্রামিং ল্যাংগুয়েজ**।

**সেমিকোলন**(;) এর মাধ্যমে জাভাস্ক্রিপ্ট স্টেটমেন্টকে আলাদা করা হয়।

### উদাহরণ

<!DOCTYPE html>

<html lang="bn">

<head>

<meta charset="utf-8">

<title>জাভাস্ক্রিপ্ট উদাহরণ</title>

</head>

<body>

<h4>জাভাক্রিপ্ট স্টেটমেন্টগুলো সেমিকোলন দ্বারা আলাদা করা হয়।</h4>

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

<script>

a = 1;

b = 2;

c = a + b;

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

</script>

</body>

</html>

Copy

### ফলাফল

|  |  |
| --- | --- |
| **Note** | এইচটিএমএলে জাভাস্ক্রিপ্ট প্রোগ্রামগুলো ব্রাউজারের মাধ্যমে সম্পাদিত হয়। |

## জাভাস্ক্রিপ্ট স্টেটমেন্ট

জাভাস্ক্রিপ্ট স্টেটমেন্ট সাধারণত ভ্যালু, অপারেটর, এক্সপ্রেশন, কিওয়ার্ড এবং কমেন্ট নিয়ে তৈরি হয়।

## জাভাস্ক্রিপ্ট ভ্যালু

জাভাস্ক্রিপ্টে দুই ধরণের ভ্যালু ব্যবহার করা হয়। যথা- ফিক্সড ভ্যালু এবং ভ্যারিয়েবল ভ্যালু।

ফিক্সড ভ্যালুকে **লিটারাল** বলা হয়। ভ্যারিয়েবল ভ্যালুকে **চলক বা ভ্যারিয়েবল** বলা হয়।

## জাভাস্ক্রিপ্ট লিটারাল

ফিক্সড ভ্যালু লেখার সবচেয়ে গুরুত্বপূর্ণ নিয়ম হলোঃ

সংখ্যাকে পূর্ণ সংখ্যা অথবা দশমিকে লেখাঃ

var a = 105.24; // দশমিকসহ নম্বর

var b = 105; // দশমিক ছাড়া নম্বর

Copy

**স্ট্রিং** হচ্ছে টেক্সট যাকে সিঙ্গেল অথবা ডাবল কোটেশনের(উদ্ধৃতি চিহ্ন) মধ্যে লেখা হয়ঃ

var a = "Tamim";

var b = 'Tamim';

Copy

## জাভাস্ক্রিপ্ট ভ্যারিয়েবল

প্রোগ্রামিং ভাষায় **ভ্যালু** সংরক্ষন করার কাজে **ভ্যারিয়েবল** ব্যবহার করা হয়।

ভ্যারিয়েবল ডিক্লেয়ার করার সময় জাভাস্ক্রিপ্টে **var** কিওয়ার্ড ব্যবহার করা হয়।

ভ্যারিয়েবলের মধ্যে মান রাখার জন্য **সমান(=)** চিহ্ন ব্যবহার করা হয়।

নিচের উদাহরণে a কে প্রথমে **var** কিওয়ার্ডের মাধ্যমে ভ্যারিয়েবল হিসেবে ডিক্লেয়ার করা হয়, তারপর সমান চিহ্ন ব্যবহার করে এর মধ্যে 10 কে ভ্যালু হিসেবে রাখা হয়েছে।

### উদাহরণ

<!DOCTYPE html>

<html lang="bn">

<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>

<h3>জাভাস্ক্রিপ্ট চলক</h3>

<h5>এই উদাহরনটিতে x কে চলক হিসেবে ধরা হয়েছে, পরে x এর মান ৬ নির্ধারন করা হয়েছে।</h5>

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

<script>

var a;

a = 10;

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

</script>

</body>

</html>

Copy

### ফলাফল

## জাভাস্ক্রিপ্ট অপারেটর

ভ্যারিয়েবলের মধ্যে মান রাখার জন্য জাভাস্ক্রিপ্টে সাধারণত **এসাইনমেন্ট অপারেটর( = )** ব্যবহার করা হয়ঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="bn">

<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>

<h3>= অপারেটর</h3>

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

<script>

var a = 32;

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

</script>

</body>

</html>

Copy

### ফলাফল

মান নির্ণয় করার জন্য জাভাস্ক্রিপ্টে সাধারণত **গাণিতিক অপারেটর( + - \*  / )** ব্যবহার করা হয়ঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="bn">

<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>যোগ করার আগে গুন করার উপর জোর দেওয়া হয়।</h4>

<h5>কিন্তু গুন করার আগে প্রথম বন্ধনীর কাজ করা হয়।</h5>

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

<script>

document.getElementById("test").innerHTML = (14 + 16) \* 3;

</script>

</body>

</html>

Copy

### ফলাফল

## জাভাস্ক্রিপ্ট এক্সপ্রেশন

কিছু ভ্যালু, চলক এবং অপারেটরের সমন্বয়ে জাভাস্ক্রিপ্ট এক্সপ্রেশন গঠিত হয় যা একটি ভ্যালু নির্ণয় করে।

উদাহরণস্বরূপ, ৫ \* ১০ এর গুণন ৫০ এ রূপান্তরিত হয়।

### উদাহরণ

<!DOCTYPE html>

<html lang="bn">

<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>

<h3>জাভাস্ক্রিপ্ট এক্সপ্রেশন</h3>

<h5>এক্সপ্রেশনের মাধ্যমে ভ্যালু নির্ণয়।</h5>

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

<script>

document.getElementById("test").innerHTML = 12 + 13;

</script>

</body>

</html>

Copy

### ফলাফল

অনেক সময় এক্সপ্রেশনের মধ্যে ভ্যালু হিসেবে ভ্যারিয়েবলও থাকতে পারেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="bn">

<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>

<h3>জাভাস্ক্রিপ্ট এক্সপ্রেশন</h3>

<h5>এক্সপ্রেশনের মাধ্যমে ভ্যালু নির্ণয়।</h5>

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

<script>

var a = 12;

document.getElementById("test").innerHTML = a + 13;

</script>

</body>

</html>

Copy

### ফলাফল

জাভাস্ক্রিপ্টে মানগুলো বিভিন্ন টাইপের হতে পারে। যেমনঃ নাম্বার এবং স্ট্রিং।

উদাহরণস্বরূপ, "সাহীদ" + " " + "মাহমুদ", যোগ হয়ে "সাহীদ মাহমুদ" হয়ঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="bn">

<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>

<h3>জাভাস্ক্রিপ্ট এক্সপ্রেশন</h3>

<h5>এক্সপ্রেশনের মাধ্যমে ভ্যালু নির্ণয়।</h5>

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

<script>

document.getElementById("test").innerHTML = "সাহীদ" + " " + "মাহমুদ";

</script>

</body>

</html>

Copy

### ফলাফল

## জাভাস্ক্রিপ্ট কিওয়ার্ড

বিভিন্ন কার্যসম্পাদন করার জন্য জাভাস্ক্রিপ্ট **কিওয়ার্ড** ব্যবহার করা হয়।

যেমন- **var** কিওয়ার্ড ব্রাউজারকে **ভ্যারিয়েবল(variable)** তৈরি করতে বলেঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="bn">

<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>

<h3>var কীওর্য়াড ভেরিয়েবল তৈরী করে।</h1>

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

<script>

var a = 2 + 5;

var b = a \* 12;

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

</script>

</body>

</html>

Copy

### ফলাফল

## জাভাস্ক্রিপ্ট কমেন্ট

জাভাস্ক্রিপ্টের সকল স্টেটমেন্ট **সম্পাদিত** হয়না।

ডাবল স্ল্যাস**(//)**এর পরের অথবা **/\*** এবং **\*/** এর মাঝের কোডগুলোকে কমেন্ট হিসাবে গণ্য করা হয়।

কমেন্টের কোডগুলোকে উপেক্ষা করা হয় এবং এগুলো সম্পাদিত হবে নাঃ

### উদাহরণ

<!DOCTYPE html>

<html lang="bn">

<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>

<h3>কমেন্ট সম্পাদিত হয় না।</h3>

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

<script>

var a = 10;

// var a = 23; ইহা সম্পাদিত হবে না

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

</script>

</body>

</html>

Copy

### ফলাফল

## জাভাস্ক্রিপ্ট আইডেন্টিফায়ার

আইডেন্টিফায়ারসমূহ হচ্ছে নাম

জাভাস্ক্রিপ্টে ভ্যারিয়েবল,কীওয়ার্ড এবং ফাংশনের নাম দেওয়ার জন্য আইডেন্টিফায়ার ব্যবহার করা হয়।

অধিকাংশ প্রোগ্রামেই নামকরনের নিয়ম একই থাকে।

জাভাস্ক্রিপ্টে প্রথম ক্যারেক্টারটি অবশ্যই অক্ষর, আন্ডারস্কোর(\_) অথবা ডলার($) চিহ্ন হবে।

পরের ক্যারেক্টারগুলো অক্ষর(characters), সংখ্যা, আন্ডারস্কোর(\_) অথবা ডলার($) চিহ্ন হতে পারে।

|  |  |
| --- | --- |
| **Note** | প্রথম অক্ষরটি কখনো সংখ্যা হবে না। এই পদ্ধতিতে জাভাস্ক্রিপ্ট খুব সহজেই নাম্বার থেকে আইডেন্টিফায়ারকে পৃথক করতে পারে। |

## জাভাস্ক্রিপ্ট কেস সেনসিটিভ

সকল জাভাস্ক্রিপ্ট আইডেন্টিফায়ার **কেস-সেনসিটিভ(case-sensitive)**।

**lastName** এবং **lastname** এই দুটি ভ্যারিয়েবল সম্পূর্ণ আলাদা

### উদাহরণ

<!DOCTYPE html>

<html lang="bn">

<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>

<h3>জাভাস্ক্রিপ্ট কেস-সেন্সেটিভ</h3>

<h5>lastName কে lastname এ পরিবর্তন করার চেষ্টা করুন।</h5>

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

<script>

var lastName = "Tamim";

var lastname = "Tahmid";

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

</script>

</body>

</html>

Copy

### ফলাফল

জাভাস্ক্রিপ্ট **VAR** অথবা **Var** কে **var** কিওয়ার্ড হিসাবে বিবেচনা করে না।

## জাভাস্ক্রিপ্ট ক্যামেল কেস

প্রোগ্রামাররা একের অধিক শব্দকে এক শব্দে লেখার জন্য সচারচর তিনটি পদ্ধতি ব্যবহার করেঃ

**হাইফেন(-)**

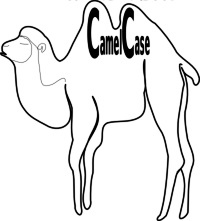
first-name, last-name, master-card, inter-city

**আন্ডারস্কোর(\_)**

first\_name, last\_name, master\_card, inter\_city

**ক্যামেল কেস**

FirstName, LastName, MasterCard, InterCity



প্রোগ্রামিং ল্যাংগুয়েজে বিশেষত জাভাস্ক্রিপ্টে ক্যামেল কেস ছোট হাতের অক্ষর দিয়ে শুরু হয়ঃ

firstName, lastName, masterCard, interCity.

|  |  |
| --- | --- |
| **Note** | জাভাস্ক্রিপ্টে হাইফেন(-) ব্যবহার করা যায় না। জাভাস্ক্রিপ্ট হাইফেনকে বিয়োগ চিহ্ন হিসাবে বিবেচনা করে। |

## জাভাস্ক্রিপ্ট ক্যারেক্টার সেট

জাভাস্ক্রিপ্ট **ইউনিকোড** ক্যারেক্টার সেট ব্যবহার করে।

**ইউনিকোডে** প্রায় সকল ক্যারেক্টার, সিম্বল এবং পাংচুয়েশন থাকে।

**জাভাস্ক্রিপ্ট প্রদর্শনের পদ্ধতি**

জাভাস্ক্রিপ্ট বিভিন্ন পদ্ধতিতে তথ্যকে প্রদর্শন করতে পারেঃ

* **window.alert()** ব্যবহার করে এলার্ট বক্সের মাধ্যমে
* **document.write()** ব্যবহার করে এইচটিএমএল ডকুমেন্টে লিখে
* **.innerHTML** ব্যবহার করে এইচটিএলএল এলিমেন্টে লিখে
* **console.log()** ব্যবহার করে ব্রাউজার কনসোলের মধ্যে দেখানো যায়

**window.alert() ব্যবহার**

তথ্য প্রদর্শনীর জন্য আপনি এলার্ট বক্স ব্যবহার করতে পারবেনঃ

**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>

<h3>আমার প্রথম ওয়েব পেইজ</h3>

<h5>আমার প্রথম অনুচ্ছেদ</h5>

<script>

window.alert("স্বাগতম!");

</script>

</body>

</html>

Copy

**ফলাফল**

**document.write() ব্যবহার**

পরীক্ষা করার উদ্দ্যেশে **document.write()** ব্যবহার করা সুবিধাজনকঃ

**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>

<h3>আমার প্রথম ওয়েব পেইজ</h3>

<h5>আমার প্রথম অনুচ্ছেদ</h5>

<script>

document.write("স্বাগতম!");

</script>

</body>

</html>

Copy

**ফলাফল**

এইচটিএমএল এলিমেন্ট সম্পূর্ন লোড হওয়ার পর document.write() ব্যবহার করলে আগের সকল এইচটিএমএলকে মুছে ফেলেঃ

**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>

<h3>আমার প্রথম ওয়েব পেইজ</h3>

<h5>আমার প্রথম অনুচ্ছেদ</h5>

<button onclick="document.write("স্বাগতম!")">ক্লিক কর</button>

</body>

</html>

Copy

**ফলাফল**

|  |  |
| --- | --- |
| **Note** | শুধুমাত্র পরীক্ষা করার জন্য document.write() পদ্ধতিটি ব্যবহার করা উচিত। |

**innerHTML ব্যবহার**

এইচটিএমএল এলিমেন্টকে এক্সেস করার জন্য জাভাস্ক্রিপ্ট **document.getElementById(id)** পদ্ধতিটি ব্যবহার করা হয়।

**id** এট্রিবিউটটি এইচটিএমএল এলিমেন্টকে নির্দেশ করে এবং **innerHTML** প্রোপার্টিটি এইচটিএমএল কন্টেন্টকে নির্দেশ 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>

<h3>আমার প্রথম ওয়েব পেইজ</h3>

<h5>আমার প্রথম অনুচ্ছেদ</h5>

স্বাগতম!

document.getElementById("test").innerHTML = "স্বাগতম!";

</body>

</html>

Copy

**ফলাফল**

|  |  |
| --- | --- |
| **Note** | এইচটিএমএল এলিমেন্টের innerHTML প্রোপার্টিকে পরিবর্তন করার মাধ্যমে এইচটিএমএলে ডাটা প্রদর্শন করা একটি সাধারণ পদ্ধতি। |

**console.log() ব্যবহার**

আপনার ব্রাউজার কনসোলে ডাটা ডিসপ্লের জন্য আপনি **console.log()** পদ্ধতি ব্যবহার করতে পারেন।

F12 এর মাধ্যমে ব্রাউজার কনসোল সক্রিয় করা হয় এবং মেনু থেকে "Console" নির্বাচন করি।

**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>

<h3>আমার প্রথম ওয়েব পেইজ</h3>

<h5>আমার প্রথম অনুচ্ছেদ</h5>

<script>

console.log("স্বাগতম!");

</script>

</body>

</html>

Copy

**ফলাফল**

স্ট্রিং(String), সংখ্যা(Number), বুলিয়ান(Boolean), অ্যারে(Array), অবজেক্ট(Object)।

## জাভাস্ক্রিপ্ট তথ্যের ধরন(Data Type)

জাভাস্ক্রিপ্ট ভ্যারিয়েবল অনেক টাইপের ডাটা জমা রাখতে পারেঃ সংখ্যা(number), স্ট্রিং(string), অ্যারে(array), অবজেক্ট(object) এবং আরো অনেকঃ

var scores = 16; // সংখ্যা

var name = "Tamim"; // স্ট্রিং

var animals = ["Tiger", "Lion", "Elephant"]; // অ্যারে

var a = {firstName:"Azizur", lastName:"Rahman"}; // অবজেক্ট

Copy

## ডাটা টাইপের ধারনা

প্রোগ্রামিং-এর ক্ষেত্রে, ডাটা টাইপ একটি গুরুত্বপূর্ন বিষয়।

ভ্যারিয়েবল নিয়ে কাজ করতে হলে ডাটা টাইপ জানা খুবই গুরুত্বপূর্ণ।

ডাটা টাইপ ছাড়া কম্পিউটার এগুলোকে(ডাটাকে) সঠিকভাবে সমাধান করতে পারেনাঃ

var a = 28 + "Tamim";

Copy

28 এর সাথে "Tamim" যোগ করা কি কোন অর্থ বহন করে? ইহা কি কোন ভুল(error) দেখাবে নাকি ফলাফল দিবে?

জাভাস্ক্রিপ্ট উপরের উদাহরণকে নিচের মত ভাববেঃ

var a = "28" + "Tamim";

Copy

|  |  |
| --- | --- |
| **Note** | যখন সংখ্যাকে স্ট্রিং-এর সাথে যোগ করা হয়, জাভাস্ক্রিপ্ট সংখ্যাকে স্ট্রিং হিসাবে গণ্য করে। |

### 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>যখন একটি সংখ্যা এবং একটি স্ট্রিং যুক্ত করা হয় তখন জাভাস্ক্রিপ্টের সংখ্যাটি স্ট্রিং হিসাবে গণ্য করে।

।</h4>

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

<script>

var a = 28 + "Tamim";

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

</script>

</body>

</html>

Copy

### ফলাফল

#### যখন একটি সংখ্যা এবং একটি স্ট্রিং যুক্ত করা হয় তখন জাভাস্ক্রিপ্টের সংখ্যাটি স্ট্রিং হিসাবে গণ্য করে। ।

28Tamim

### 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>যখন একটি সংখ্যা এবং একটি স্ট্রিং যুক্ত করা হয়,তখন জাভাস্ক্রিপ্ট সংখ্যাকে স্ট্রিং হিসাবে গণ্য করে। </h4>

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

<script>

var a = "Tamim" + 28;

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

</script>

</body>

</html>

Copy

### ফলাফল

#### যখন একটি সংখ্যা এবং একটি স্ট্রিং যুক্ত করা হয় তখন জাভাস্ক্রিপ্টের সংখ্যাটি স্ট্রিং হিসাবে গণ্য করে। ।

28Tamim

জাভাস্ক্রিপ্ট এক্সপ্রেশনকে বাম থেকে ডান দিকে সম্পাদন করে। ডাটা টাইপের ধারাবাহিকতা ভিন্ন ভিন্ন ফলাফল প্রদর্শন করতে 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>

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

<script>

var a = 28 + 12 + "Tamim";

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

</script>

</body>

</html>

Copy

### ফলাফল

40Tamim

### 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 a = "Tamim" + 28 + 12;

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

</script>

</body>

</html>

Copy

### ফলাফল

Tamim2812

প্রথম উদাহরণে, 28 এবং 12কে সংখ্যা হিসাবে বিবেচনা করে যোগ করে তারপর "Tamim" যেহেতু স্ট্রিং তাই জাভাস্ক্রিপ্ট সম্পূর্নটাকে স্ট্রিং বিবেচনা করে।

দ্বিতীয় উদাহরণে, প্রথম অপারেন্ড স্ট্রিং হওয়ায় সকল অপারেন্ডকেই স্ট্রিং হিসাবে গণ্য করা হয়েছে।

## জাভাস্ক্রিপ্ট ডায়নামিক ডাটা টাইপ

জাভাস্ক্রিপ্টে ডাটা টাইপ ডায়নামিক। এর অর্থ একই ভ্যারিয়েবলে বিভিন্ন টাইপের ডাটা রাখা যেতে পারেঃ

var a; // এখানে a হচ্ছে undefined

var a = 9; // এখানে a হচ্ছে সংখ্যা(Number)

var a = "Tamim"; // এখানে a হচ্ছে স্ট্রিং(String)

Copy

## জাভাস্ক্রিপ্ট স্ট্রিং(String)

স্ট্রিং(string) হচ্ছে কিছু অক্ষরের(character) ধারা। যেমন- "আজিজুর রহমান"।

স্ট্রিংকে উদ্ধৃতি(quote) চিহ্নের মধ্যে লিখতে হয়। আপনি সিঙ্গেল অথবা ডাবল কোটেশন ব্যবহার করতে পারেনঃ

var newCar = "Volvo XC60"; // ডাবল কোট ব্যবহার করে

var newCar = 'Volvo XC60'; //সিঙ্গেল কোট ব্যবহার করে

Copy

আপনি স্ট্রিং-এর মধ্যেও কোটেশন ব্যবহার করতে পারেন, তবে স্ট্রিং-এর উভয় পাশের কোটেশনের সাথে মিলতে পারবে নাঃ

### 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 wel1 = "Welcome To Satt Academy"

var wel2 = "Welcome To 'Satt' Academy"

var wel3 = 'Welcome To "Satt" Academy'

document.getElementById("test").innerHTML =

wel1 + "<br>" + wel2 + "<br>" + wel3;

</script>

</body>

</html>

Copy

### ফলাফল

Welcome To Satt Academy  
Welcome To 'Satt' Academy  
Welcome To "Satt" Academy

এই টিউটোরিয়ালের পরবর্তী পরিচ্ছেদে স্ট্রিং সম্পর্কে আপনি আরো শিখবেন।

## জাভাস্ক্রিপ্ট সংখ্যা

জাভাস্ক্রিপ্টে শুধু এক ধরণের সংখ্যা আছে।

সংখ্যাকে দশমিক অথবা দশমিক ছাড়াও লেখা যায়ঃ

var a = 12.00; // দশমিক(decimals) যুক্ত সংখ্যা

var b = 12; // দশমিক ছাড়া সংখ্যা

Copy

অতিরিক্ত বড় অথবা অতিরিক্ত ছোট সংখ্যাকে বৈজ্ঞানিক (exponential) পদ্ধতিতে লেখা 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>

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

<script>

var a = 12.00;

var b = 12;

var c = 324e4;

var d = 324e-4;

document.getElementById("test").innerHTML = a + "<br>" + b + "<br>" + c + "<br>" + d

</script>

</body>

</html>

Copy

### ফলাফল

12  
12  
3240000  
0.0324

এই টিউটোরিয়ালের পরবর্তী পরিচ্ছেদে সংখ্যা সম্পর্কে আপনি আরো শিখবেন।

## জাভাস্ক্রিপ্ট বুলিয়ান(Boolean)

বুলিয়ানের শুধুমাত্র দুইটি মান থাকেঃ true অথবা false ।

var a = true;

var b = false;

Copy

শর্ত পরীক্ষা(Condition Test) করার জন্য প্রায়ই বুলিয়ান ব্যবহার করা হয়।

এই টিউটোরিয়ালের পরবর্তী পরিচ্ছেদে শর্ত পরীক্ষা(Condition Test) সম্পর্কে আপনি আরো অনেক কিছু শিখবেন।

## জাভাস্ক্রিপ্ট অ্যারে(Array)

জাভাস্ক্রিপ্ট অ্যারেকে তৃতীয় বন্ধনীর(Square Bracket) মাধ্যমে লিখতে হয়।

অ্যারের উপাদানগুলো(items) কমার(,) মাধ্যমে আলাদা করতে হয়।

নিচের উদাহরণে animals নামের একটি অ্যারে ভ্যারিয়েবল ডিক্লেয়ার করা হয়েছে, যার মধ্যে তিনটি মান রয়েছেঃ

### 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 animals = ["Lion", "Tiger", "Horse", "Hen"];

document.getElementById("test").innerHTML = animals[1];

</script>

</body>

</html>

Copy

### ফলাফল

Tiger

অ্যারের ইনডেক্স(index) শুরু হয় 0 দিয়ে, অর্থাৎ প্রথম উপাদানটি হবে [0],দ্বিতীয় উপাদানটি হবে [1], এবং এভাবে চলতে থাকবে।

এই টিউটোরিয়ালের পরবর্তী পরিচ্ছেদে অ্যারে(Array) সম্পর্কে আপনি আরো শিখবেন।

## জাভাস্ক্রিপ্ট অবজেক্ট(Object)

দ্বিতীয় বন্ধনীর(curly bracket) মাধ্যমে জাভাস্ক্রিপ্ট অবজেক্ট লেখা হয়।

অবজেক্টের প্রোপার্টিগুলো name:value জোড়া আকারে লিখতে হয়, একাধিক প্রোপার্টিকে কমার(,) মাধ্যমে পৃথক করতে হয়।

### 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>একটি জাভাস্ক্রিপ্ট অবজেক্ট তৈরি </h4>

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

<script>

var member = {firstName:"Azizur", lastName:"Rahman", age:32, eyeColor:"black"};

document.getElementById("test").innerHTML =

member.firstName + " " + member.lastName + " " + "এর বয়স" + " " + member.age + " " + " বছর।";

</script>

</body>

</html>

Copy

### ফলাফল

#### একটি জাভাস্ক্রিপ্ট অবজেক্ট তৈরি

Azizur Rahman এর বয়স 32 বছর।

উপরের উদাহরণে person অবজেক্টের চারটি প্রোপার্টি আছেঃ firstName, lastName, age, and eyeColor

এই টিউটোরিয়ালের পরবর্তী পরিচ্ছেদে অবজেক্ট(Object) সম্পর্কে আপনি আরো শিখবেন।

## typeof অপারেটর

জাভাস্ক্রিপ্ট ভ্যারিয়েবলের টাইপ জানতে আপনি **typeof** অপারেটর(operator) ব্যবহার করতে পারেনঃ

### 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>

<h3> typeof অপারেটর দ্বারা ভ্যারিয়েবল, অবজেক্ট, ফাংশন, অথবা এক্সপ্রেশন এর টাইপকে বুঝায় ।</h3>

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

<script>

document.getElementById("test").innerHTML =

typeof "Tahmid" + "<br>" +

typeof 3.14 + "<br>" +

typeof NaN + "<br>" +

typeof false + "<br>" +

typeof new Date() + "<br>" +

typeof function () {} + "<br>" +

typeof newDay + "<br>" +

typeof [1,2,3,4] + "<br>" +

typeof {name:'আজিজুর', age:34} + "<br>" +

typeof null;

</script>

</body>

</html><

Copy

### ফলাফল

### typeof অপারেটর দ্বারা ভ্যারিয়েবল, অবজেক্ট, ফাংশন, অথবা এক্সপ্রেশন এর টাইপকে বুঝায় ।

string

number

number

boolean

object

function

undefined

object

object

object

|  |  |
| --- | --- |
| **Note** | জাভাস্ক্রিপ্টে অ্যারে হচ্ছে বিশেষ ধরণের অবজেক্ট(object)। তাই অ্যারের টাইপ অবজেক্ট(Object)। |

## অসংজ্ঞায়িত(Undefined)

জাভাস্ক্রিপ্টে ভ্যারিয়েবলকে ভ্যালু ছাড়া ডিক্লেয়ার করলে এর একটি ভ্যালু থাকে **undefined**। typeof এর মানও **undefined** ।

### 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 member;

document.getElementById("test").innerHTML =

member + "<br>" + typeof member;

</script>

</body>

</html>

Copy

### ফলাফল

undefined  
undefined

ভ্যারিয়েবলের মান **undefined** সেট করে এর মান খালি রাখা যায়।

### 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 member = undefined;

document.getElementById("test").innerHTML =

member + "<br>" + typeof member; // ভ্যালু undefined, টাইপও undefined

</script>

</body>

</html>

Copy

### ফলাফল

undefined  
undefined

## খালি মান(Empty Values)

খালি মান(value) আর অসজ্ঞায়িত দুইটি ভিন্ন জিনিস।

খালি স্ট্রিং-এর ভ্যালু এবং টাইপ(type) দুইটাই থাকে।

### 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 member = "";

document.getElementById("test").innerHTML =

member + "<br>" + typeof member; // ভ্যারিয়েবলের ভ্যালু "", এবং typeof হচ্ছে string

</script>

</body>

</html>

Copy

### ফলাফল

string

## ফাকা(Null)

জাভাস্ক্রিপ্টে null দ্বারা বুঝায় "কিছুই না"। এটা দ্বারা বুঝায় যার কোন অস্তিত্ব নাই।

জাভাস্ক্রিপ্টে, **null** এর ডাটা টাইপ হচ্ছে অবজেক্ট(object)।

|  |  |
| --- | --- |
| **Note** | জাভাস্ক্রিপ্টে null এর typeof অপারেটরের ভ্যালু হচ্ছে অবজেক্ট, আপনি একে bug হিসেবে বিবেচনা করতে পারেন।কারণ এটা null হওয়া উচিৎ ছিল। |

আপনি অবজেক্টের মান null সেট করে একে খালি রাখতে পারেনঃ

### 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 member = null;

document.getElementById("test").innerHTML =

member + "<br>" + typeof member; // মান হচ্ছে null, কিন্তু টাইপ এখনো অবজেক্ট

</script>

</body>

</html>

Copy

### ফলাফল

null  
object

## Undefined এবং Null এর মধ্যে পার্থক্য

### 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>

document.getElementById("test").innerHTML =

typeof undefined + "<br>" +

typeof null + "<br>" +

(null === undefined) + "<br>" +

(null == undefined);

</script>

</body>

</html>

Copy

### ফলাফল

undefined  
object  
false  
true

Top of Form

Bottom of Form