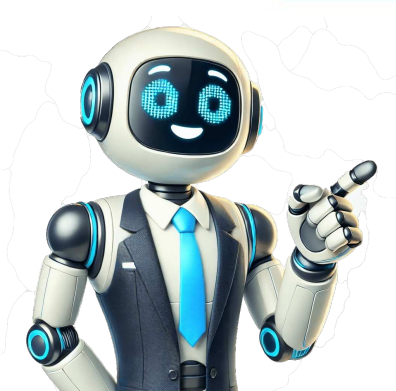


[Click Here](#)



Baseline Widely available Note: This feature is available in Web Workers.The hostname property of the URL interface is a string containing either the domain name or IP address of the URL. If the URL does not have a hostname, this property contains an empty string, "". IPv4 and IPv6 addresses are normalized, such as stripping leading zeros, and domain names are converted to IDN.This property can be set to change the hostname of the URL. If the URL's scheme is not hierarchical (which the URL standard calls "special schemes"), then it has no concept of a host and setting this property has no effect.The hostname is percent-encoded when setting but not percent-decoded when reading.const url = new URL(" ");console.log(url.hostname); // Logs: 'developer.mozilla.org'url.hostname = ".com";console.log(url.hostname); // Logs: 'xn--6qq79v.com'SpecificationURL # dom-url-hostnameThe URL interface it belongs to.In this tutorial, we will see how to extract the hostname portion of a URL in JavaScript.What is a URL?An alternative term for a web address is a URL. For example, tutorialpoints.com is a word-based URL. An IP address can also be used as a URL (ex. 192.168.2.24). Since names are simpler to recall than numbers, most users submit the name's address when searching on the internetA URL is a method by which web browsers ask web servers for specific pages. The syntax/format of a URL is given below.Syntaxscheme://prefix.domain:port/path/filenameParametersscheme ? specifies the kind of Internet service (http or https is being used in general)prefix ? specifies a domain prefix (www is the default for http)domain ? describes the internet domain name (ex.tutorialpoints.com)port ? specifies the host's port number (80 is the default for http)path ? establishes a server-side pathfilename ? specifies the name of a resource or document HTTP - hypertext transfer protocolHTTP is used in common web pages. This is not an encrypted protocol.HTTPS - secure hypertext transfer protocolHTTPS is used in secure web pages. This is an encrypted protocol.FTP - File transfer protocolFTP is used to download or upload files.Now let us see the way to get the hostname in a URL.Using hostname property of current window location In this section, we will see how to get the hostname in a url using the hostname property of the current window locationSyntaxFollow the syntax below to get the hostname.Firstwindow.location.hostname;Here we access the hostname property from the current window's location.AlgorithmSTEP 1 ? Get the current URL.STEP 2 ? Display the hostname from the current URL using the current window location hostname syntax.ExampleIn this example, we have set empty Dom to display output. We have written the code to get the hostname from the current website location. Using the current window location's hostname property var varUrl = window.location.href; var varNullDom = document.getElementById("idNullDom"); varNullDom.innerHTML = "Current url " + varUrl; var varHostDom = document.getElementById("idHostDom"); varHostDom.innerHTML = "Location hostname " + (window.location.hostname); Using hostname Property of the URL.In this section, we will see how to get the hostname in a URL using the hostname property of any URL. Here, we have created the custom URL object using the new URL() constructor.SyntaxFollow the syntax below to get the hostname.var url = new URL("www.sample.com");var hostname = url.hostname;Here we access the hostname property from the URL directly. We use new URL method to create a new URL and then the hostname property of this url is taken.ExampleIn this example, the code creates a sample url using new URL() method. The he hostname is displayed by accessing the hostname property of the URL itself. Using the any url's hostname property var sampUrl = new URL(" "); var varSampDom = document.getElementById("idSampDom"); varSampDom.innerHTML = "url hostname " + (sampUrl.hostname); In this tutorial, we have learned to get the hostname in a URL using the hostname property in JavaScript.The hostname property is a built-in JavaScript property and easy to access the hostname from the current URL as well as from any URL created using new URL() method.If you are working with URLs in JavaScript, you may need to extract specific parts of the URL, such as the hostname. The hostname is the part of the URL that identifies the domain name of the website you are visiting. There are several ways to get the hostname from a URL in JavaScript.Method 1: Using the window.location ObjectThe easiest way to get the hostname from a URL in JavaScript is to use the window.location object. This object contains information about the current URL of the web page, including the hostname.const url = window.location.href;const hostname = window.location.hostname;console.log('URL:', url);console.log('Hostname:', hostname);In the above example, we first get the current URL using window.location.href. We then extract the hostname using window.location.hostname and save it in the hostname variable. Finally, we log both the URL and the hostname to the console.Method 2: Using the URL ObjectAnother way to get the hostname from a URL in JavaScript is to use the URL object. This object provides a convenient way to parse and manipulate URLs.const url = ' ';const parsedUrl = new URL(url);const hostname = parsedUrl.hostname;console.log('URL:', url);console.log('Hostname:', hostname);In this example, we create a new URL object by passing in the URL we want to parse. We then extract the hostname using the hostname property of the parsed URL object and save it in the hostname variable. Finally, we log both the URL and the hostname to the console.Method 3: Using Regular ExpressionsIf you prefer using regular expressions, you can also extract the hostname from a URL using a regular expression pattern.const url = ' ';const regex = /^https?:\/\/([~?#]+)(?:[/?#]!\$)/i;const matches = url.match(regex);const hostname = matches && matches[1];console.log('URL:', url);console.log('Hostname:', hostname);In this example, we define a regular expression pattern that matches the protocol (http or https) and extracts the hostname. We then use the match() method to apply the regular expression pattern to the URL and store the result in the matches variable. Finally, we extract the hostname from the matches array and save it in the hostname variable. We log both the URL and the hostname to the console.These are three ways you can get the hostname from a URL in JavaScript. Choose the method that works best for your use case. Return the hostname of the current URL.let hostname = location.hostname;Try it Yourself The location.hostname propertyreturns the host (IP adress or domain) of a URL.The location.hostname propertycan also be set, to navigate to the same URL with a new hostname. The location.host Property SyntaxReturn the hostname property: Set the hostname property: location.hostname = hostname Property ValueValueDescriptionhostnameThe hostname of the URL.Return Value Type Description A stringThe host (IP address or domain) of a URL. Browser Supportlocation.hostname is supported in all browsers: Chrome Edge Firefox Safari Opera IE Yes Yes Yes Yes Yes Yes The url.hostname is an inbuilt application programming interface of class URL with in url module which is used to get and set the hostname portion of the URL. The key difference between url.host and url.hostname is that url.hostname does not include the port.Syntax:const url.hostnameReturn value: It gets and sets the hostname portion of the URL.Below programs illustrate the use of url.hostname Method:Example 1: javascript // node program to demonstrat the // url.hostname API as Setter //importing the module 'url'const http = require('url');// creating and initializing myURLconst myURL = new URL(' '); // Display href value of myURL before changeconsole.log("Before Change");console.log(myURL.href);// assigning hostname portion// using hostnameconst console.log();myURL.hostname = 'example.org'; // Display href value of myURL after changeconsole.log("After Change");console.log(myURL.href);Output:Example 2: javascript // node program to demonstrate the // url.hostname API as Getter //importing the module 'url'const http = require('url');// creating and initializing myURLconst myURL = new URL(' '); // getting the hostname portion// using hostnameconst hostname = myURL.hostname; // Display hostname value console.log(hostname);Output:NOTE: The above program will compile and run by using the node myapp.js command.Reference: Node.js URL.hash API Node.js URL.host API Node.js URL.origin API Node.js URL.username API Node.js URL.search API Node.js URL.port API Node.js URL.pathname API Node.js URL.password API Node.js URL.href API Node.js URL.hostname API Node.js URL.pathToFileURL API Node.js URL.fileURLToPath API Node.js url.parse(urlString, parseQueryString, slashesDenoteHost) API Special Schemes of Node.js URL.protocol API Node.js URL.protocol API Node.js URL.domainToASCII Node.js URL.domainToUnicode Node.js URL.resolve(from,to) API Node.js URL.format(urlObject) API Node.js URL.format API Node.js URLSearchParams.append() Node.js URLSearchParams.delete() Node.js URLSearchParams.entries() Node.js URLSearchParams.forEach() Node.js URLSearchParams.get() Node.js urlSearchParams.get() Method Node.js URLSearchParams.getAll() Node.js URLSearchParams.sort() Node.js urlSearchParams.toString() Method Node.js urlSearchParams.keys() Method Node.js urlSearchParams.values() Node.js URLSearchParams.set() Node.js URLSearchParams.toString() Node.js URLSearchParams.keys() Node.js URLSearchParams.has() Method Node.js urlObject.auth API Node.js urlObject.hash API Node.js urlObject.host API Node.js urlObject.search API Node.js urlObject.protocol API Node.js urlObject.href API Node.js urlObject.query API Node.js urlObject.slashes API Node.js urlObject.port API Node.js urlObject.pathname API

Land of stories book 5 summary. Land of stories book 6 summary. Land of stories summary. Story of land.