

# URL vs URI: Most important Differences You Must Know

## What is the URL?

A URL is a global address of documents and protocols to retrieve resource on a computer network. URLs occur most frequently in reference to web pages (HTTP) but can also be used for database access using JDBC, email (mailto), file transfer (FTP), and many other applications. URL stands for Uniform Resource Locator.

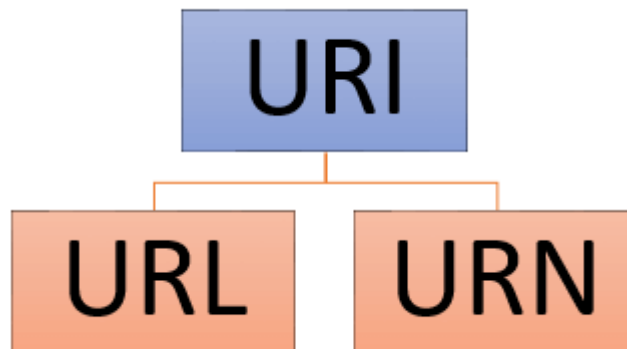
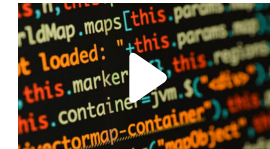
In this URL Vs. URI tutorial, you will learn:

- [What is the URL?](#)
- [What is URI?](#)
- [Ven Diagram of URIs and URL](#)
- [Syntax of URL](#)
- [Syntax of URI](#)
- [Confusion about URN](#)
- [Why URL?](#)
- [Why URI?](#)
- [URL Vs. URI](#)

## What is URI?

A URI is a string containing characters that identify a physical or logical resource. URI follows syntax rules to ensure uniformity. Moreover, it also maintains extensibility via a hierarchical naming scheme. The full form of URI is Uniform Resource Identifier.

What is required ?



[./images/2/022220\\_0627\\_URLvsURIMos1.png](#)

Types of URI

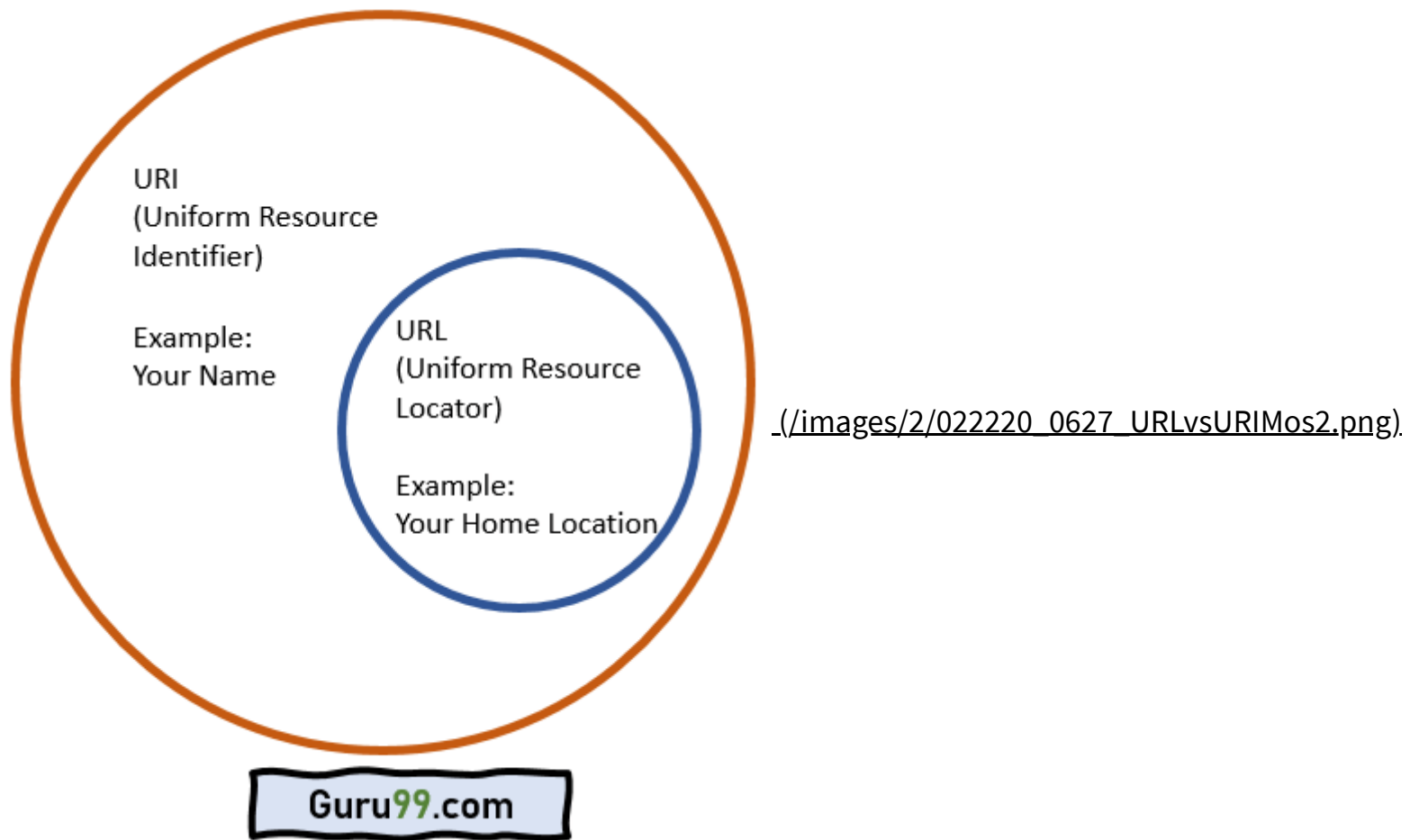
As mention in the above figure, there are two types of URI:

1. **URL:** URL specifies a location on the computer network and technique for retrieving it.
2. **URN:** Uniform Resource Name (URN) is an internet resource that specifies URN scheme.

## KEY DIFFERENCES:

- URL is a subset of URI that specifies where a resource exists and the mechanism for retrieving it, while URI is a superset of URL that identifies a resource
- The main aim of URL is to get the location or address of a resource whereas the main aim of URI is to find a resource.
- URL is used to locate only web pages, on the other hand, URI is used in HTML, XML and other files.
- URL contains components such as protocol, domain, path, hash, query string, etc. while, URI contains components like scheme, authority, path, query, etc.
- Example of URL is : <https://google.com> (<https://google.com>) while example of URI is :urn:isbn:0-486-27557-4.

## Venn Diagram of URIs and URL



Venn diagram of URI and URL

As mention in the above diagram, "your name" can be a URI because it identifies you. It cannot be URL since it does not assist any person to find your home location.

On the other hand, "your home location" can be URI as well as URL. The reason is both identify you and gives a home location for you.

## Syntax of URL

Here is a Syntax of URL:

**http://www.domainname.com/folder-name/web page-file-name.htm**

We can divide the above URL into the following parts:

- **Protocol:** It is the first part of the URL. Here, the protocol name is Hypertext Transfer Protocol (HTTP).
- **http://www.domainname.com/:** It is your domain name. It is also known as server id or the host.
- **/folder-name/:** It indicates that the website page referenced in "filed" in a given folder on the webserver.
- **web-page-file-name.htm:** It is actually a web page file name. The ".htm" is an extension for the HTML file, which shows that it is a static web page. File names can have different extensions or it is depend on how you have set up a web server. There could be no extension at all, and the URL could end with a slash line (/).

### Example:

This example URL has a folder but no extension

<https://career.guru99.com/category/heavy-industries/> (<https://career.guru99.com/category/heavy-industries/>).

This example URL has no folder

<https://www.guru99.com/what-is-sap.html> (<https://www.guru99.com/what-is-sap.html>).

This example URL has no extension

<https://career.guru99.com/top-33-investment-banking-interview-questions-answers/> (<https://career.guru99.com/top-33-investment-banking-interview-questions-answers/>).

# Syntax of URI

Here is a syntax of URI:

**URI = scheme:[//authority]path[?query][#fragment]**

The URI includes the following parts:

- **Scheme component:** It is a non-empty component followed by a colon (:). Scheme contains a sequence of characters starting with a letter and followed by any combination of digits, letters, period (.), hyphen (-), or plus (+). Examples of well-known schemes include HTTP, HTTPS, mailto, file, FTP, etc. URI schemes must be registered with the Internet Assigned Numbers Authority (IANA).
- **Authority component:** It is an optional field and is preceded by //. It consists of
  1. Optional userinfo subcomponent that might consist of a username and password (optional).
  2. A host subcomponent containing either an IP address or a registered name.
  3. An optional port subcomponent that is followed by a colon (:)
- **Path:** A path contains a sequence of segments that are separated by a slash.
- **Query component:** It is optional and preceded by a question mark (?). Query component contains a query string of non-hierarchical data.
- **Fragment component:** It is an optional field and preceded by a hash (#). Fragment component includes a fragment identifier giving direction to a secondary resource.

## Example of URI

No protocol mentioned

[www.guru99.com](https://www.guru99.com) (<https://www.guru99.com>).

Domain not mentioned

what-is-sap.html

Protocol mentions

ldap://[2001:db8::7]/c=GB?objectClass=one  
mailto:[abc@example.com](mailto:abc@example.com) (<mailto:abc@example.com>).  
tel:+1-816-555-1212  
telnet://192.0.2.16:80/

## Confusion about URN

There is a confusion about URN that, if you implement protocols like https, ftp, etc, then it is called a URL, even though it is a URI.

The problem with such a debate is that appropriate [RFC](https://www.ietf.org/rfc/rfc2141.txt) (<https://www.ietf.org/rfc/rfc2141.txt>) are extremely dense and sometimes even contradictory. For example, RFC 3986 says, URI can be either a name, locator or both.

## Why URL?

Here are the important reasons of using URL:

- The Information written in the URL gives you the ability to switch from one web page to another with just one mouse click.
- URL tells you how to access a particular resource.
- Whenever you type a URL into your browser or click any hypertext link, your web browser sends a request to a web server to download one or more files.
- Every URL is unique and identifies one particular file.
- A website URL or domain is one of the most crucial parts of your website. By using simple words or string that usually end with a .org, .com, or .net, you would be able to get traffic to your website.

## Why URI?

Here are the important reasons of using URI:

- A Uniform Resource Identifier is essential to the semantic web because it prevents ambiguity.
- A URI search the name as well as the location of a resource or file, which is in a uniform format.
- It has a string of characters for the specific filename and path.
- URI provides a method for resources to be accessed by other systems over the World Wide Web or across a network. It is used by web browsers and P2P (Peer to Peer) file-sharing software to find and download files.
- URI allows new file types to be defined without affecting old files you have.
- You can assign a single resource to associate with multiple representations.

## URL Vs. URI



[./images/2/022220\\_0627\\_URLvsURIMos3.png](#)



Here are the main differences between URL and URI:

URL	URI
URL stands for Uniform Resource Locator.	URI stands for Uniform Resource Identifier.
URL is a subset of URI that specifies where a resource exists and the mechanism for retrieving it.	A URI is a superset of URL that identifies a resource either by URL or URN (Uniform Resource Name) or both.
The main aim is to get the location or address of a resource	The main aim of URI is to find a resource and differentiate it from other resources using either name or location.
URL is used to locate only web pages	Used in HTML, XML and other files XSLT (Extensible Stylesheet Language Transformations) and more.
The scheme must be a protocol like HTTP, FTP, HTTPS, etc.	In URI, the scheme may be anything like a protocol, specification, name, etc.
Protocol information is given in the URL.	There is no protocol information given in URI.
Example of URL: <a href="https://google.com">https://google.com</a> ( <a href="https://google.com">https://google.com</a> ).	Example of URI: urn:isbn:0-486-27557-4
It contains components such as protocol, domain, path, hash, query string, etc.	It contains components like scheme, authority, path, query, fragment component, etc.
All URLs can be URIs	Not all URIs are URLs since a URI can be a name instead of a locator.

 [Prev](#)

[Report a Bug](#)

[Next](#) 

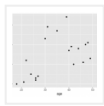
## YOU MIGHT LIKE:

---

[R PROGRAMMING](#)

[DEVOPS](#)

[BLOG](#)



(/r-k-means-clustering.html) (/r-k-means-clustering.html)

## K-means Clustering in R with Example

(/r-k-means-clustering.html)



(/remote-access-desktop-software.html) (/remote-access-desktop-software.html)

## Top Remote Access Software (Remote Desktop Software) 2020

(/remote-access-desktop-software.html)



(/best-ecommerce-platform.html) (/best-ecommerce-platform.html)

## 27 BEST eCommerce Platforms in 2020

(/best-ecommerce-platform.html)

### DEVOPS

---



(/elk-stack-tutorial.html) (/elk-stack-tutorial.html)

## ELK Stack Tutorial: Learn Elasticsearch, Logstash, and Kibana

(/elk-stack-tutorial.html)

### R PROGRAMMING

---



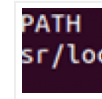
(/r-factor-categorical-continuous.html) (/r-factor-categorical-continuous.html)

## Factor in R: Categorical Variable & Continuous Variables

(/r-factor-categorical-continuous.html)

### LINUX

---



(/linux-environment-variables.html) (/linux-environment-variables.html)

## List of Environment Variables in Linux/Unix

(/linux-environment-variables.html)

# Software Engineering Tutorial

25) Website Vs Web Application (/difference-web-application-website.html)

26) UX Vs UI (/ui-vs-ux.html)

27) URL Vs URI (/url-vs-uri-difference.html)

28) AR Vs VR (/difference-between-ar-vr.html)

29) Encryption Vs Decryption (/difference-encryption-decryption.html)

**f** [\(https://www.facebook.com/guru99com/\)](https://www.facebook.com/guru99com/).

**t** [\\_](https://twitter.com/guru99com) [\(https://twitter.com/guru99com/\)](https://twitter.com/guru99com/). **in**

[\(https://www.linkedin.com/company/guru99/\)](https://www.linkedin.com/company/guru99/).



[\\_](https://www.youtube.com/channel/UC19i1XD6k88KqHlET8atqFQ) [\(https://www.youtube.com/channel/UC19i1XD6k88KqHlET8atqFQ/\)](https://www.youtube.com/channel/UC19i1XD6k88KqHlET8atqFQ/).



[\\_](https://forms.aweber.com/form/46/724807646.htm) [\(https://forms.aweber.com/form/46/724807646.htm/\)](https://forms.aweber.com/form/46/724807646.htm/).

## About

[About Us \(/about-us.html\)](/about-us.html).

[Advertise with Us \(/advertise-us.html\)](/advertise-us.html).

[Write For Us \(/become-an-instructor.html\)](/become-an-instructor.html).

[Contact Us \(/contact-us.html\)](/contact-us.html).

## Career Suggestion

[SAP Career Suggestion Tool \(/best-sap-module.html\)](/best-sap-module.html).

[Software Testing as a Career \(/software-testing-career-complete-guide.html\)](/software-testing-career-complete-guide.html).

## Interesting

[eBook \(/ebook-pdf.html\)](/ebook-pdf.html).

[Blog \(/blog/\)](/blog/).

[Quiz \(/tests.html\)](/tests.html).

[SAP eBook \(/sap-ebook-pdf.html\)](/sap-ebook-pdf.html).

## Execute online

[Execute Java Online \(/try-java-editor.html\)](/try-java-editor.html)

[Execute Javascript \(/execute-javascript-online.html\)](/execute-javascript-online.html)

[Execute HTML \(/execute-html-online.html\)](/execute-html-online.html)

[Execute Python \(/execute-python-online.html\)](/execute-python-online.html)

© Copyright - Guru99 2020

[Privacy Policy \(/privacy-policy.html\)](/privacy-policy.html) | [Affiliate](#)

[Disclaimer \(/affiliate-earning-disclaimer.html\)](/affiliate-earning-disclaimer.html) | [ToS](#)  
[\(/terms-of-service.html\)](/terms-of-service.html)