

## Complete Lesson Plan: What Happens When You Enter a URL in a Browser?

### Stage 1: Desired Results

#### Learning Goals

Cognitive Level	Learning Goals
<b>Remembering</b> (Knowledge Acquisition)	- Define what a URL (Uniform Resource Locator) is.
	- Identify the components of a URL (protocol, domain, path, query parameters).
	- Recall the basic steps of how a browser retrieves a webpage.
<b>Understanding</b> (Comprehension)	- Explain the role of DNS (Domain Name System) in resolving domain names to IP addresses.
	- Describe the client-server communication process in HTTP/HTTPS.
	- Summarize the function of a browser's rendering engine.
<b>Applying</b> (Skill Application)	- Demonstrate how to inspect a webpage's network activity using browser developer tools.
	- Use the command line to manually perform a DNS lookup.
	- Apply knowledge of HTTP requests by making API calls using a browser or tool like Postman.
<b>Analyzing</b> (Critical Thinking)	- Compare HTTP and HTTPS in terms of security and encryption.
	- Examine the impact of slow DNS resolution on webpage loading speed.
	- Differentiate between GET and POST requests in web browsing.
<b>Evaluating</b> (Critical Thinking)	- Assess the importance of caching in optimizing webpage loading speed.
	- Evaluate the security risks associated with an unsecured connection.
	- Justify the need for browser cookies and session storage in maintaining user experience.

<b>Creating</b> (Innovation)	- Design a basic HTML page and host it on a local server to visualize network requests.
	- Construct a simple explanation or animation to teach others how the web works.

### Understandings and Essential Questions

Enduring Understandings	Essential Questions
- The process of accessing a webpage involves multiple systems, including DNS, HTTP, and rendering engines.	- What happens behind the scenes when you type a URL into a browser?
- The internet relies on client-server architecture for communication.	- How does the browser retrieve and display a webpage?
- DNS resolution is critical for converting human-friendly domain names into machine-readable IP addresses.	- Why is DNS essential for web browsing?
- HTTP/HTTPS protocols define how browsers and servers communicate.	- How do HTTP and HTTPS differ, and why does it matter?
- Browsers parse and render HTML, CSS, and JavaScript to display web content.	- How does the browser interpret and display webpage content?
- Security and performance considerations affect user experience.	- What factors influence how fast and secure a webpage loads?

### Stage 2: Evidence

#### Performance Tasks

Performance Task	Criteria / Rubrics
<b>Trace the URL Journey:</b> Students will create a detailed step-by-step flowchart illustrating what happens when a URL is entered into a browser.	- Accuracy in depicting the request-response cycle (DNS lookup, HTTP request, server response, rendering).
	- Clear labeling and logical sequencing of steps.
	- Use of appropriate technical terminology.

<b>Network Inspection Activity:</b> Students will use browser developer tools to analyze network activity and explain key HTTP request-response elements.	- Ability to navigate and utilize network inspection tools effectively.
	- Identification of request/response headers and status codes.
<b>DNS Lookup Experiment:</b> Students will perform a manual DNS lookup using the command line and document their findings.	- Successful execution of DNS lookup commands.
	- Accurate interpretation of IP address resolution.
	- Reflection on the importance of DNS in web browsing.

#### Other Evidence

Evidence Type	Description
<b>Quizzes / Tests (Optional)</b>	Short quiz assessing key concepts like HTTP/HTTPS, DNS, and caching.
<b>Observations (Required)</b>	Instructor monitors students as they use network tools and command-line utilities.
<b>Class Discussions (Required)</b>	Students discuss performance factors affecting webpage loading and security implications.

### Stage 3: Plan Learning Experiences and Instruction

#### Engage (up to 5 min)

**Objective:** Spark curiosity and connect prior knowledge to the topic.

- Show a short animation or infographic illustrating the journey of a URL from browser to webpage.
- Ask students: *“What do you think happens behind the scenes when you enter a URL?”*
- Conduct a quick poll on common misconceptions (e.g., “The browser directly fetches the webpage from the internet”).

#### Explore (up to 5 min)

**Objective:** Allow students to investigate and make observations.

- Have students use **tracert (Windows) or traceroute (Mac/Linux)** to track the path of a data request and analyze the number of hops involved.
- Encourage students to share their findings and discuss how their requests are routed.

### Explain (10–15 min)

**Objective:** Provide direct instruction and clarify concepts.

- Explain the step-by-step process:
  1. DNS resolution
  2. Establishing a connection (TCP/IP, HTTPS handshake)
  3. Request processing on the web server
  4. Response transmission back to the browser
  5. Rendering by the browser (HTML, CSS, JavaScript)
- Demonstrate how to inspect **network activity** using browser developer tools.
- Show an example of **HTTP request-response cycles** using a real webpage.

### Elaborate / Extend (up to 10 min)

**Objective:** Deepen understanding through application.

- Divide students into small groups and assign each group a specific part of the URL journey (e.g., DNS, HTTP requests, server response, browser rendering).
- Have each group create a **mini-presentation or visual diagram** explaining their assigned process.
- Conduct a **peer-sharing session**, where groups teach one another.

### Evaluate / Performance Task

**Objective:** Assess students' understanding and application.

- Students complete the **Trace the URL Journey** performance task by creating a flowchart of the process.
- Instructor circulates, providing feedback and ensuring logical accuracy.
- Students explain their diagrams to a partner and answer questions.

### Wrap Up

**Objective:** Summarize learning and reinforce key points.

- Conduct a quick **Q&A session** where students ask lingering questions.
- Provide a **one-minute summary challenge**, where students summarize the URL journey in 60 seconds.

**Homework (if needed)**

- Optional: Students write a short reflection on *why security measures (e.g., HTTPS) are essential in modern web browsing*.