**Basics of Internet/Web?**

The Internet is a large network of computers which communicate all together.

When two computers need to communicate, you have to link them

* physically (usually with an [Ethernet cable](https://en.wikipedia.org/wiki/Ethernet_crossover_cable)) or
* wirelessly (for example with [Wi-Fi](https://en.wikipedia.org/wiki/WiFi) or [Bluetooth](https://en.wikipedia.org/wiki/Bluetooth) systems)

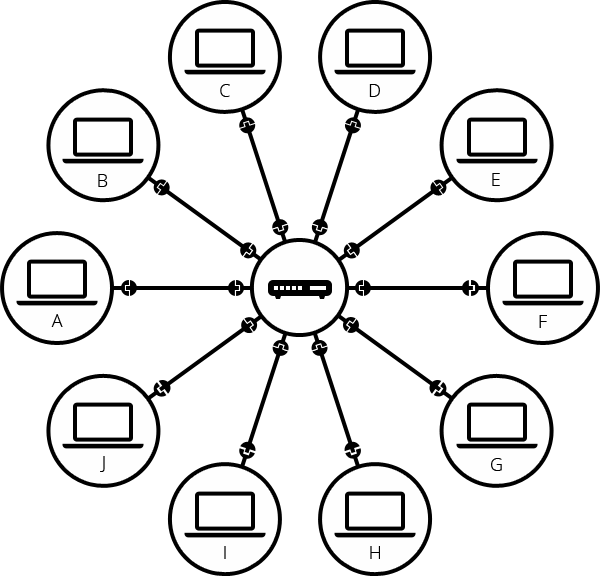
Such a network is not limited to two computers. You can connect as many computers as you wish. But it gets complicated quickly**. If you're trying to connect, say, ten computers, you need 45 cables, with nine plugs per computer**

**So what??**

**Solution:**

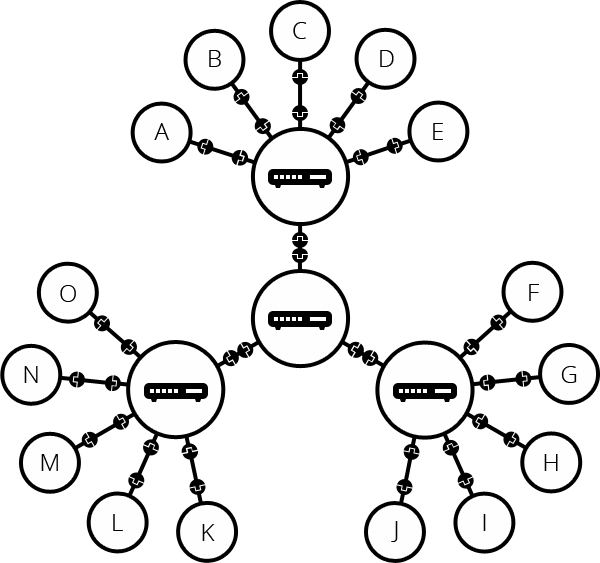
Each computer on a network is connected to a special tiny computer called a **router**.

Router makes sure that a message sent from a given computer arrives at the right destination computer.



**But what about connecting hundreds, thousands, billions of computers?**

By connecting computers to routers, then routers to routers, we are able to scale infinitely.

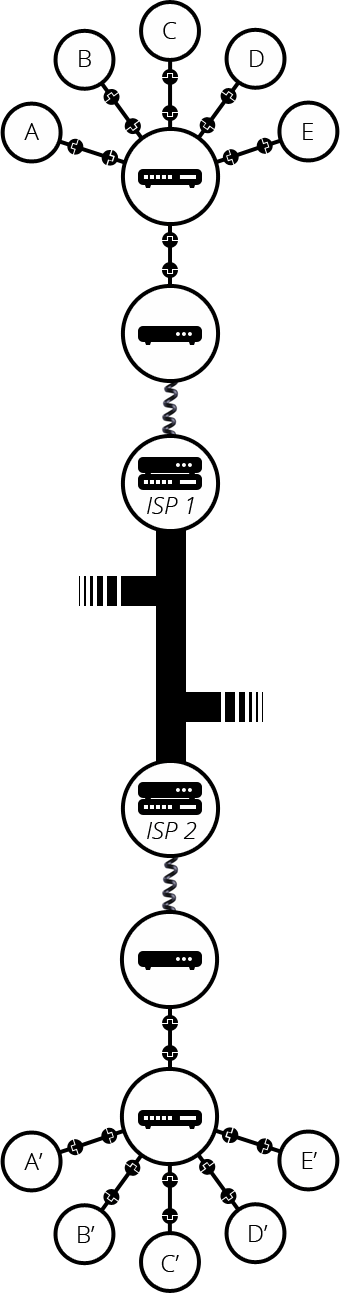


**We built that network for our own purposes. There are other networks out there: your friends, your neighbors, anyone can have their own network of computers. But it's not really possible to set cables up between your house and the rest of the world, so how can you handle this?**

To connect our network to the telephone infrastructure, we need a special piece of equipment called a **modem**. *This*modem*turns the information from our network into information manageable by the telephone infrastructure and vice versa.*

The next step is to send the messages from our network to the network we want to reach. To do that, we will connect our network to an **Internet Service Provider (ISP).**

An ISP is a company that manages some special routers that are all linked together and can also access other ISPs' routers. So the message from our network is carried through the network of ISP networks to the destination network. The Internet consists of this whole infrastructure of networks.



**Folder Structure**

****

**HTML Elements – Block and Inline**

A block-level element always takes up the full width available

An inline element does not start on a new line.

Here are the block-level elements in HTML:

[<address>](https://www.w3schools.com/tags/tag_address.asp)

[<article>](https://www.w3schools.com/tags/tag_article.asp)

[<aside>](https://www.w3schools.com/tags/tag_aside.asp)

[<blockquote>](https://www.w3schools.com/tags/tag_blockquote.asp)

[<canvas>](https://www.w3schools.com/tags/tag_canvas.asp)

[<dd>](https://www.w3schools.com/tags/tag_dd.asp)

[<div>](https://www.w3schools.com/tags/tag_div.asp)

[<dl>](https://www.w3schools.com/tags/tag_dl.asp)

[<dt>](https://www.w3schools.com/tags/tag_dt.asp)

[<fieldset>](https://www.w3schools.com/tags/tag_fieldset.asp)

[<figcaption>](https://www.w3schools.com/tags/tag_figcaption.asp)

[<figure>](https://www.w3schools.com/tags/tag_figure.asp)

[<footer>](https://www.w3schools.com/tags/tag_footer.asp)

[<form>](https://www.w3schools.com/tags/tag_form.asp)

[<h1>-<h6>](https://www.w3schools.com/tags/tag_hn.asp)

[<header>](https://www.w3schools.com/tags/tag_header.asp)

[<hr>](https://www.w3schools.com/tags/tag_hr.asp)

[<li>](https://www.w3schools.com/tags/tag_li.asp)

[<main>](https://www.w3schools.com/tags/tag_main.asp)

[<nav>](https://www.w3schools.com/tags/tag_nav.asp)

[<noscript>](https://www.w3schools.com/tags/tag_noscript.asp)

[<ol>](https://www.w3schools.com/tags/tag_ol.asp)

[<p>](https://www.w3schools.com/tags/tag_p.asp)

[<pre>](https://www.w3schools.com/tags/tag_pre.asp)

[<section>](https://www.w3schools.com/tags/tag_section.asp)

[<table>](https://www.w3schools.com/tags/tag_table.asp)

[<tfoot>](https://www.w3schools.com/tags/tag_tfoot.asp)

[<ul>](https://www.w3schools.com/tags/tag_ul.asp)

[<video>](https://www.w3schools.com/tags/tag_video.asp)

Here are the inline elements in HTML:

[<a>](https://www.w3schools.com/tags/tag_a.asp)

[<abbr>](https://www.w3schools.com/tags/tag_abbr.asp)

[<acronym>](https://www.w3schools.com/tags/tag_acronym.asp)

[<b>](https://www.w3schools.com/tags/tag_b.asp)

[<bdo>](https://www.w3schools.com/tags/tag_bdo.asp)

[<big>](https://www.w3schools.com/tags/tag_big.asp)

[<br>](https://www.w3schools.com/tags/tag_br.asp)

[<button>](https://www.w3schools.com/tags/tag_button.asp)

[<cite>](https://www.w3schools.com/tags/tag_cite.asp)

[<code>](https://www.w3schools.com/tags/tag_code.asp)

[<dfn>](https://www.w3schools.com/tags/tag_dfn.asp)

[<em>](https://www.w3schools.com/tags/tag_em.asp)

[<i>](https://www.w3schools.com/tags/tag_i.asp)

[<img>](https://www.w3schools.com/tags/tag_img.asp)

[<input>](https://www.w3schools.com/tags/tag_input.asp)

[<kbd>](https://www.w3schools.com/tags/tag_kbd.asp)

[<label>](https://www.w3schools.com/tags/tag_label.asp)

[<map>](https://www.w3schools.com/tags/tag_map.asp)

[<object>](https://www.w3schools.com/tags/tag_object.asp)

[<output>](https://www.w3schools.com/tags/tag_output.asp)

[<q>](https://www.w3schools.com/tags/tag_q.asp)

[<samp>](https://www.w3schools.com/tags/tag_samp.asp)

[<script>](https://www.w3schools.com/tags/tag_script.asp)

[<select>](https://www.w3schools.com/tags/tag_select.asp)

[<small>](https://www.w3schools.com/tags/tag_small.asp)

[<span>](https://www.w3schools.com/tags/tag_span.asp)

[<strong>](https://www.w3schools.com/tags/tag_strong.asp)

[<sub>](https://www.w3schools.com/tags/tag_sub.asp)

[<sup>](https://www.w3schools.com/tags/tag_sup.asp)

[<textarea>](https://www.w3schools.com/tags/tag_textarea.asp)

[<time>](https://www.w3schools.com/tags/tag_time.asp)

[<tt>](https://www.w3schools.com/tags/tag_tt.asp)

[<var>](https://www.w3schools.com/tags/tag_var.asp)

**Cascading Style Sheet(CSS)**

There are three types of CSS which are given below: 

* Inline CSS

**<p style = "color:#009900; font-size:50px;**

**font-style:italic; text-align:center;">**

**</p>**

* Internal or Embedded CSS

**<head>**

**<title>Internal CSS</title>**

**<style>**

**.main {**

**text-align:center;**

**}**

**.GFG {**

**color:#009900;**

**font-size:50px;**

**font-weight:bold;**

**}**

**.geeks {**

**font-style:bold;**

**font-size:20px;**

**}**

**</style>**

**</head>**

* External CSS – styles.css

**Css selectors**

* Simple selectors (select elements based on name, id, class)
* [Combinator selectors](https://www.w3schools.com/css/css_combinators.asp) (select elements based on a specific relationship between them)

1. descendant selector (space)

<style>

div p {

background-color: yellow;

}

</style>

<div>

<p>Paragraph 1 in the div.</p>

<p>Paragraph 2 in the div.</p>

<section><p>Paragraph 3 in the div.</p></section>

</div>

<p>Paragraph 4. Not in a div.</p>

<p>Paragraph 5. Not in a div.</p>

1. child selector (>)

<style>

div > p {

background-color: yellow;

}

</style>

<div>

<p>Paragraph 1 in the div.</p>

<p>Paragraph 2 in the div.</p>

<section>

<!-- not Child but Descendant -->

<p>Paragraph 3 in the div (inside a section element).</p>

</section>

<p>Paragraph 4 in the div.</p>

</div>

<p>Paragraph 5. Not in a div.</p>

<p>Paragraph 6. Not in a div.</p>

1. adjacent sibling selector (+) – selects the <p> adjacent to <div>

div + p {

background-color: yellow;

}

<div>

<p>Paragraph 1 in the div.</p>

<p>Paragraph 2 in the div.</p>

</div>

<p>Paragraph 3. After a div.</p>

<p>Paragraph 4. After a div.</p>

<div>

<p>Paragraph 5 in the div.</p>

<p>Paragraph 6 in the div.</p>

</div>

<p>Paragraph 7. After a div.</p>

<p>Paragraph 8. After a div.</p>

1. general sibling selector (~) – selects all the <p> adjacent to <div>

div ~ p {

background-color: yellow;

}

<div>

<p>Paragraph 2.</p>

</div>

<p>Paragraph 3.</p>

<code>Some code.</code>

<p>Paragraph 4.</p>

* [Pseudo-class selectors](https://www.w3schools.com/css/css_pseudo_classes.asp) (select elements based on a certain state)

|  |  |
| --- | --- |
| /\* unvisited link \*/  a:link {  color: red;  } | /\* visited link \*/  a:visited {  color: green;  } |
| /\* mouse over link \*/  a:hover {  color: hotpink;  } | /\* selected link \*/  a:active {  color: blue;  } |

* [Pseudo-elements selectors](https://www.w3schools.com/css/css_pseudo_elements.asp) (select and style a part of an element)
* [Attribute selectors](https://www.w3schools.com/css/css_attribute_selectors.asp) (select elements based on an attribute or attribute value)

[class|=top] {

background: yellow;

}

<h1 class="top-header">Welcome</h1>

<p class="top-text">Hello world!</p>

<p class="topcontent">Are you learning CSS?</p>

[class^="top"] {

background: yellow;

}

<h2>CSS [attribute^="value"] Selector</h2>

<h1 class="top-header">Welcome</h1>

<p class="top-text">Hello world!</p>

<p class="topcontent">Are you learning CSS?</p>

[class$="test"] {

background: yellow;

}

<div class="first\_test">The first div element.</div>

<div class="second">The second div element.</div>

<div class="my-test">The third div element.</div>

<p class="mytest">This is some text in a paragraph.</p>

**CSS Specificity:**

There are four categories which define the specificity level of a selector:

1. **Inline styles** - Example: <h1 style="color: pink;">
2. **IDs** - Example: #navbar
3. **Classes, pseudo-classes, attribute selectors** - Example: .test, :hover, [href]
4. **Elements and pseudo-elements** - Example: h1, :before

**Display Property**

p.ex1 {display: none;} - will not be displayed

p.ex2 {display: inline;} – will be added inside the paragraph

p.ex3 {display: block;} – Will be a separate block in next line

p.ex4 {display: inline-block;} – it will be given height and width of its own