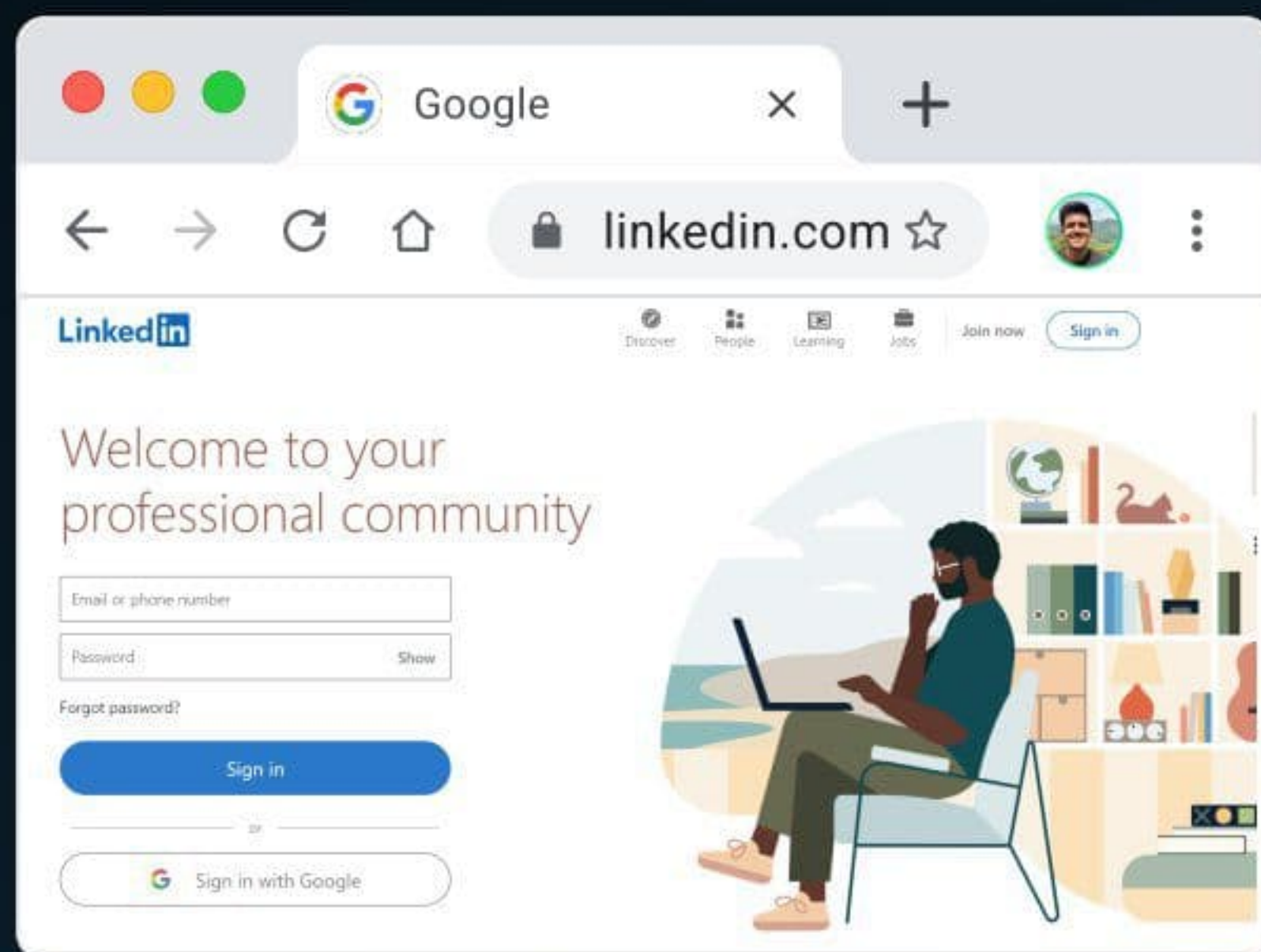




HTML

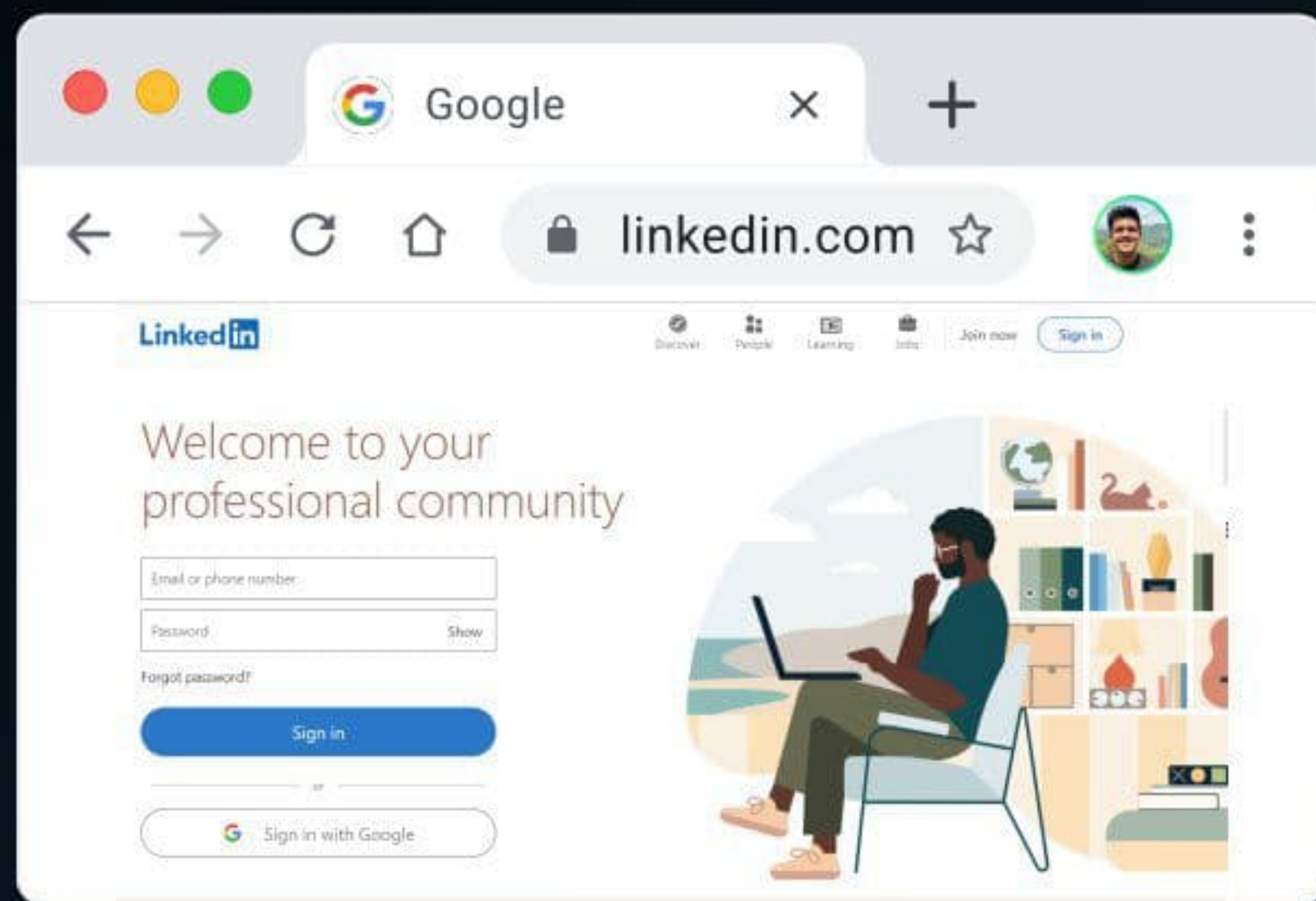
Lifecycle of a webpage ✨



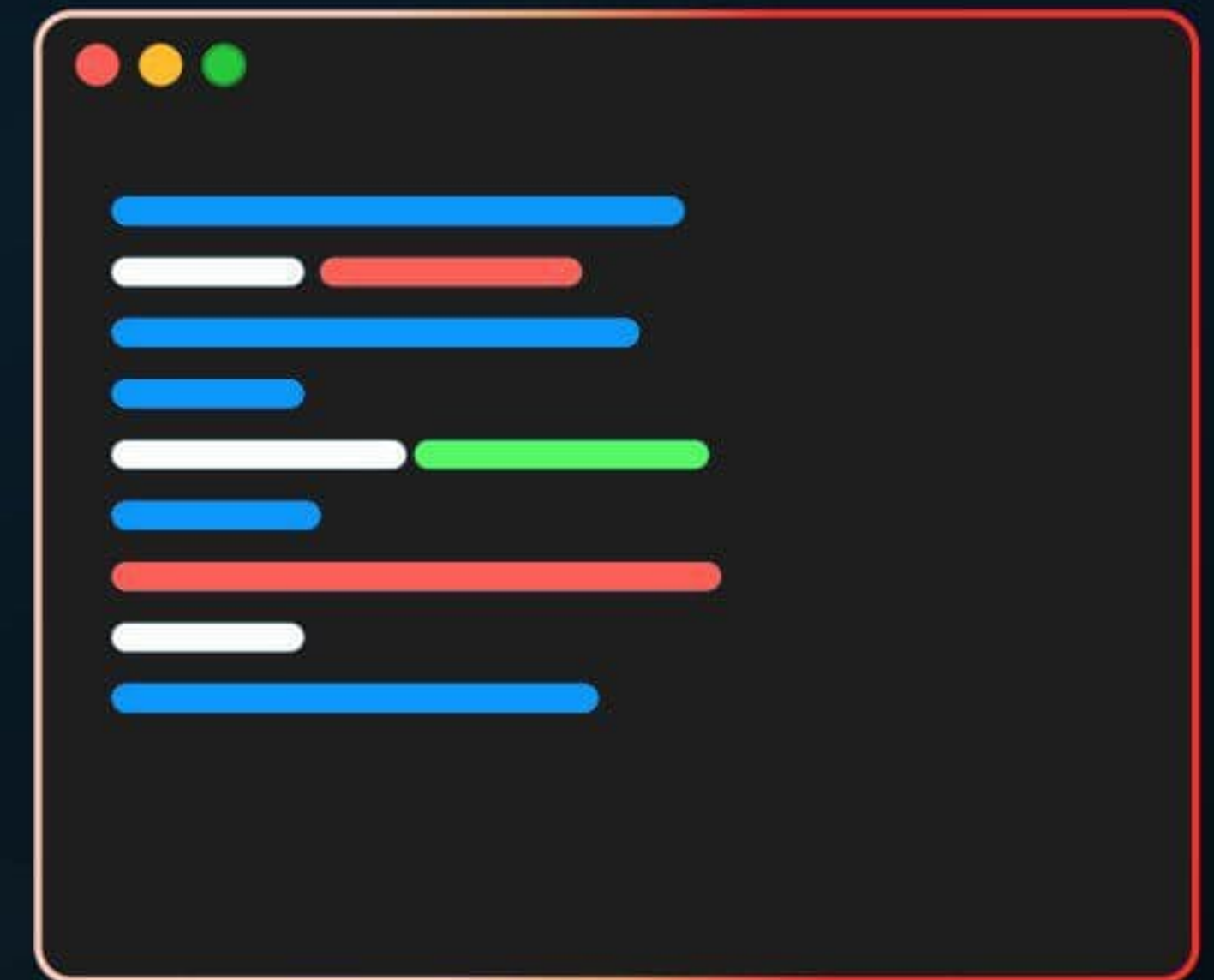


The steps involved

Have you ever wondered how does your browser render HTML pages from the internet ?



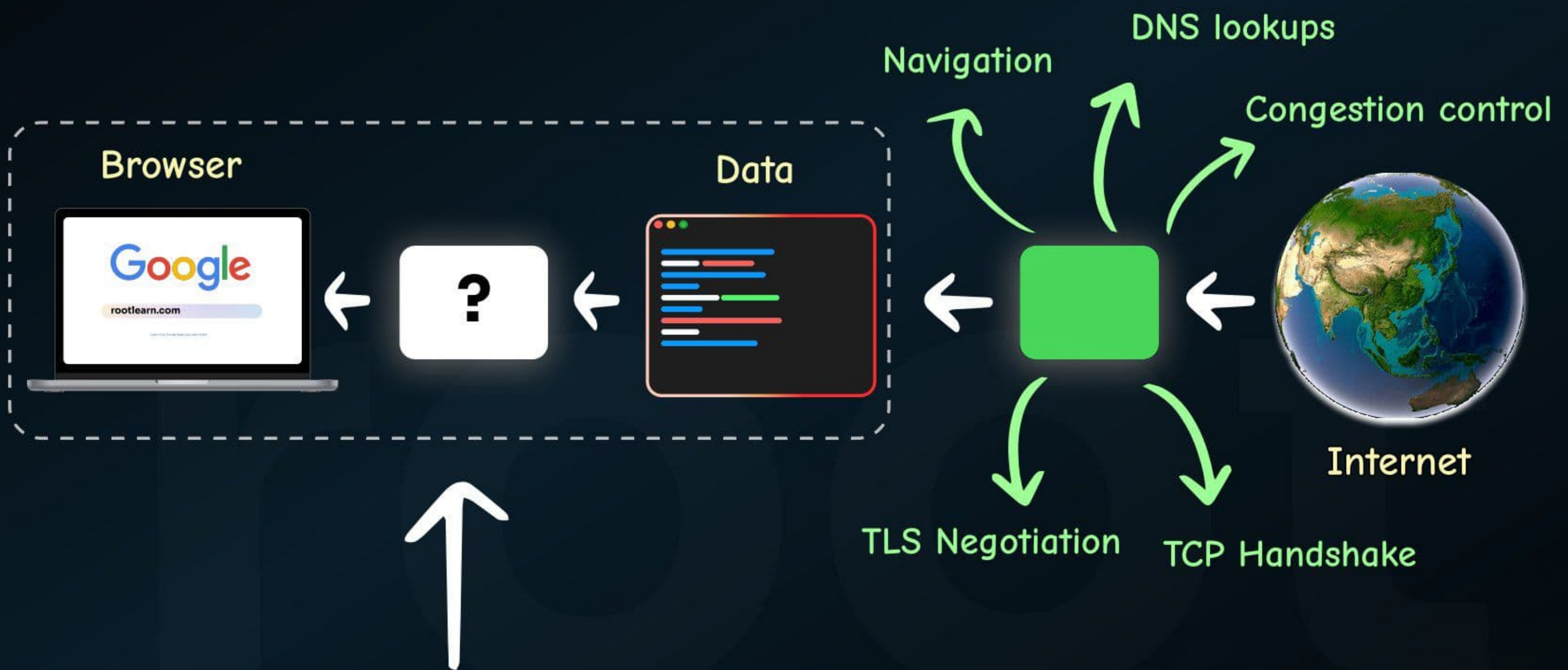
There is a lot going on
that we don't see





Before we begin

When you type in some url in your browser, a lot happens behind the scenes on how it fetches data from the internet



The part that we'll focus on





1. Byte Decoder

The data that you actually receive from the internet is in the form of bytes

The byte stream decoder converts the bytes to a set of tokens

```
3C 62 6F 64 74 87 4C 79 3E 61
3E 85 70 71 6F H4 2F 12 41 3C
62 6F 6F 74 87 4C 79 3E 61 3E
85 70 71 6F H4 2F 12 41 3C 62
6F 6F 74 87 4C 79 3E 61 3E 85
70 71 6F H4 2F 12 41
```

Byte Decoder

bytes to characters

```
<html><head><title>rootlearn
</title></head><body><head
er>ROOT</header><p>I'm Mani
sh</p></div>
```



characters to tokens

startTag: html

startTag: head

startTag: title

content: rootlearn

endTag: title

endTag: head

startTag: body

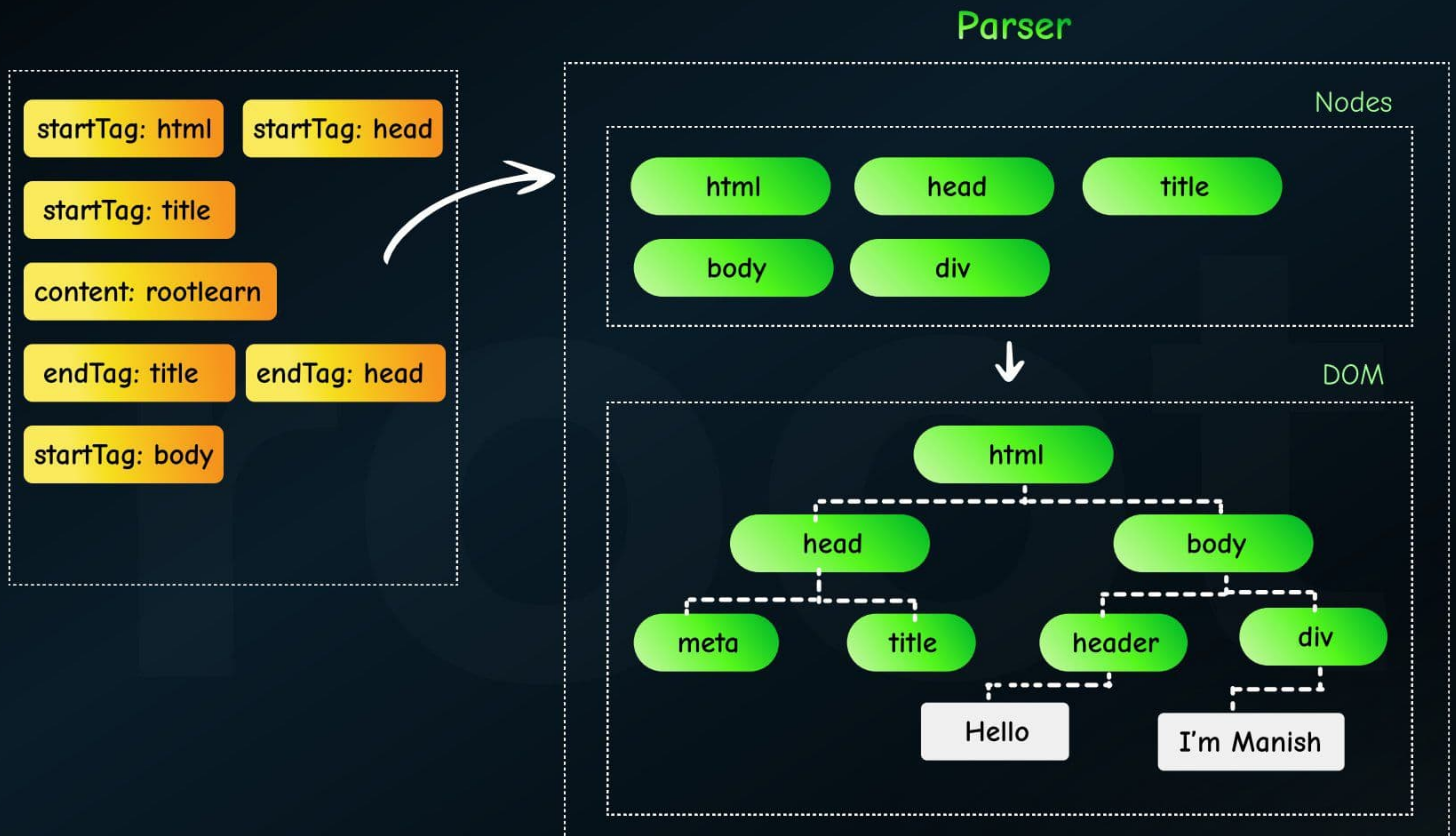
...

...



2. Parsing

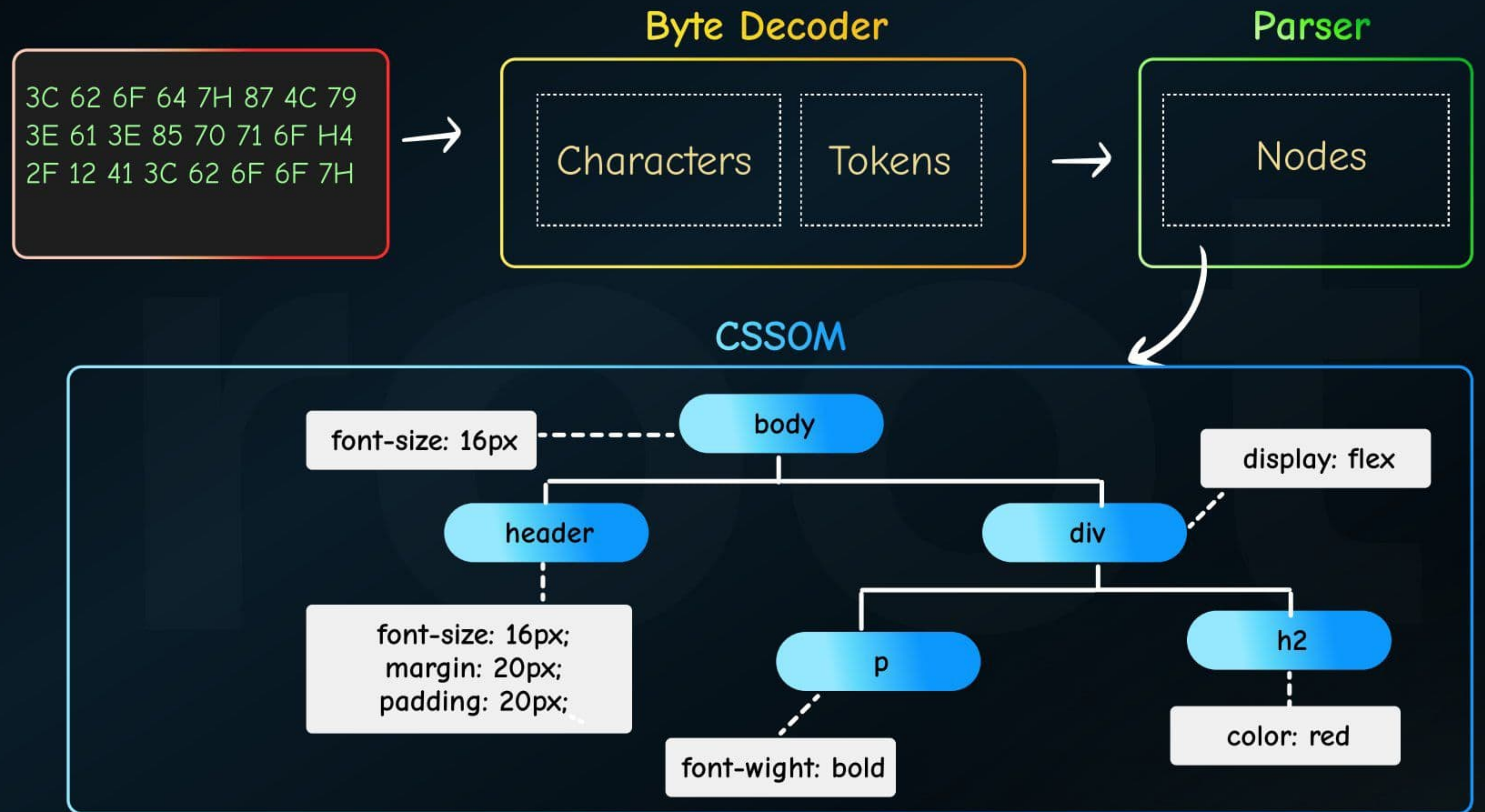
The parser starts converting the tokens it receives to a set of nodes and creates the DOM tree





3. Recalculating styles

It repeats this whole process with a CSS file as well and creates a CSSOM tree

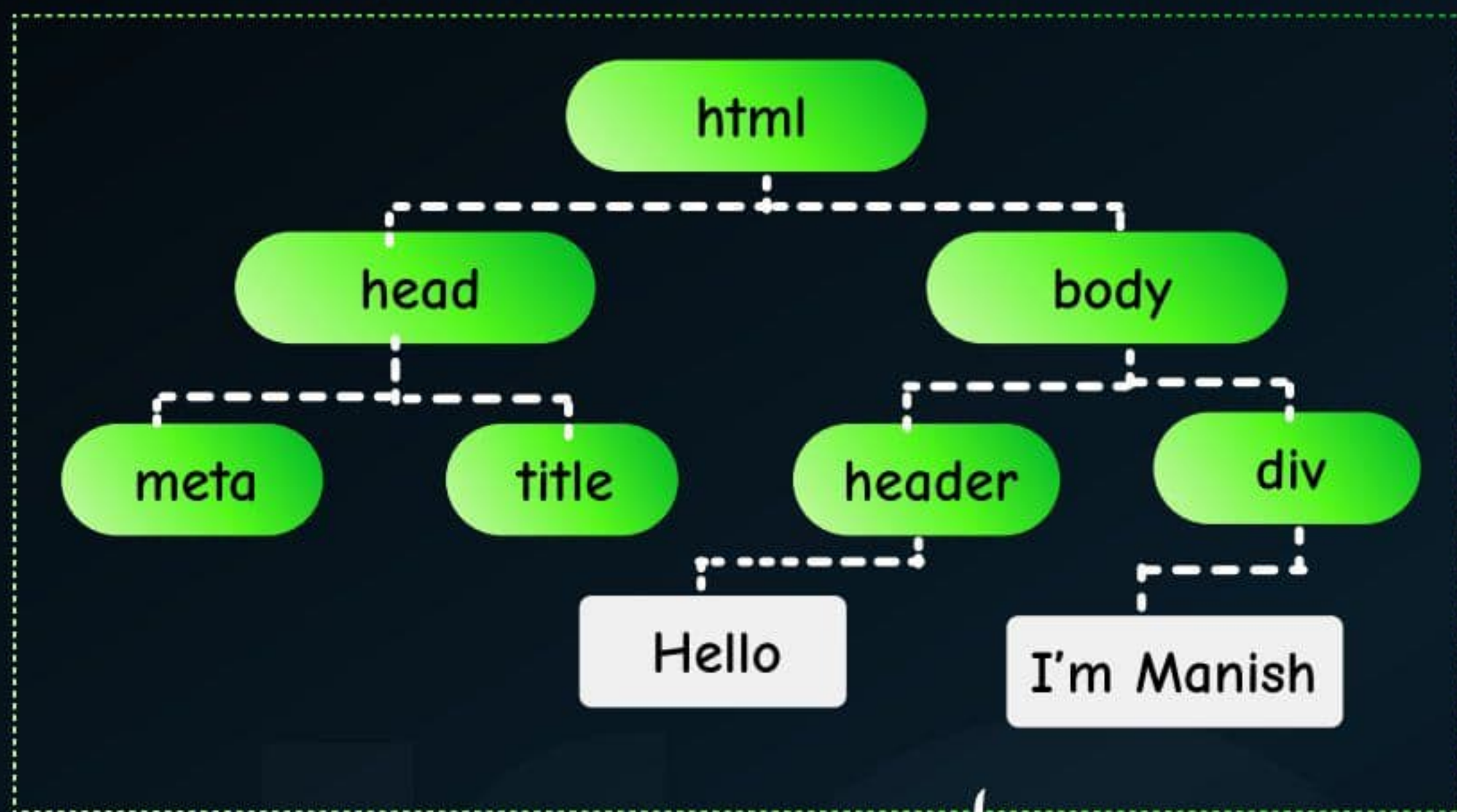




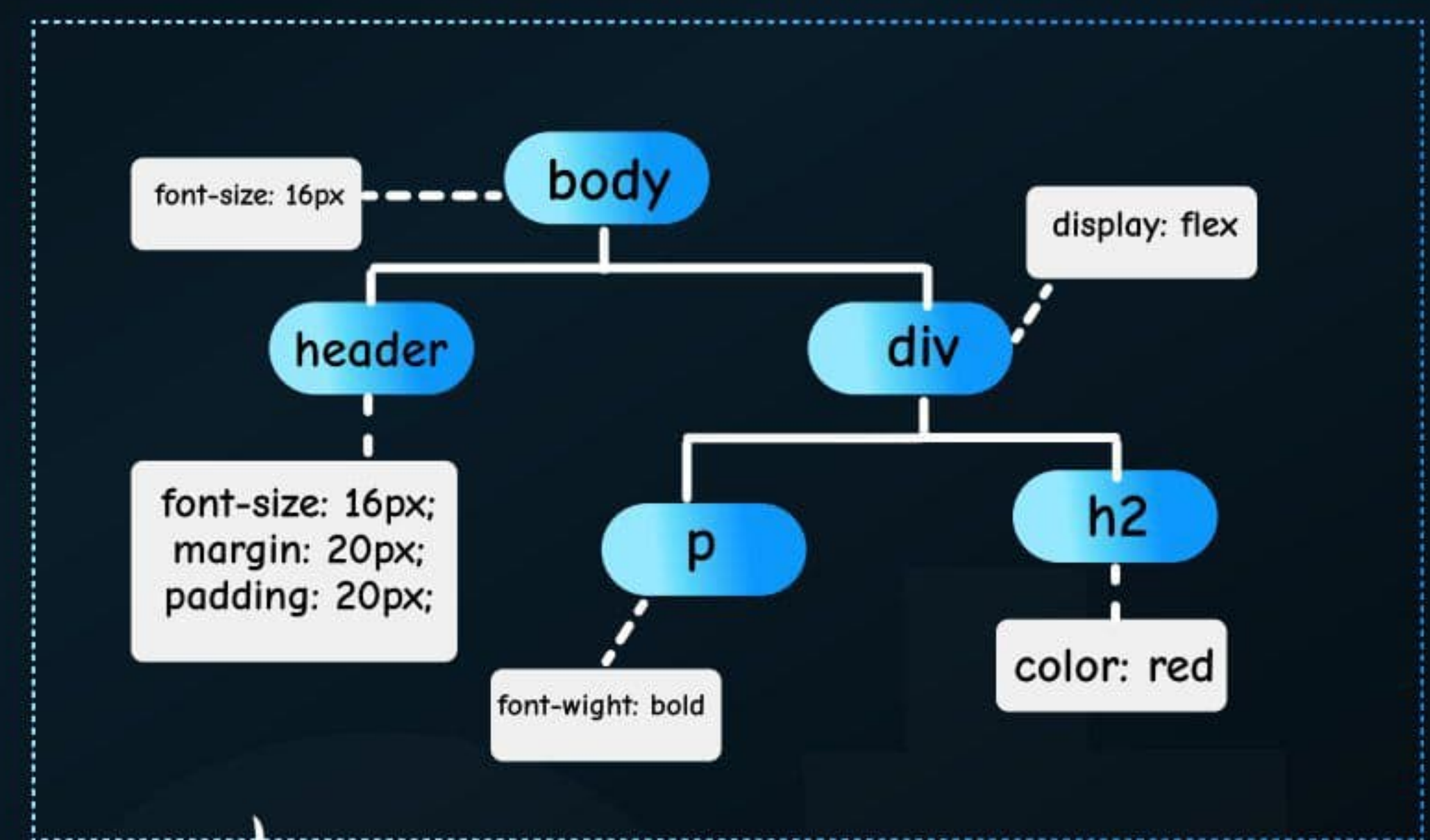
Generating the render tree

The browser combines the DOM and CSSOM and creates the render tree. It's the visible content on the screen

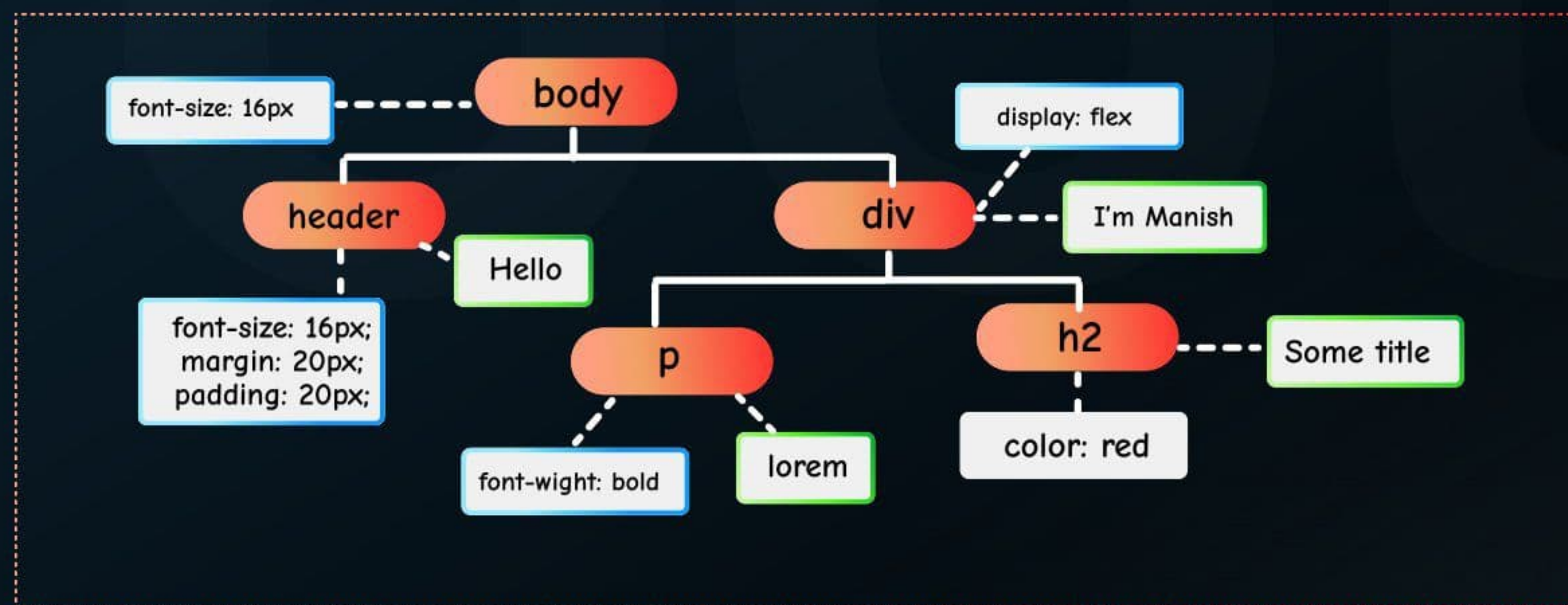
DOM



CSSOM



Render Tree

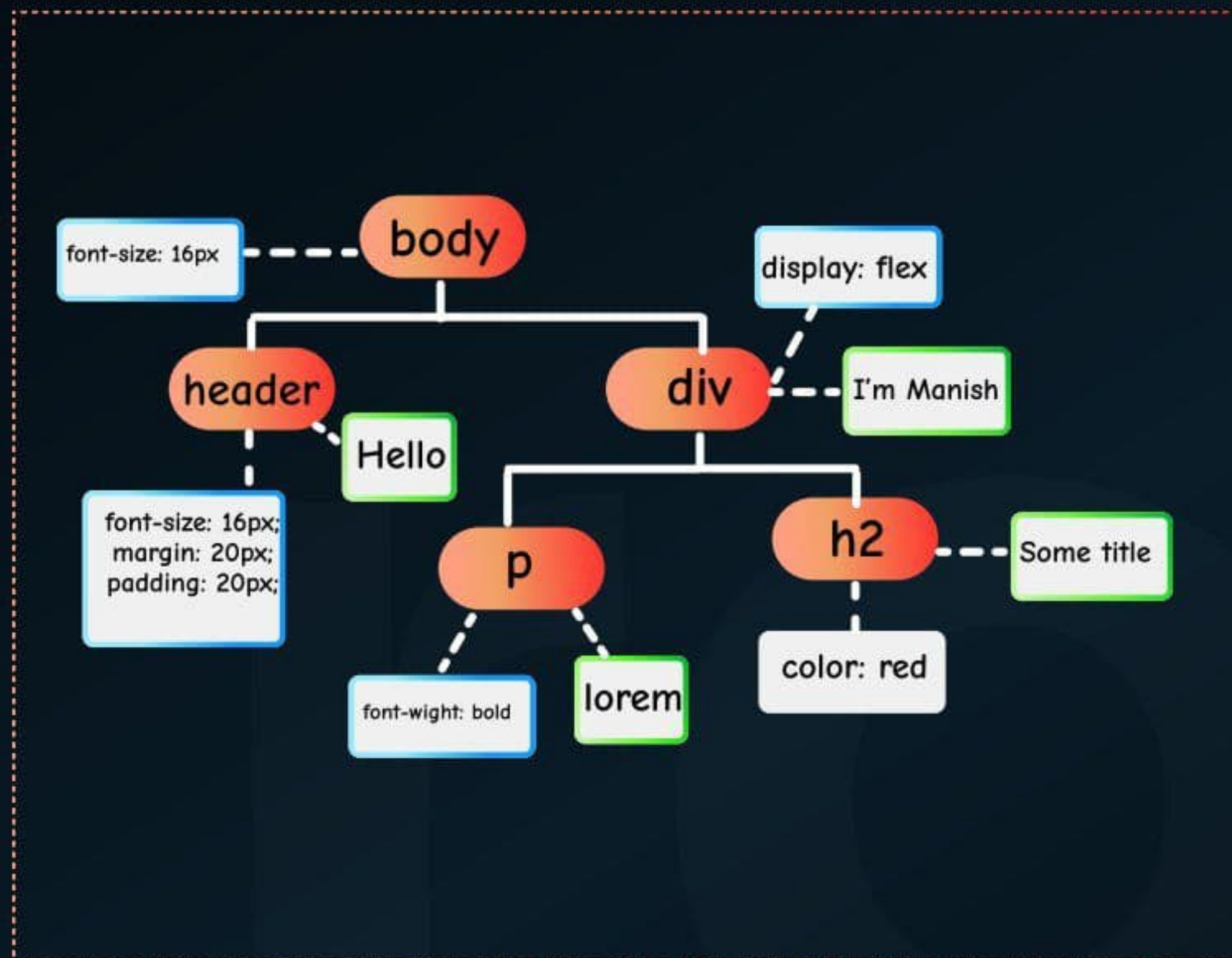




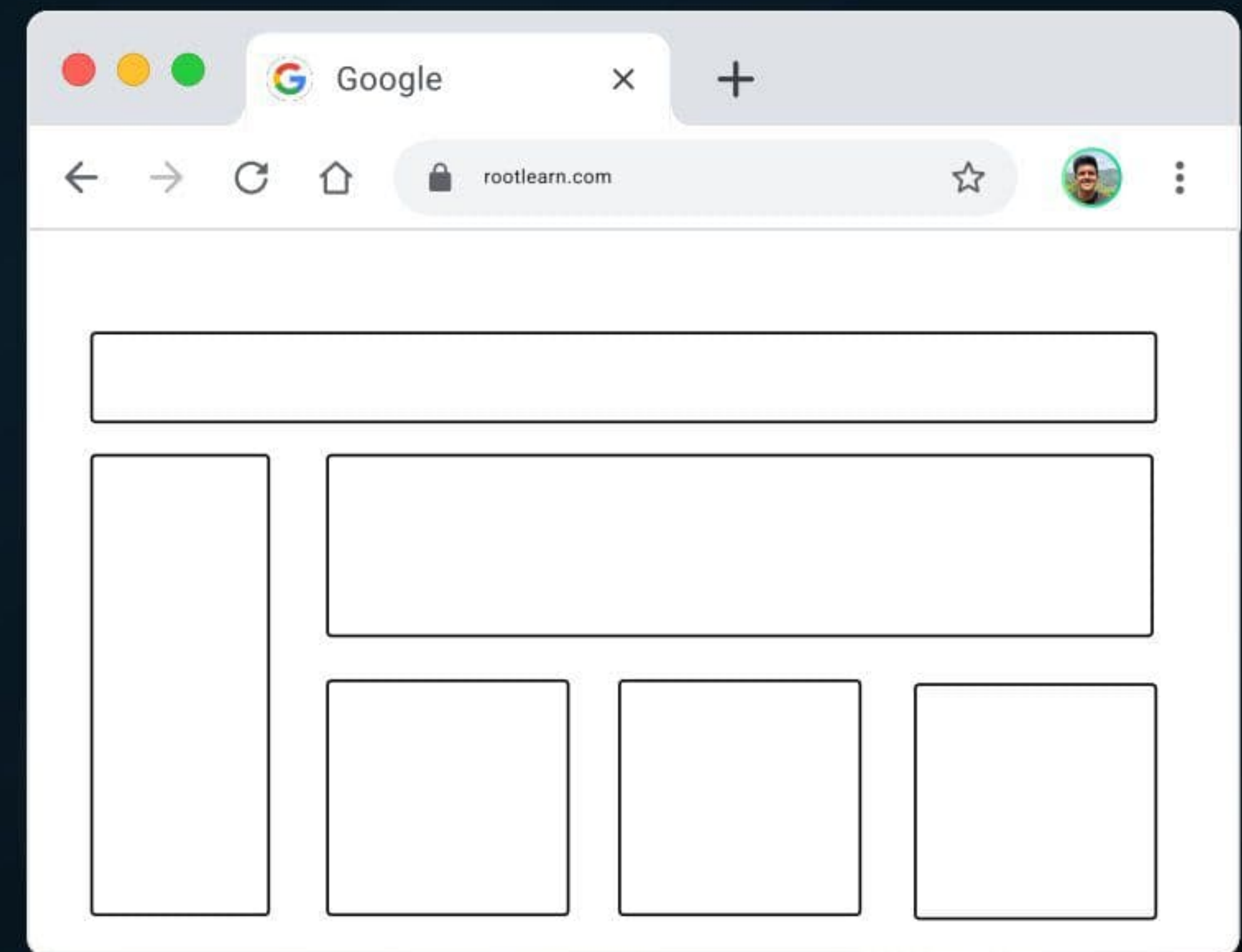
4. Layout

The browser then starts generating the layout of the page using the render tree

Render Tree



Layout



The browser is trying to figure out where to position elements

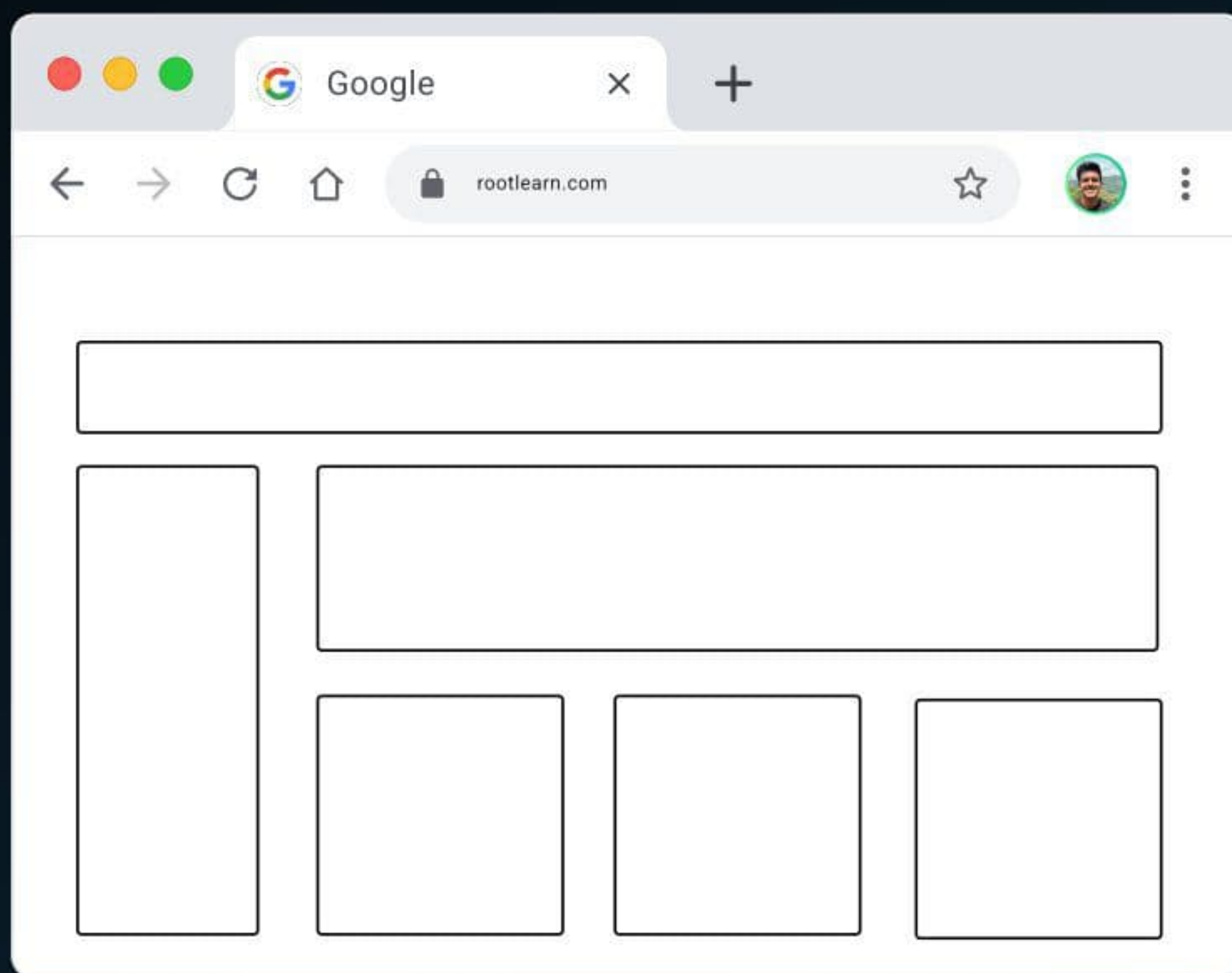




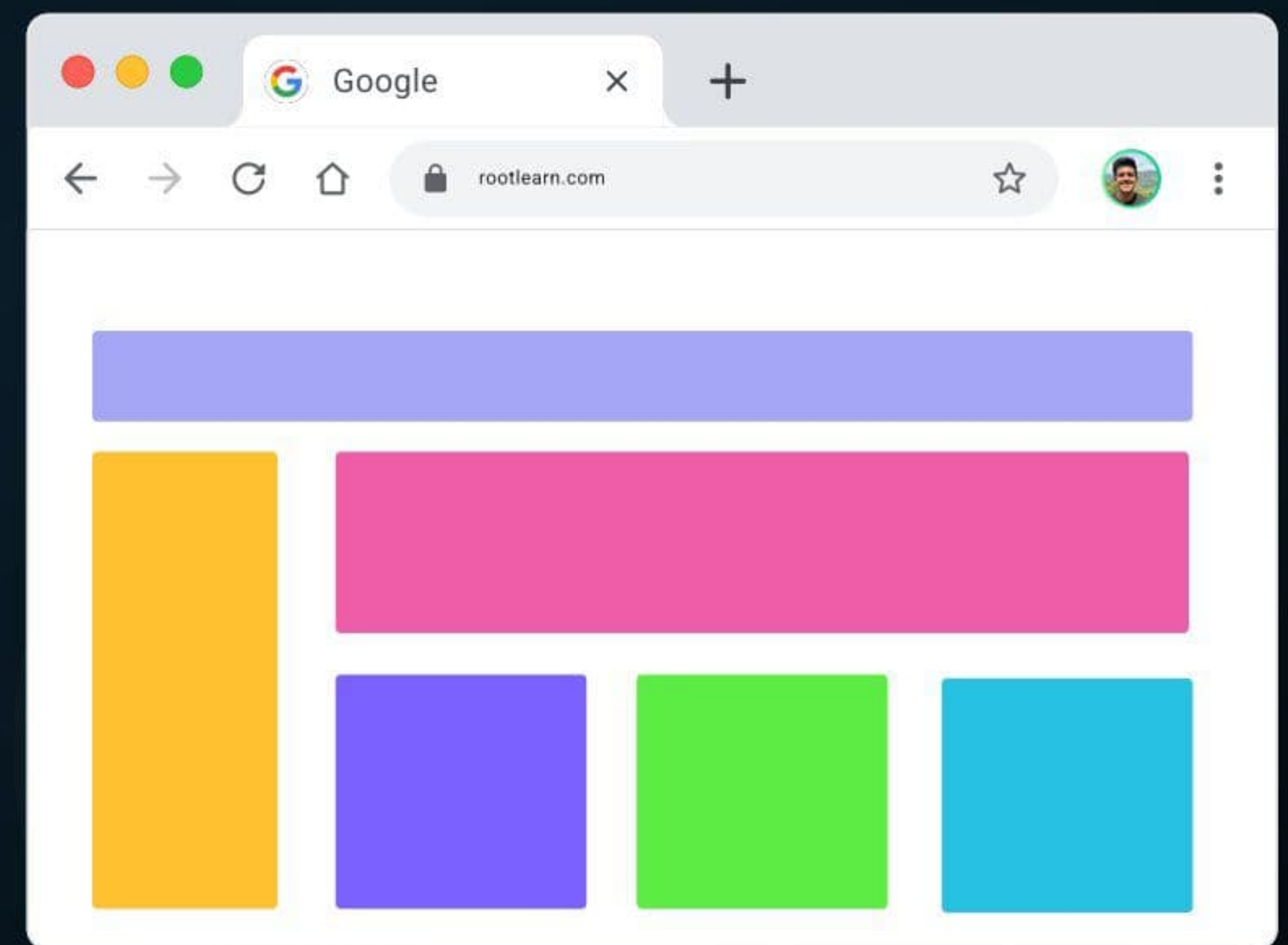
5. Paint

Once the layout is generated, the next step is to paint the pixels on the screen

Layout



Paint



PS: It also adds in text although it's not shown in the visuals



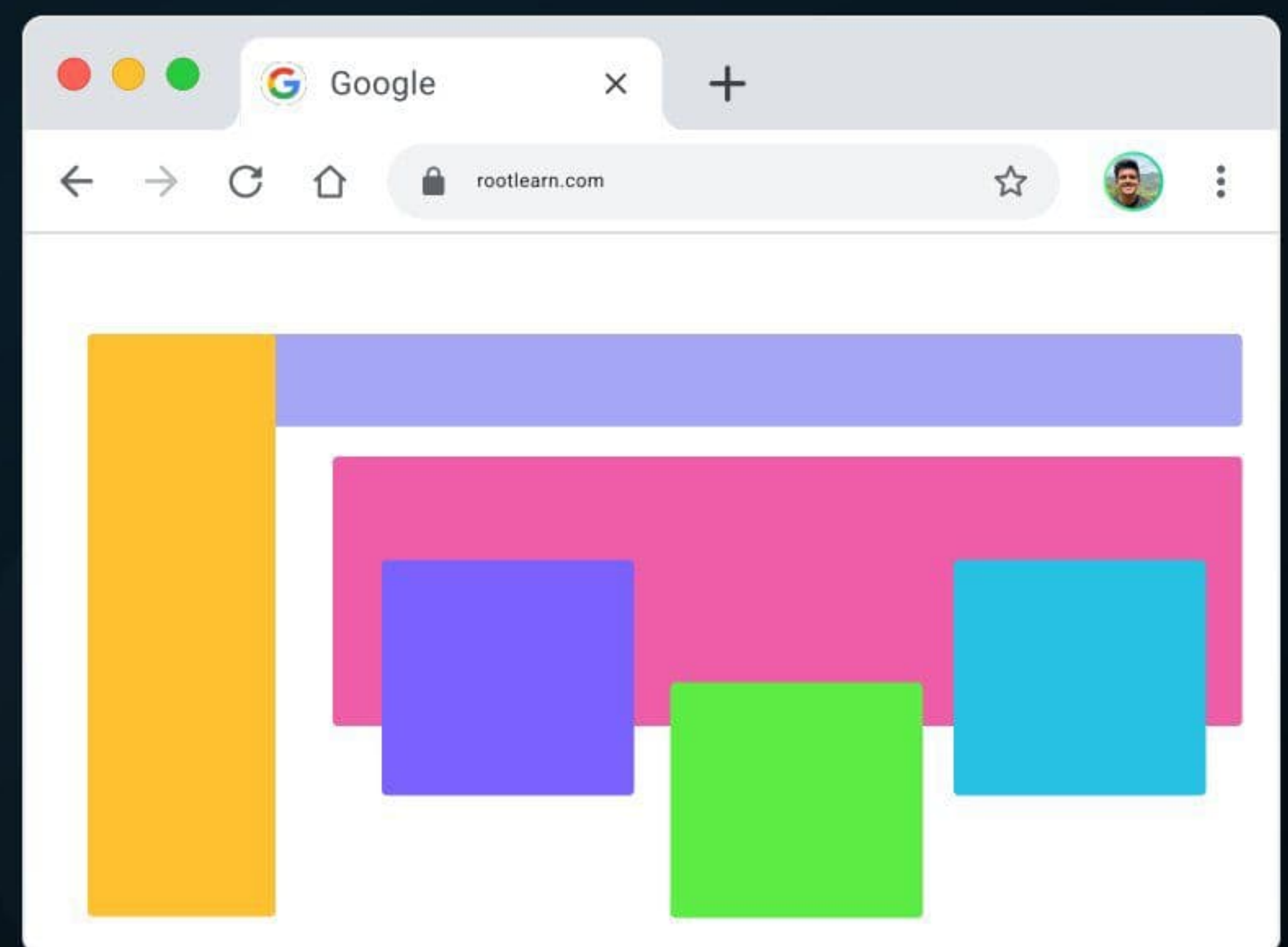
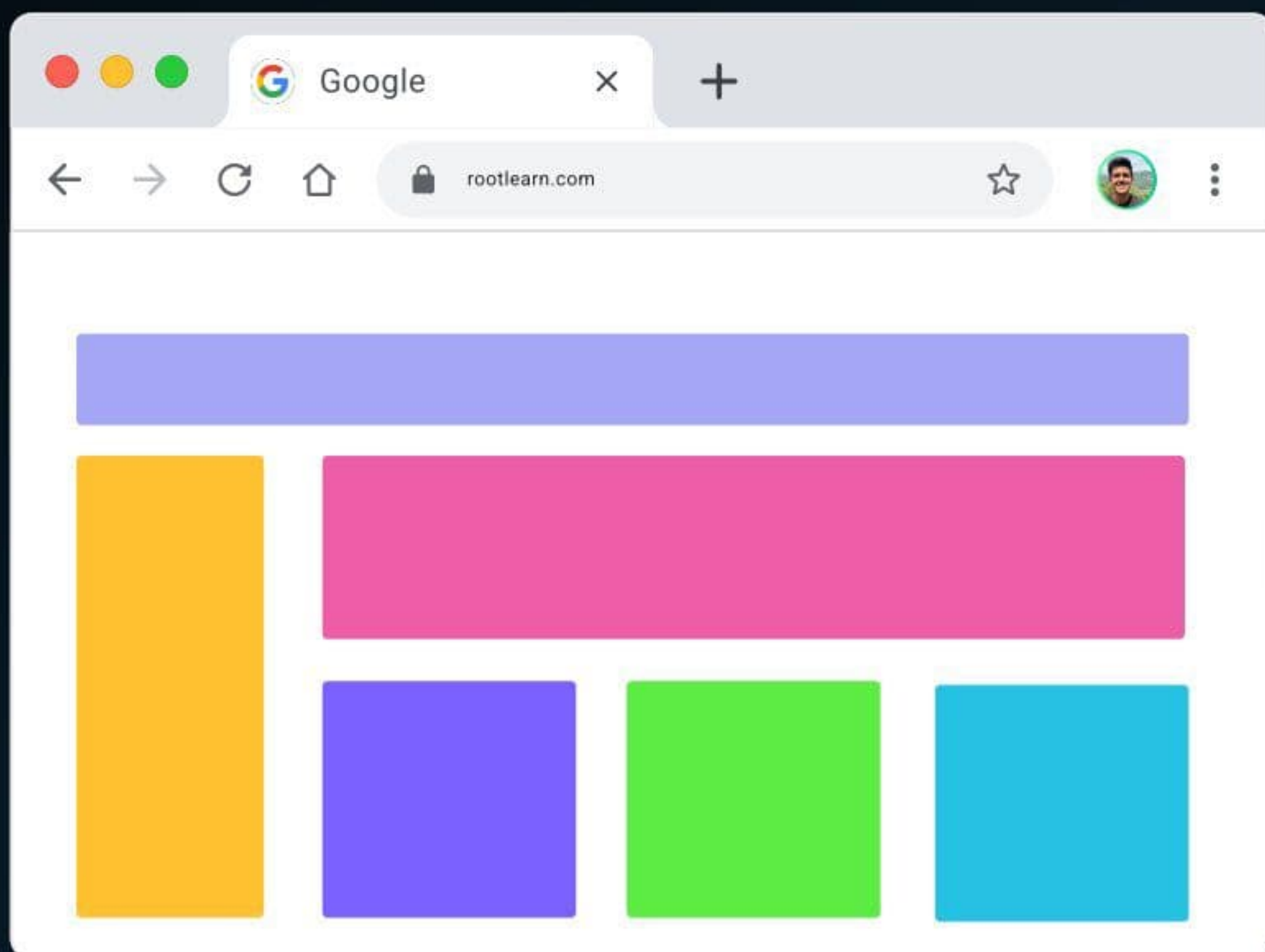


9/11

6. Compositing

In this step the browser builds layers on the screen depending on how elements need to appear (above/below some other elements)

Things like **position: absolute**





10/11

To summarize it again, here's what happens from the point the browser receives the data till it renders on the page

