Creating virtual scenes with A-Frame

A-Frame Course I

Section 2

Alberto Sanchez-Acedo

doi.org/XXX/zenodo.XXX

This work was supported by the Autonomous Community of Madrid (Spain) with a grant for industrial doctorates (IND2022/SOC-23503) with the collaboration agreement with Prodigioso Volcán S.L; Universidad Rey Juan Carlos (ID 501100007511).

Universidad Rey Juan Carlos · Prodigioso Volcán · COMCIENCIA · Comunidad de Madrid · Gobierno de España · Ministerio de Ciencia e Innovación · ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

VAR LAB

CIBERIMAGINARIO
GRUPO DE INVESTIGACIÓN

Prodigioso Volcán

CREATING VIRTUAL SCENCES WITH A-FRAME | A-FRAME COURSE I

In this course you will learn about A-Frame, an HTML-based framework that allows you to create Virtual Reality scenes and immersive web environments using the tool's own tags and components.

Universidad Rey Juan Carlos
Prodigioso Volcán
COMCIENCIA
Comunidad de Madrid
GOBIERNO DE ESPAÑA MINISTERIO DE CIENCIA E INNOVACIÓN
ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Section 2

A-Frame. Introducing the platform

SECTION OBJECTIVES

- Introducing A-Frame
- Unterstanding HTML language
- Learn how to search for basic documentation on the A-Frame website
- Start working on a virtual project

Sections points

1 2 3 4

Section points

Click on each of the points to find out what we will be studying in this module.

1. Theoretical introduction A-Frame

Developed by Mozilla, A-Frame is an open source domain-specific language (DSL) for generating Virtual Reality content in the browser (Gill, 2017). A-Frame is an HTML-based framework that allows generating virtual environments and WebXR scenes by using HTML tags and A-Frame's own components.

The use of HTML tags allows to add elements such as basic geometries (box, sphere, plane...) to the virtual scene, as well as other components that we will see below.

In addition, we will explain how HTML tags are composed (tag, attribute, value) and how they are used to give properties to the elements and geometries of our virtual scene. Among these properties, you can work with the color, size, position or rotation of the elements.

A-Frame can be developed from an HTML file without having to install anything. In this course we will use the Glitch platform (https://glitch.com/), an online code editor that is hosted and deployed instantly and free of charge.

EXAMPLES DEVELOPED WITH A-FRAME

To add new elements and more properties to components, JavaScript can be used. However, this course will not cover the use of JavaScript for the creation of basic virtual scenes.

LONG
COVID

FERNANDO

WELCOME!!!

2. HTML language

2. 1. HTML structure

To develop our virtual scene we will use lines of HTML code (tag language).

The basic structure of an HTML code consists of the following tags:

<!DOCTYPE html> **<html></html>** **<head></head>** **<body></body>**

HTML tags are composed of a name (html, body, head...) enclosed in angle brackets ("<>" and "</>").

The tags must always maintain the same order in the scene structure (as shown in the example). Between the opening and closing tags the necessary content (also in the form of tags) will be added to create the scene.

BASIC A-FRAME SCENE

HTML tags are written with an opening and a closing tag.

Opening tags	Closing tags
<html>	</html>
<head>	</head>
<body>	</body>

```
<!DOCTYPE html>
<html>
    <head>
        <script src="https://aframe.io/releases/1.5.0/aframe.min.js"></script>
    </head>
    <body>
        <a-scene>
            <a-box position="-1 0.5 -3" rotation="0 45 0" color="#4CC3D9"></a-box>
            <a-sphere position="0 1.25 -5" radius="1.25" color="#EF2D5E"></a-
            sphere>
            <a-cylinder position="1 0.75 -3" radius="0.5" height="1.5"
            color="#FFC65D"></a-cylinder>
            <a-plane position="0 0 -4" rotation="-90 0 0" width="4" height="4"
            color="#7BC8A4"></a-plane>
            <a-sky color="#ECECEC"></a-sky>
        </a-scene>
    </body>
</html>
```

```
<!DOCTYPE html>
<html>
    <head>
        <script src="https://aframe.io/releases/1.5.0/aframe.min.js"></script>
    </head>
    <body>
        <a-scene>
            <a-box position="-1 0.5 -3" rotation="0 45 0" color="#4CC3D9"></a-box>
            <a-sphere position="0 1.25 -5" radius="1.25" color="#EF2D5E"></a-
            sphere>
            <a-cylinder position="1 0.75 -3" radius="0.5" height="1.5"
            color="#FFC65D"></a-cylinder>
            <a-plane position="0 0 -4" rotation="-90 0 0" width="4" height="4"
            color="#7BC8A4"></a-plane>
            <a-sky color="#ECECEC"></a-sky>
        </a-scene>
    </body>
</html>
```

```
<!DOCTYPE html>
<html>
    <head>
        <script src="https://aframe.io/releases/1.5.0/aframe.min.js"></script>
    </head>
    <body>
        <a-scene>
            <a-box position="-1 0.5 -3" rotation="0 45 0" color="#4CC3D9"></a-box>
            <a-sphere position="0 1.25 -5" radius="1.25" color="#EF2D5E"></a-
            sphere>
            <a-cylinder position="1 0.75 -3" radius="0.5" height="1.5"
            color="#FFC65D"></a-cylinder>
            <a-plane position="0 0 -4" rotation="-90 0 0" width="4" height="4"
            color="#7BC8A4"></a-plane>
            <a-sky color="#ECECEC"></a-sky>
        </a-scene>
    </body>
</html>
```

```
<!DOCTYPE html>
<html>
    <head>
        <script src="https://aframe.io/releases/1.5.0/aframe.min.js"></script>
    </head>
    <body>
        <a-scene>
            <a-box position="-1 0.5 -3" rotation="0 45 0" color="#4CC3D9"></a-box>
            <a-sphere position="0 1.25 -5" radius="1.25" color="#EF2D5E"></a-
            sphere>
            <a-cylinder position="1 0.75 -3" radius="0.5" height="1.5"
            color="#FFC65D"></a-cylinder>
            <a-plane position="0 0 -4" rotation="-90 0 0" width="4" height="4"
            color="#7BC8A4"></a-plane>
            <a-sky color="#ECECEC"></a-sky>
        </a-scene>
    </body>
</html>
```

Click on each of the tag icons to learn the basic HTML code.

The code shows the default basic scene presented by A-Frame which you can access by clicking on the A-Frame Basic Scene icon.

This scene shows several a-frame elements that are basic geometries (sphere, box, plane).

2. HTML language

2. 1. 1. <!DOCTYPE html>

This tag is the initial tag of an HTML document that tells the browser the version you are working with and is necessary to correctly process the document's code.

2. 1. 2. <html> tag

Starts the HTML document and collects all the content that makes up a web page.

Within the <html> tag, the <head> tag and the <body> tag must appear.

<html>...</html>

```
<!DOCTYPE html>
<html>
    <head>
        <script src="https://aframe.io/releases/1.5.0/aframe.min.js"></script>
    </head>
    <body>
        <a-scene>
            <a-box position="-1 0.5 -3" rotation="0 45 0" color="#4CC3D9"></a-box>
            <a-sphere position="0 1.25 -5" radius="1.25" color="#EF2D5E"></a-
            sphere>
            <a-cylinder position="1 0.75 -3" radius="0.5" height="1.5"
            color="#FFC65D"></a-cylinder>
            <a-plane position="0 0 -4" rotation="-90 0 0" width="4" height="4"
            color="#7BC8A4"></a-plane>
            <a-sky color="#ECECEC"></a-sky>
        </a-scene>
    </body>
</html>
```

2. HTML language

2. 1. 3. <head> tag

The <head> tag contains the lines of code necessary for our web page to function and which are not displayed visually. It contains metadata such as the title of the page, links to style sheets, scripts, etc.

For the elaboration of an a-frame environment, it is necessary to add inside <head> the necessary script to inform the browser that we are working in a virtual a-frame environment.

<head>...</head>

```
<!DOCTYPE html>
<html>
    <head>
        <script src="https://aframe.io/releases/1.5.0/aframe.min.js"></script>
    </head>
    <body>
        <a-scene>
            <a-box position="-1 0.5 -3" rotation="0 45 0" color="#4CC3D9"></a-box>
            <a-sphere position="0 1.25 -5" radius="1.25" color="#EF2D5E"></a-
            sphere>
            <a-cylinder position="1 0.75 -3" radius="0.5" height="1.5"
            color="#FFC65D"></a-cylinder>
            <a-plane position="0 0 -4" rotation="-90 0 0" width="4" height="4"
            color="#7BC8A4"></a-plane>
            <a-sky color="#ECECEC"></a-sky>
        </a-scene>
    </body>
</html>
```

2. HTML language

2. 1. 4. <body> tag

Inside the <body> tag will be all the visible content on our website.

When designing an a-frame environment, it is necessary to add the <a-scene> tag inside <body>, as shown in the example:

<body>...</body>

Inside the <a-scene> tag will go all the elements that will appear in our virtual scene.

```
<!DOCTYPE html>
<html>
    <head>
        <script src="https://aframe.io/releases/1.5.0/aframe.min.js"></script>
    </head>
    <body>
        <a-scene>
            <a-box position="-1 0.5 -3" rotation="0 45 0" color="#4CC3D9"></a-box>
            <a-sphere position="0 1.25 -5" radius="1.25" color="#EF2D5E"></a-
            sphere>
            <a-cylinder position="1 0.75 -3" radius="0.5" height="1.5"
            color="#FFC65D"></a-cylinder>
            <a-plane position="0 0 -4" rotation="-90 0 0" width="4" height="4"
            color="#7BC8A4"></a-plane>
            <a-sky color="#ECECEC"></a-sky>
        </a-scene>
    </body>
</html>
```

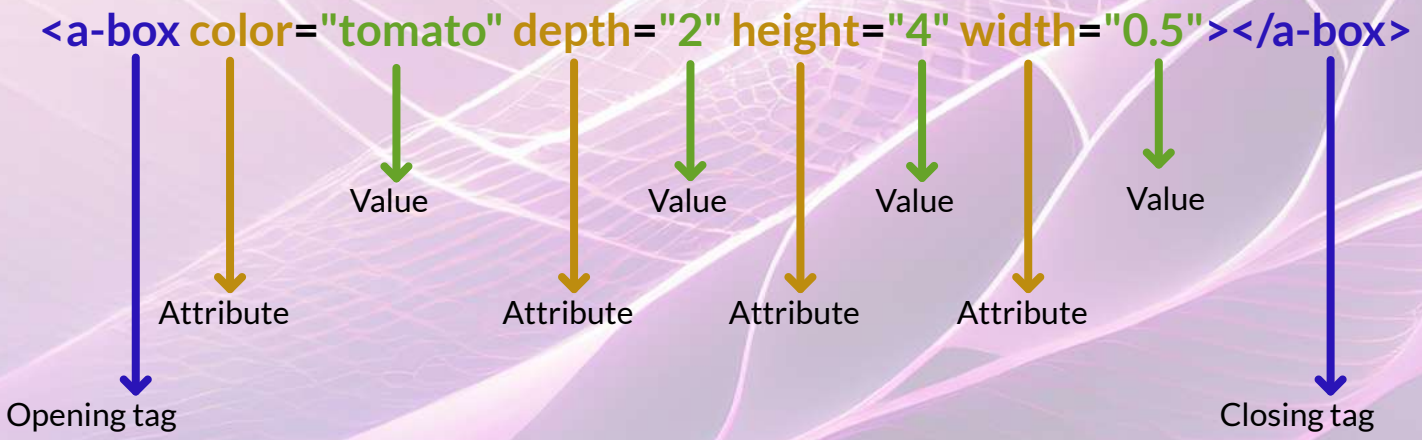
2. HTML language

2. 2. Composition HTML tags

We have already seen that html tags have an opening and a closing tag.

Within the opening tag you can add **attributes** to the a-frame elements themselves to give them a set of properties or **values**. The attributes are added before closing the opening tag with the symbol (>).

The following example is a line of code to add a box (a-box) to a virtual scene with attributes and their respective values:



The values (shown in green in the example) must always be enclosed in quotation marks.

Attributes are shown in yellow, values in green and labels in blue.

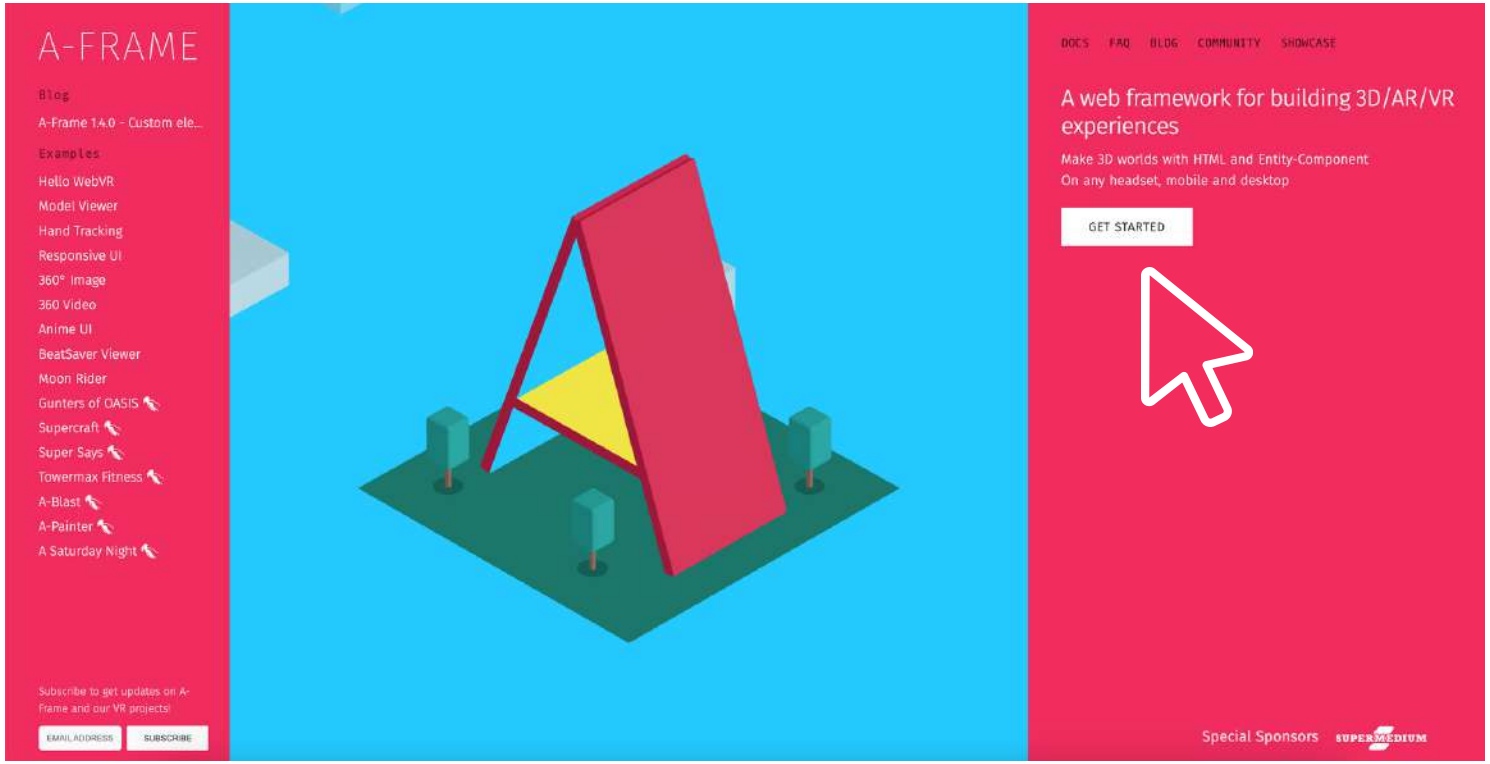
In this example all attributes have a value assigned to them, but in some cases, there are attributes that do not require a value to be assigned to them.

3. A-Frame. Get to know the platform

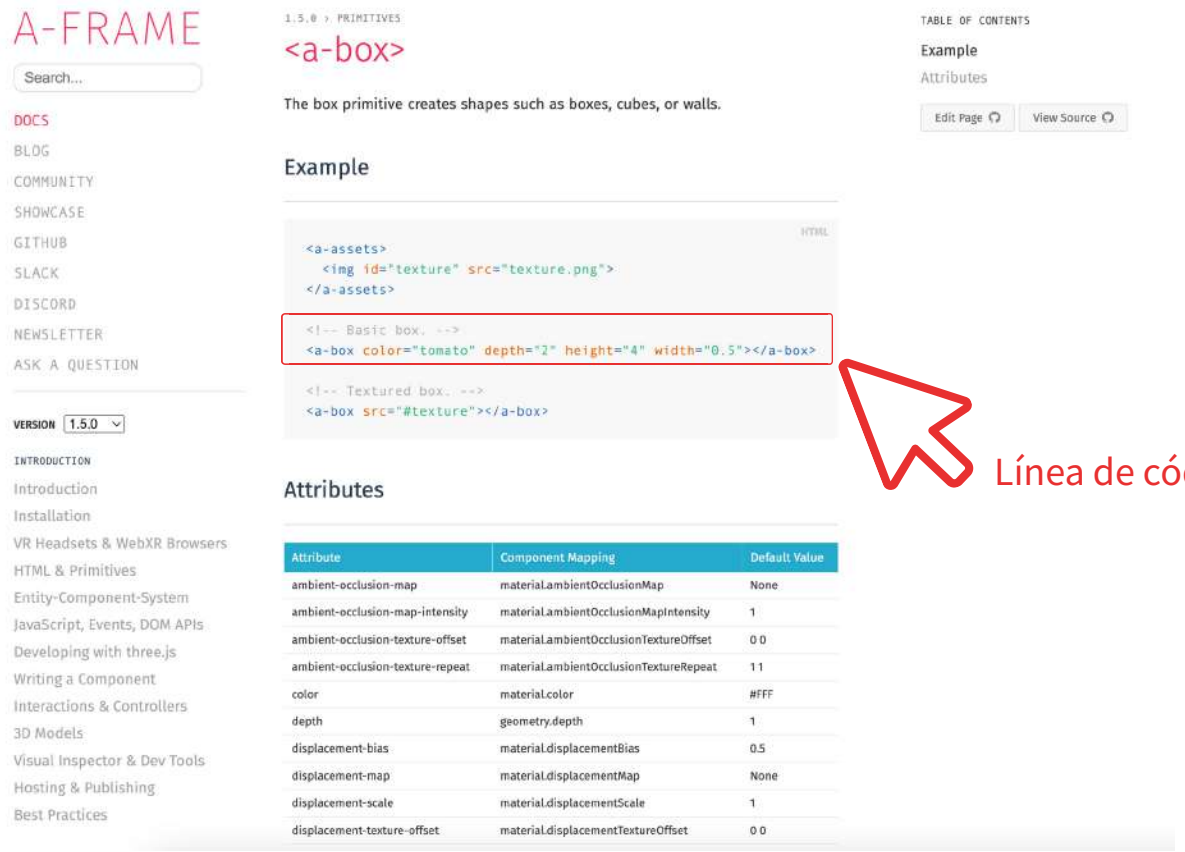
To start working with A-Frame it is necessary to have access to the platform (https://aframe.io/). In this web is where we are going to find all the necessary documentation to develop our basic scene.

1. Web access
2. Access to documentation
3. How to search for elements?
4. Example a-box
5. Basic a-box code line
6. a-box attributes

Once we enter the website, we have to click on "Get Started" to access all the documentation.

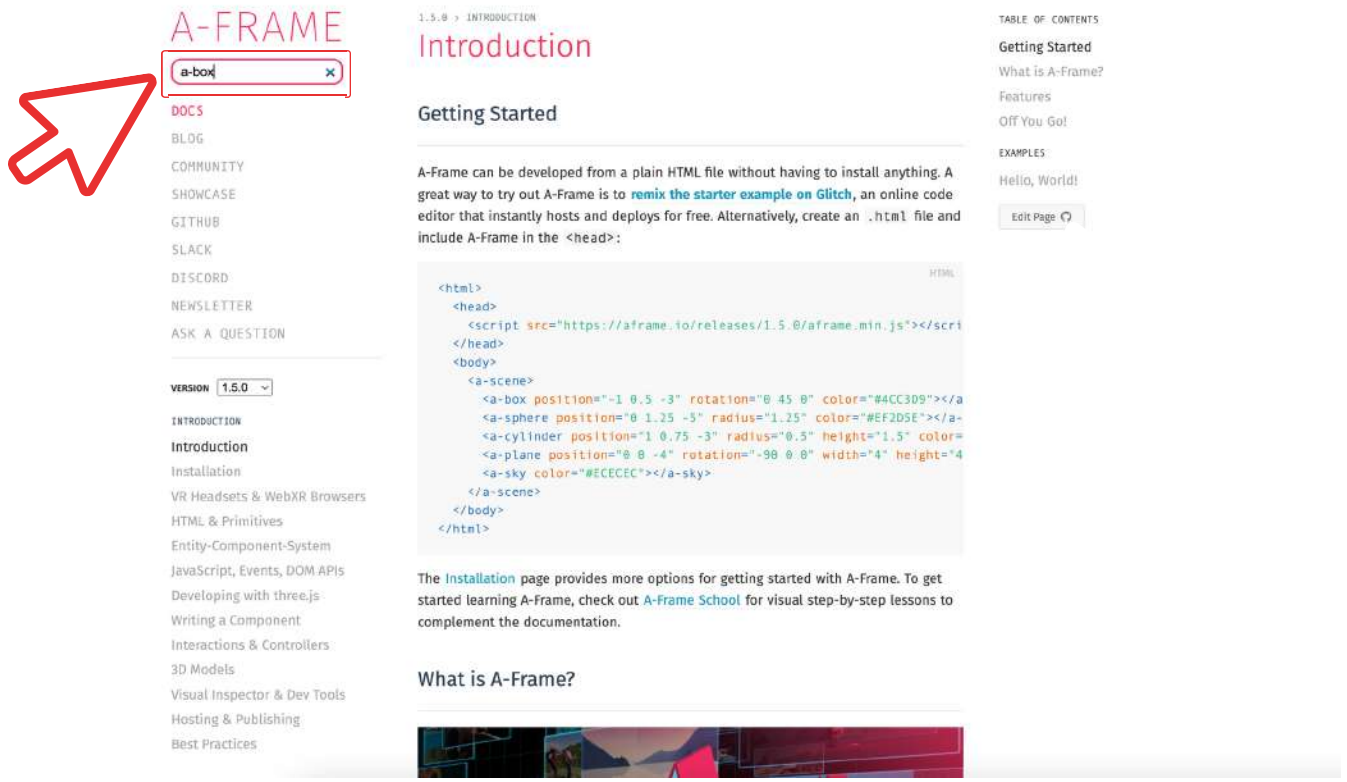


In this window we find all the documentation related to the box element.

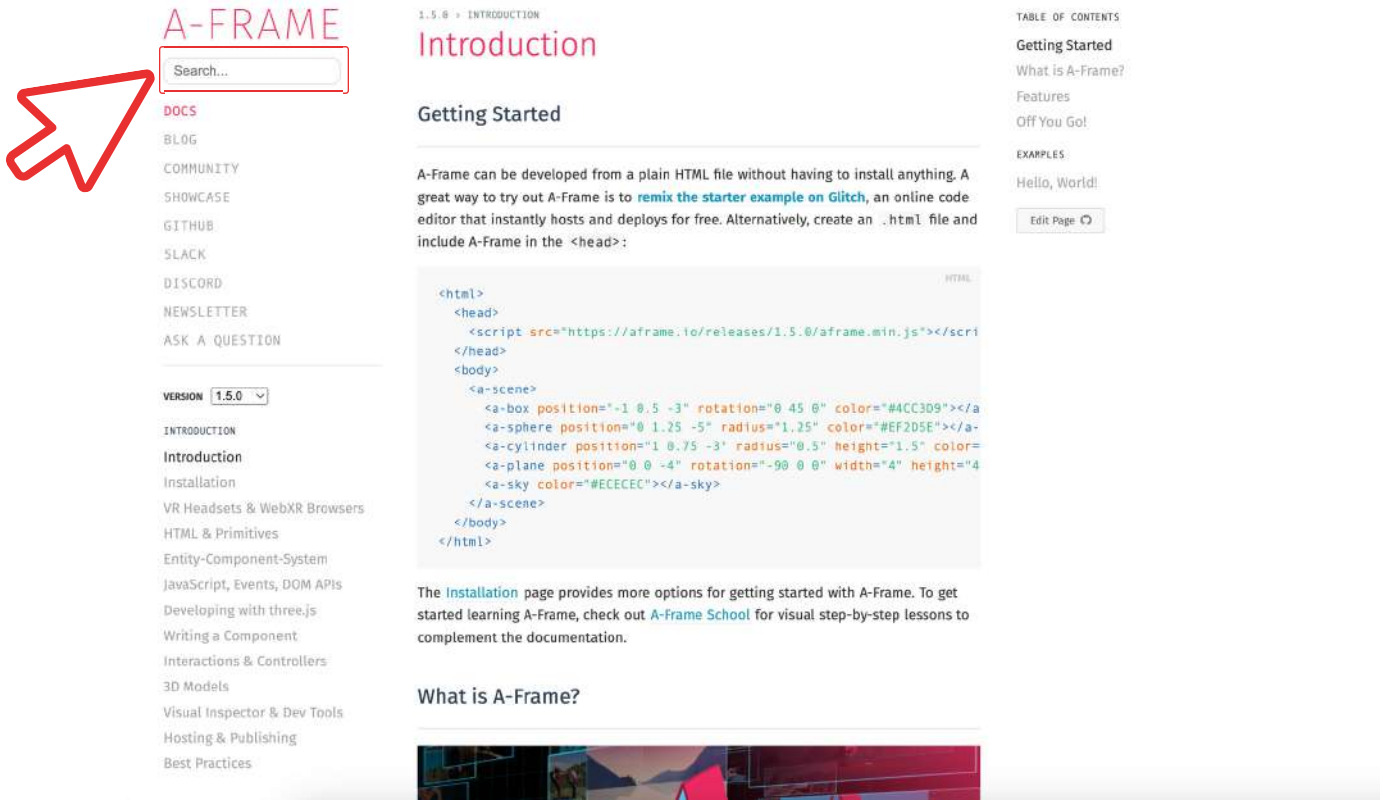


Línea de código básica para añadir la caja

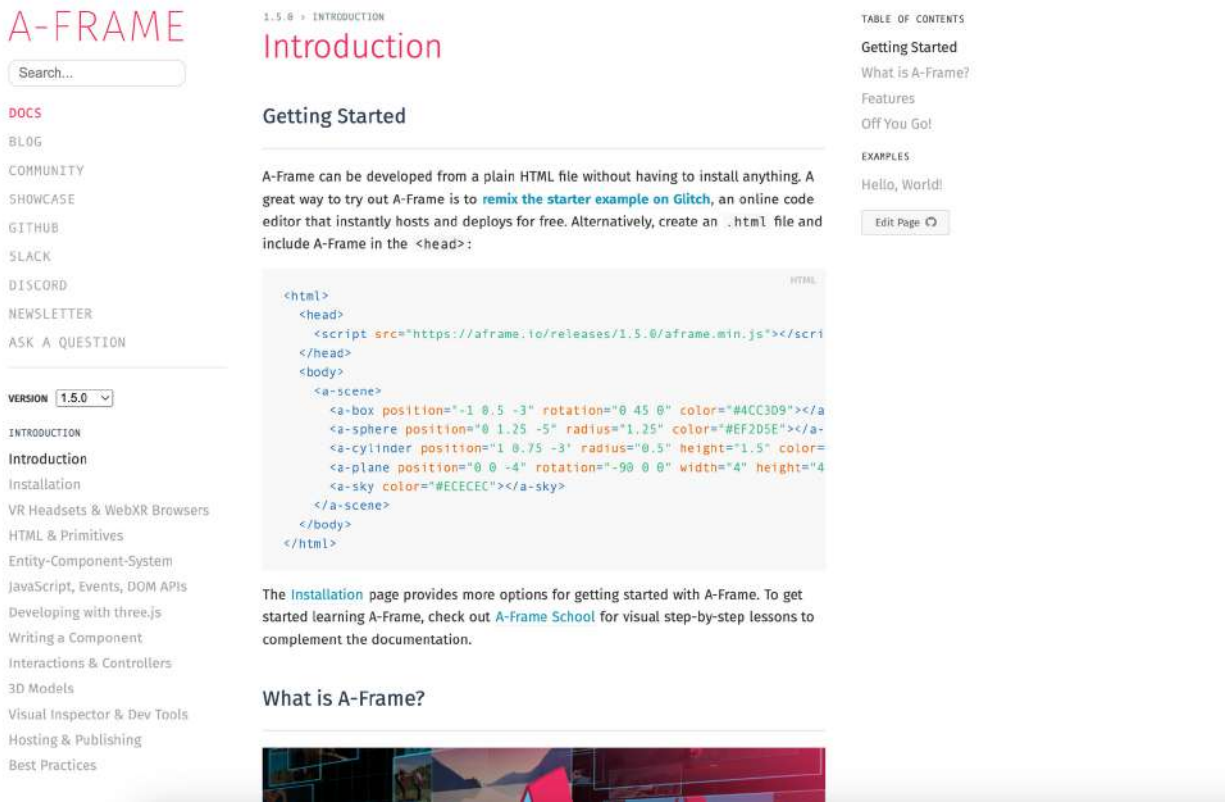
Let's see an example. We are going to search how to add a box (a-box) to our scene. We type in the search bar a-box (or simply box) to see all the documentation concerning the box element.



To find the lines of HTML code needed to integrate the different elements in our virtual scene we will search for them in the search bar.



We will access this interface, where we will find all the necessary documentation to develop our virtual scene.

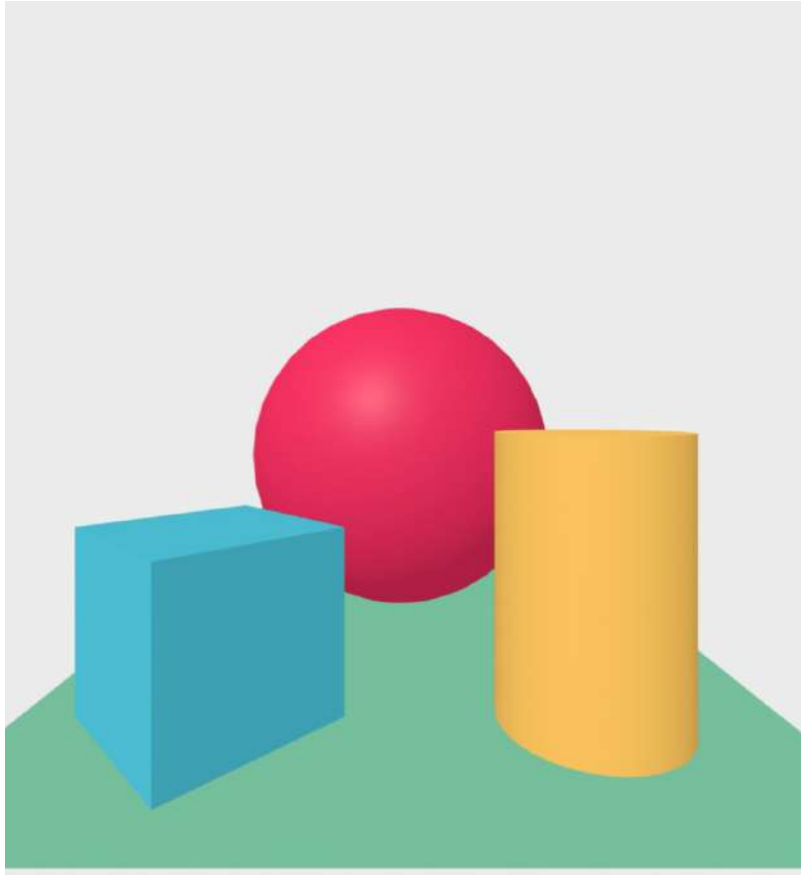


In this window we find all the documentation related to the box element.

En esta sección encontraremos todos los atributos que podemos añadir a los elementos (en este caso al elemento a-box)

Exercise

Creation of the first basic A-Frame scene with Glitch



In this exercise we will learn how to use the Glitch platform to create virtual scenes using A-Frame.

Utilizaremos el ejemplo básico que A-Frame proporciona.

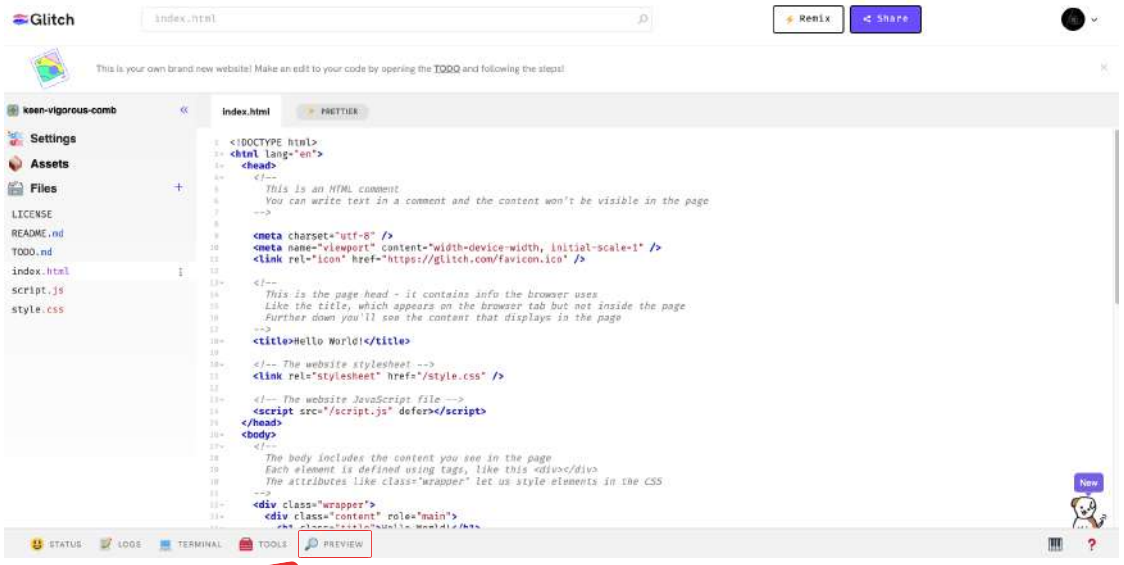
During the development of the course we will design step by step, in the practical exercises of each module, a virtual museum room to which we will integrate elements such as images, videos, 3D objects...

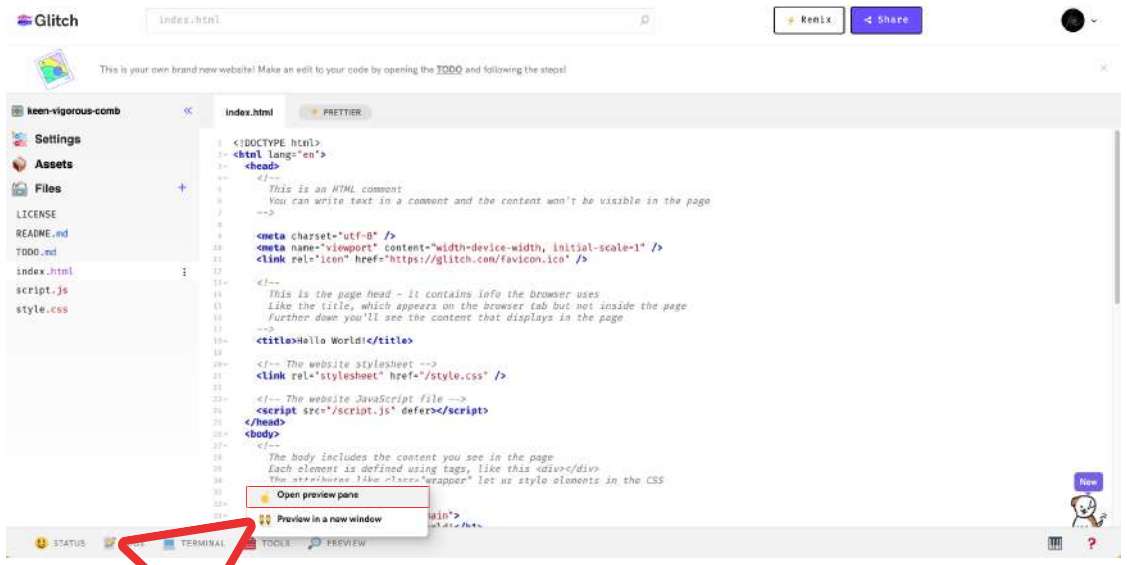
Start working

To develop our first virtual scene we will use the online platform Glitch (https://glitch.com/).

1. Registration
2. Creation of a project
3. Knowing the interface
4. Code preview
5. Preview pane
6. A-Frame Code
7. A-Frame code preview

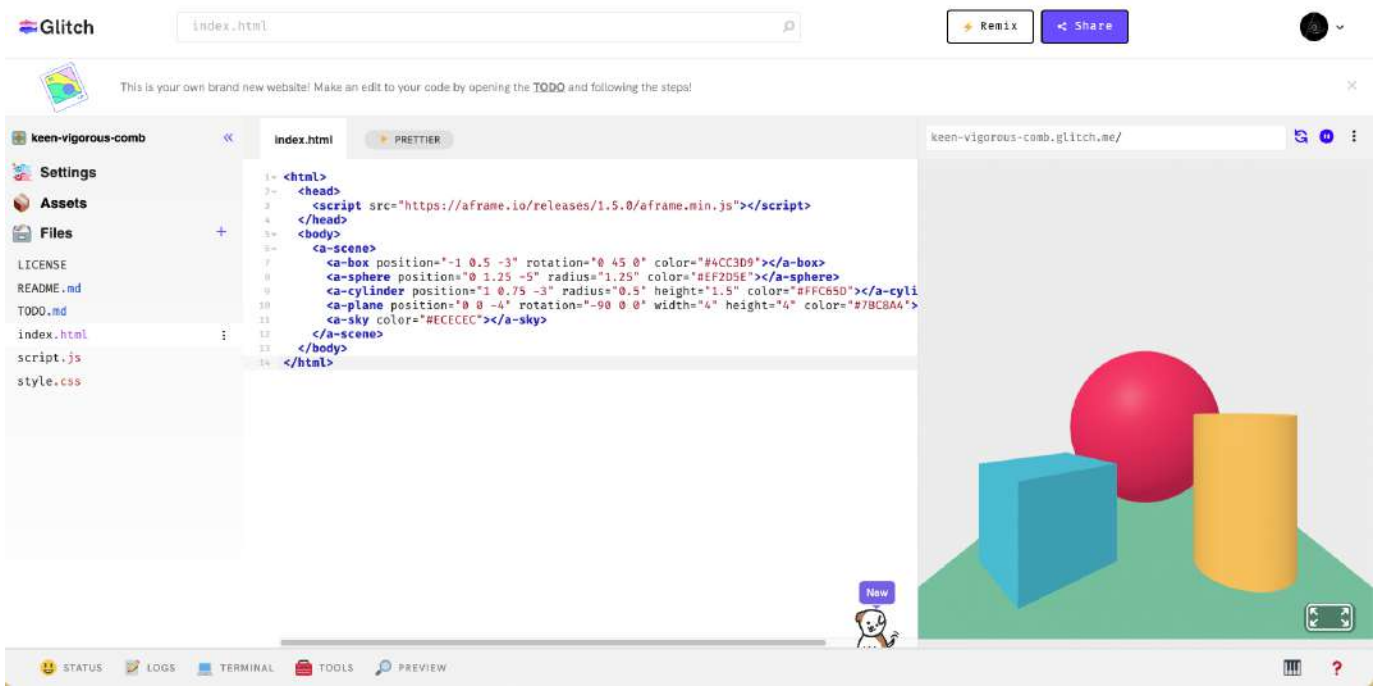
The index.html tab allows us to work with code and to be able to visualize what we are doing at the moment. To do this, we have to click on "Preview" and then on "Open preview pane"



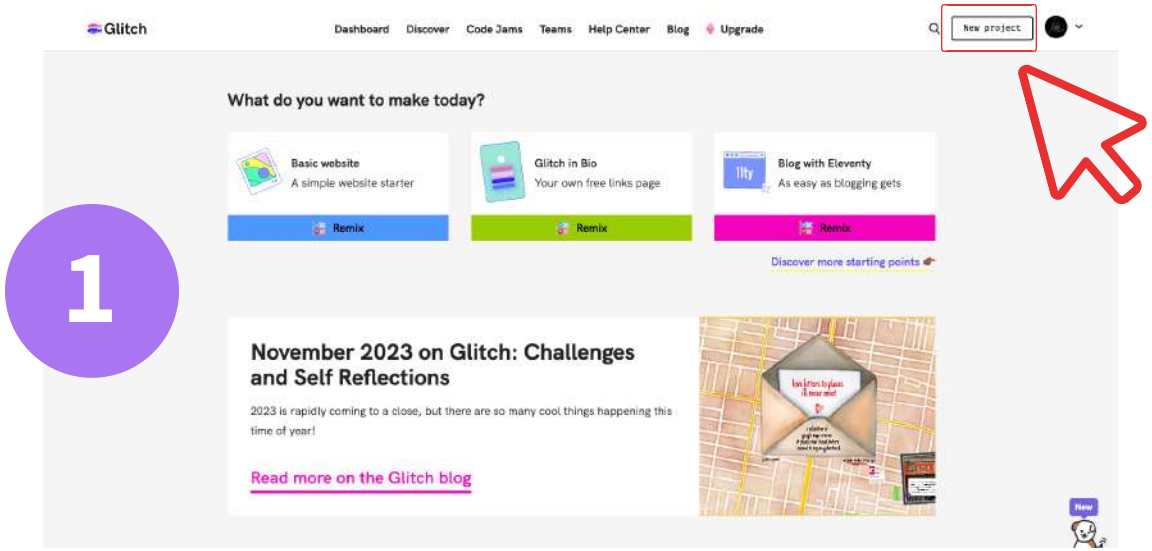


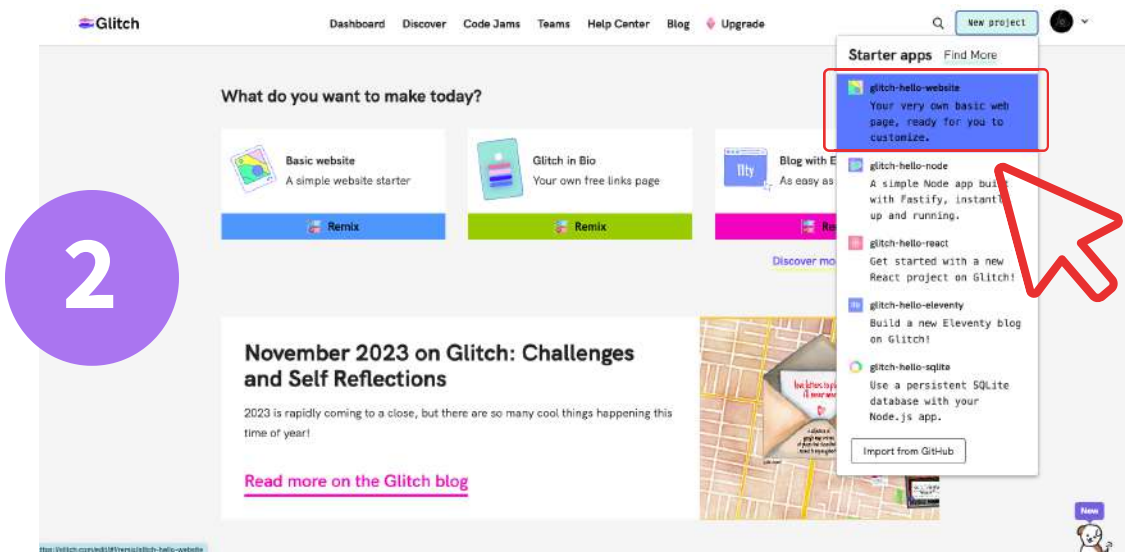
The code that we have copied, we will paste it in the project that we have created previously in glitch (in the tab index.html).

The result has to be the following:

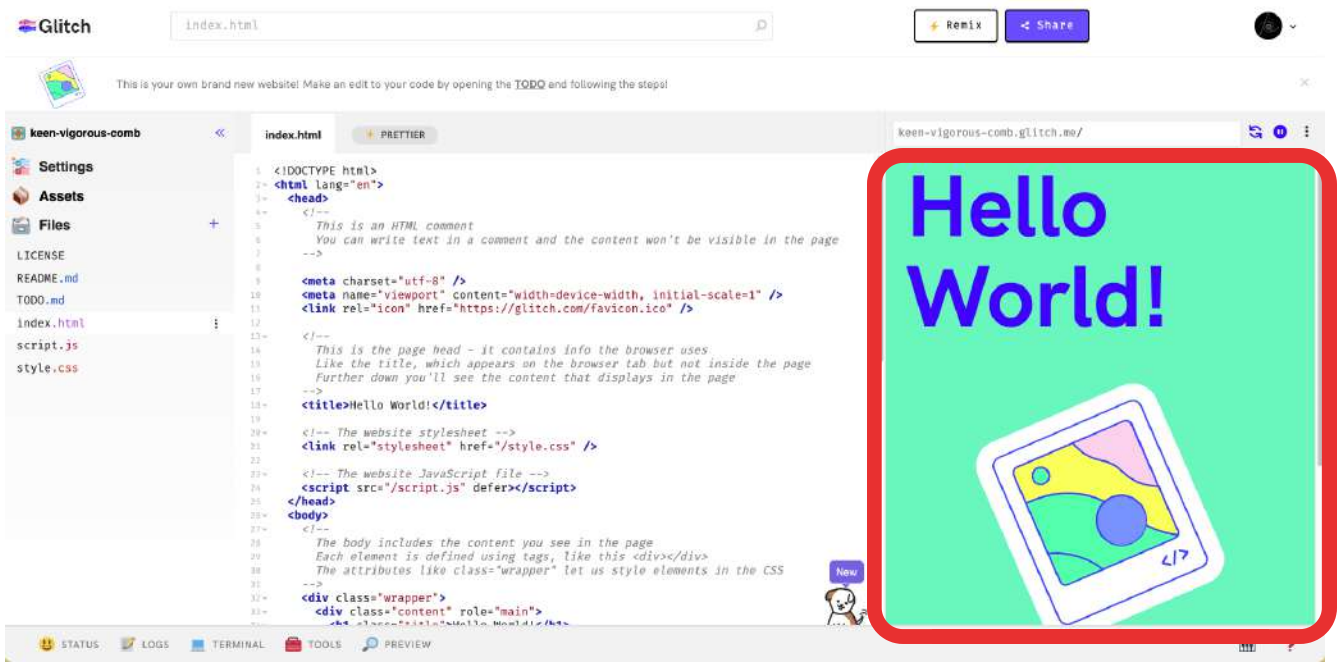


Once registered, to start working on a project we have to click on "New project" and then on "glitch-hello-website".

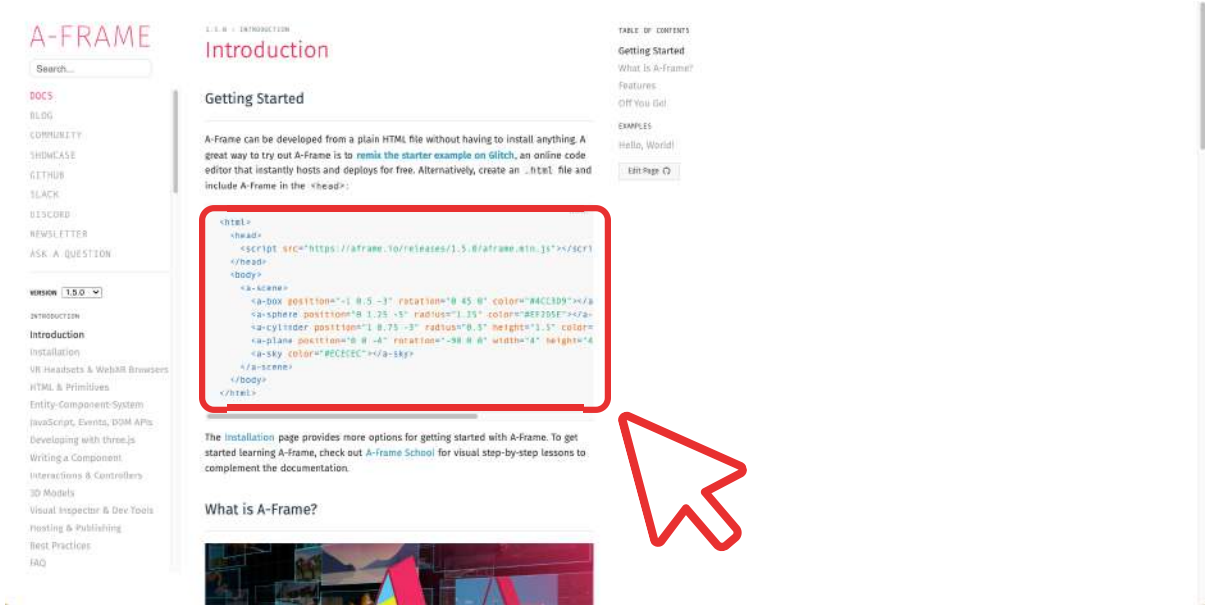




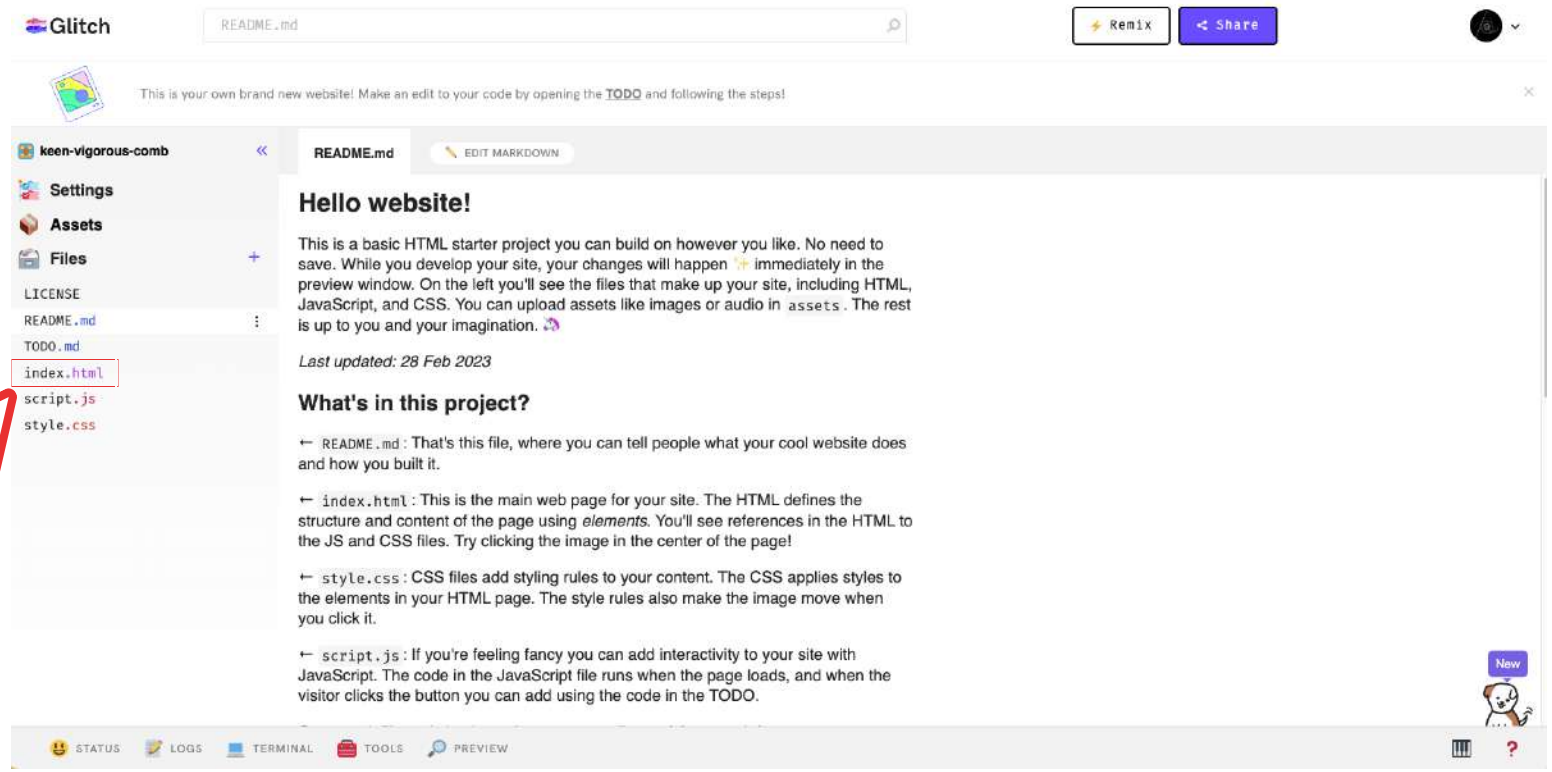
A preview tab of the code we are creating will appear on the right:



To start working on an A-Frame scene, we have to copy the default code provided by the A-Frame platform and paste it into our glitch project.



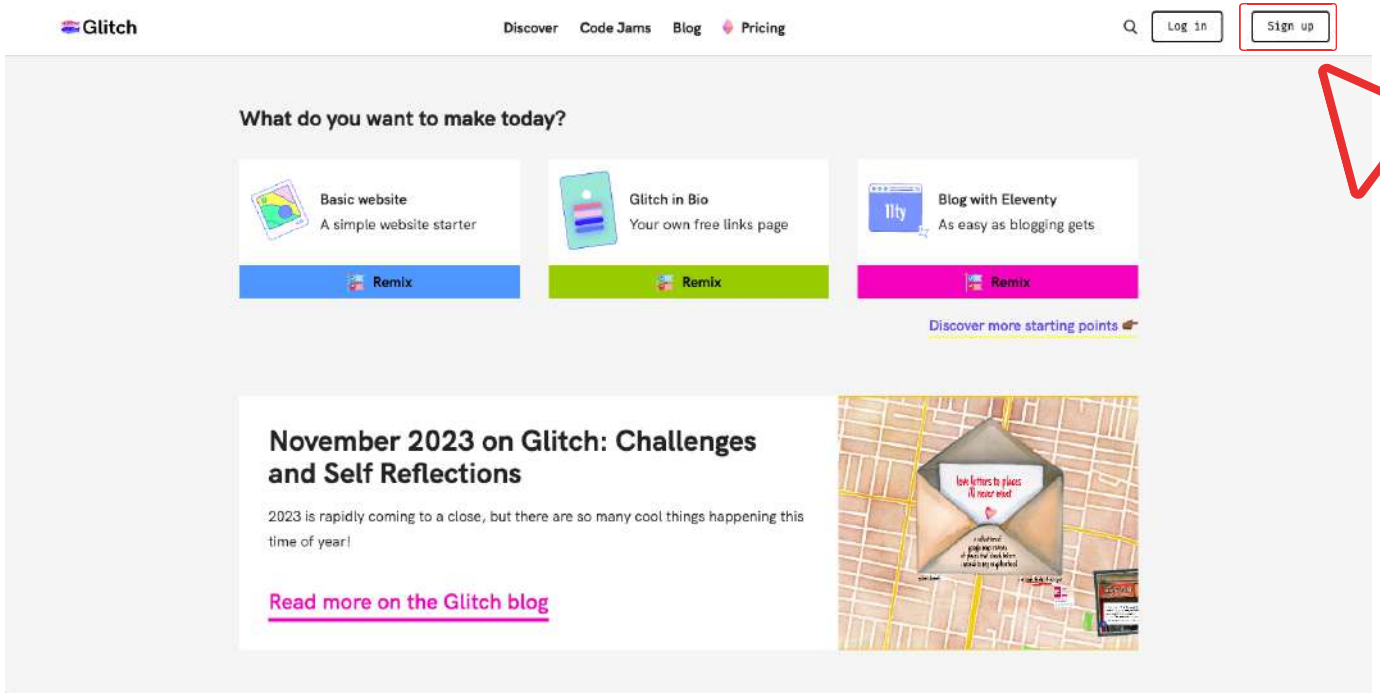
Once a new project has been created, the following interface will appear:



To start working with code we have to click on index.html

(As the course progresses we will discover what the rest of the tabs are).

Once we enter Glitch.com, we have to create an account in "Sign up" to start working.



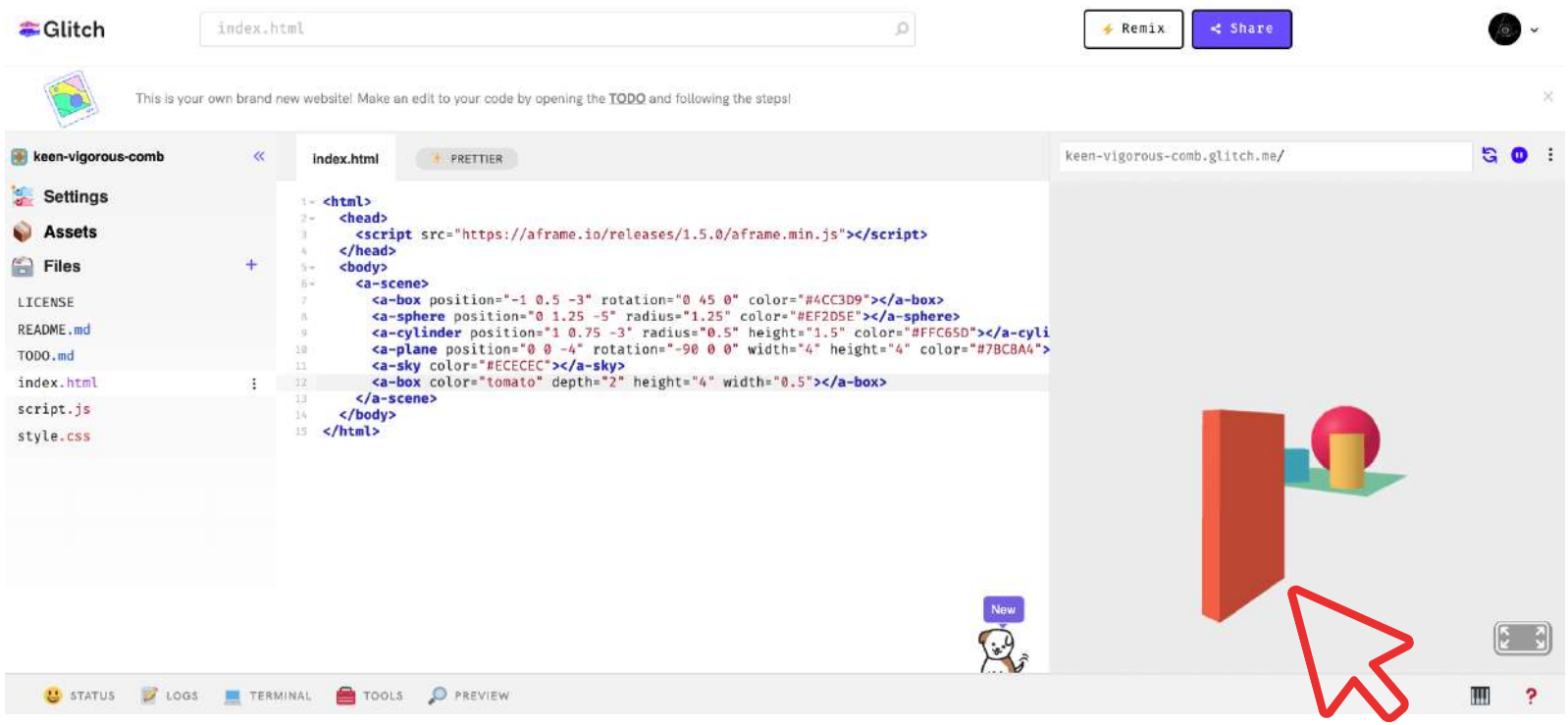
Add your first box

In the theoretical part we have explained how to add a basic box, let's put it into practice:

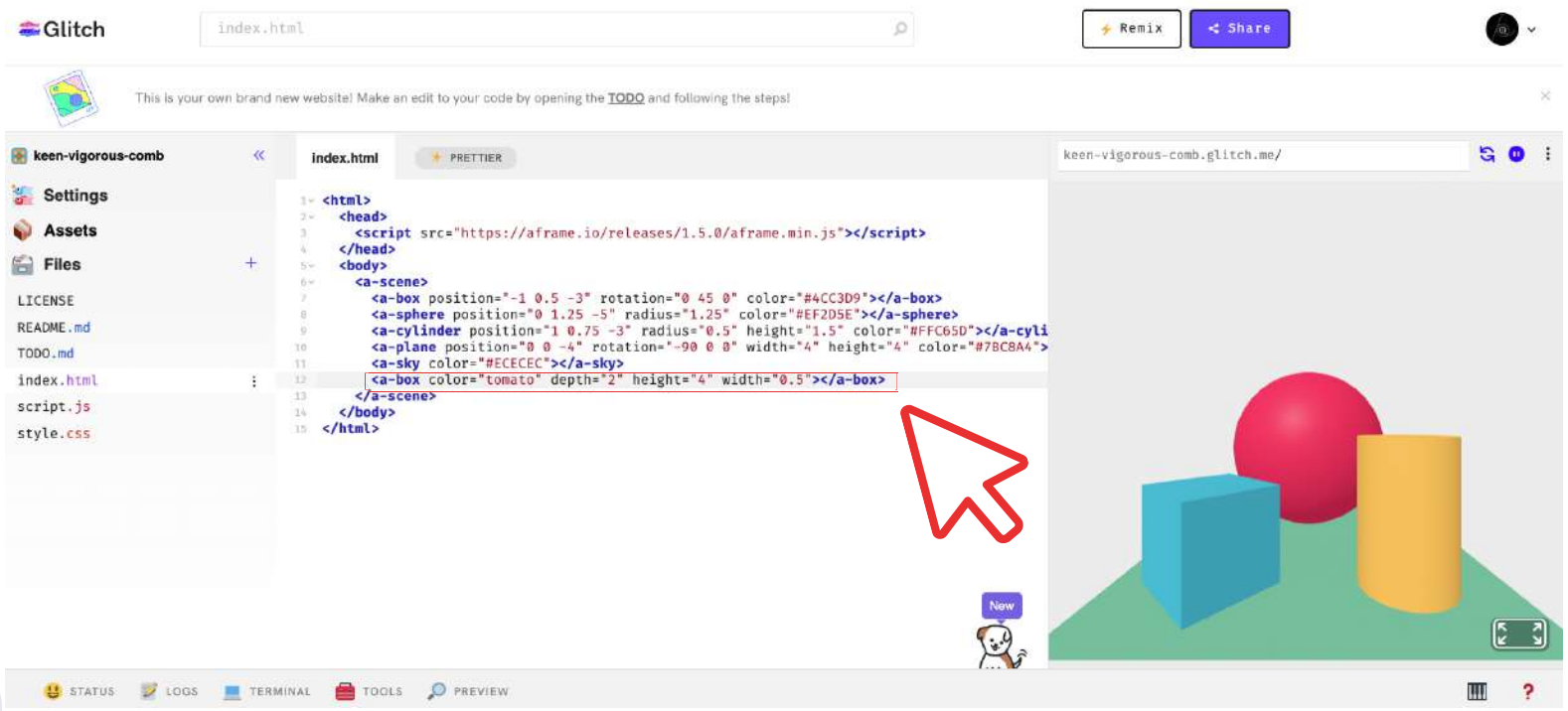
1. a-box example
2. Basic a-box code line
3. Integrating a-box element in Glitch
4. a-box visualization added
5. a-box attributes
6. Modifying a-box attributes
7. Final results

To see the box we have just added, we can navigate the environment with the arrow keys and the mouse.

The default box is displayed as shown in the image:



Once copied the line of code, we will paste it in our project in Glitch. Always inside the <a-scene> tag, as shown in the image:



1 **We will work with the attributes of color, position, width, depth and height:**

```
<!-- Basic box. -->
<a-box color="tomato" depth="2" height="4" width="0.5"></a-box>
```

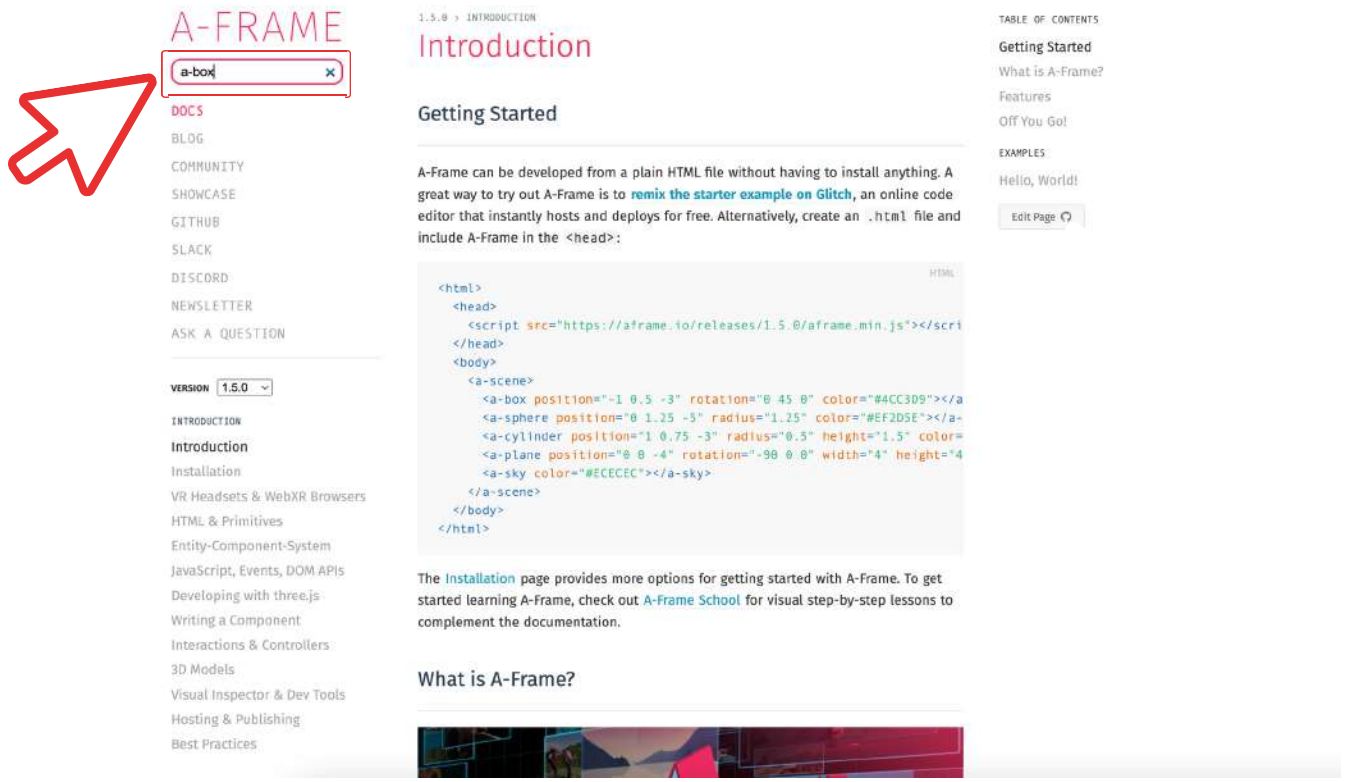
2 **We will change the color to a hexadecimal color and the proportions of the box, as shown below:**

```
<!-- Caja cambiando atributos -->
<a-box color="#572634" depth="3" height="3" width="3"></a-box>
```

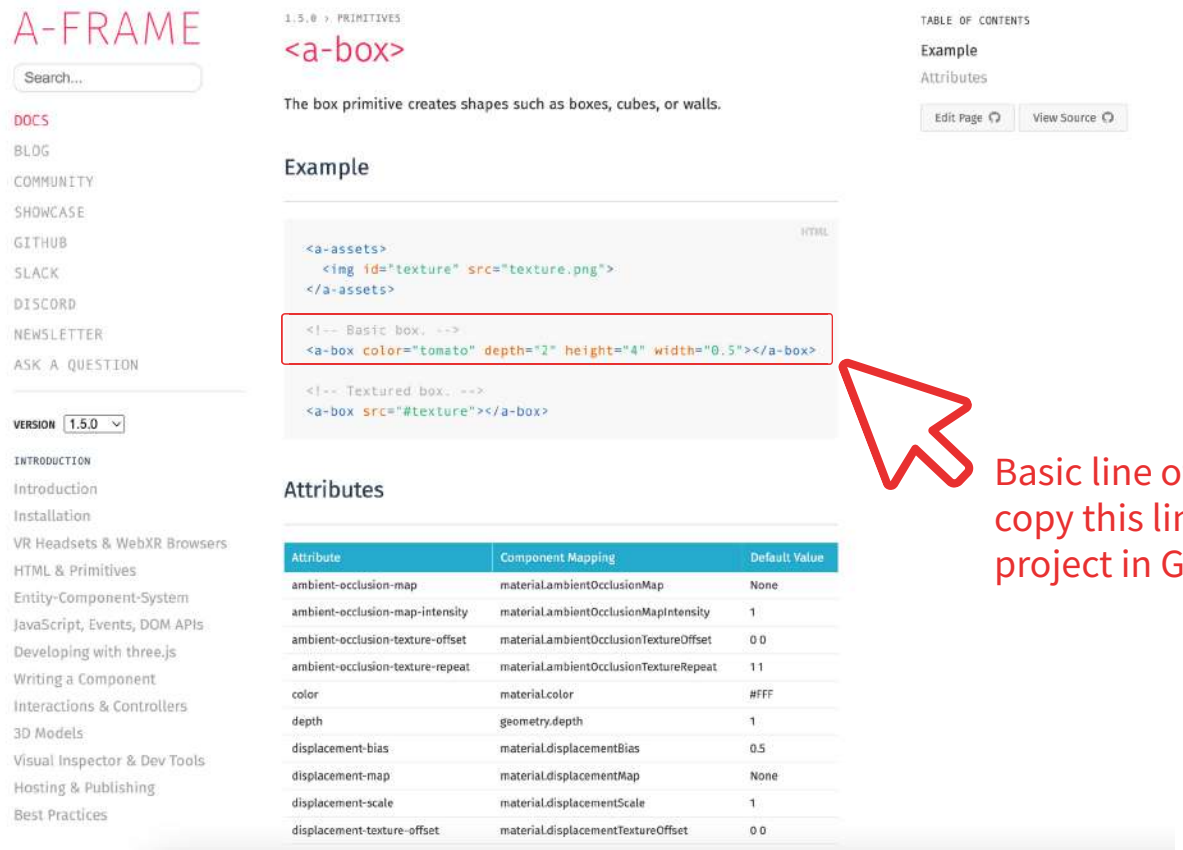
3 **We add the "position" attribute to our line of code:**

```
<!-- Caja cambiando atributos -->
<a-box color="#572634" depth="1" height="1" width="1" position="0 2.5 -2"></a-box>
```

To see the basic line of code of a box (a-box) we type in the search bar "a-box" (or simply box) to see all the documentation concerning the box element.

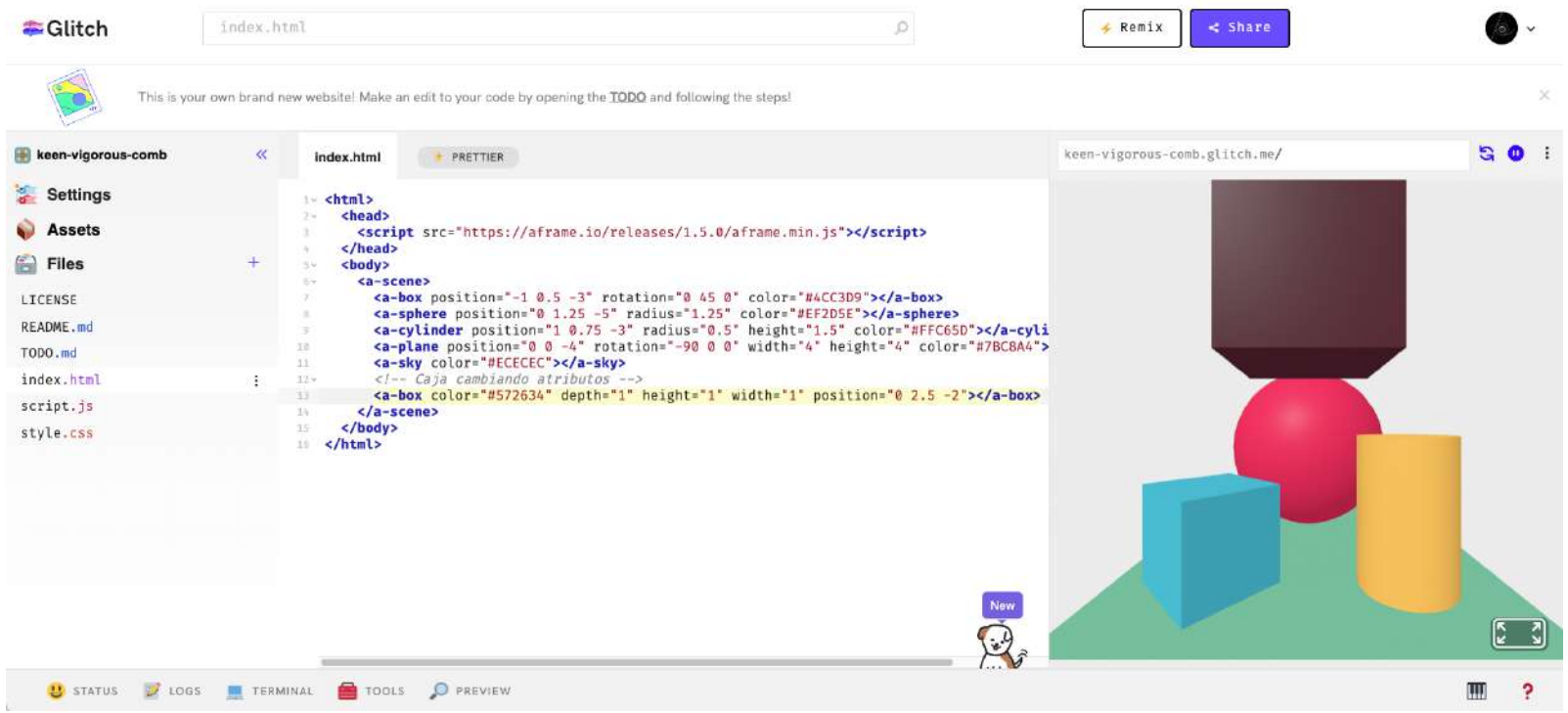


In this window we find all the documentation related to the box element.



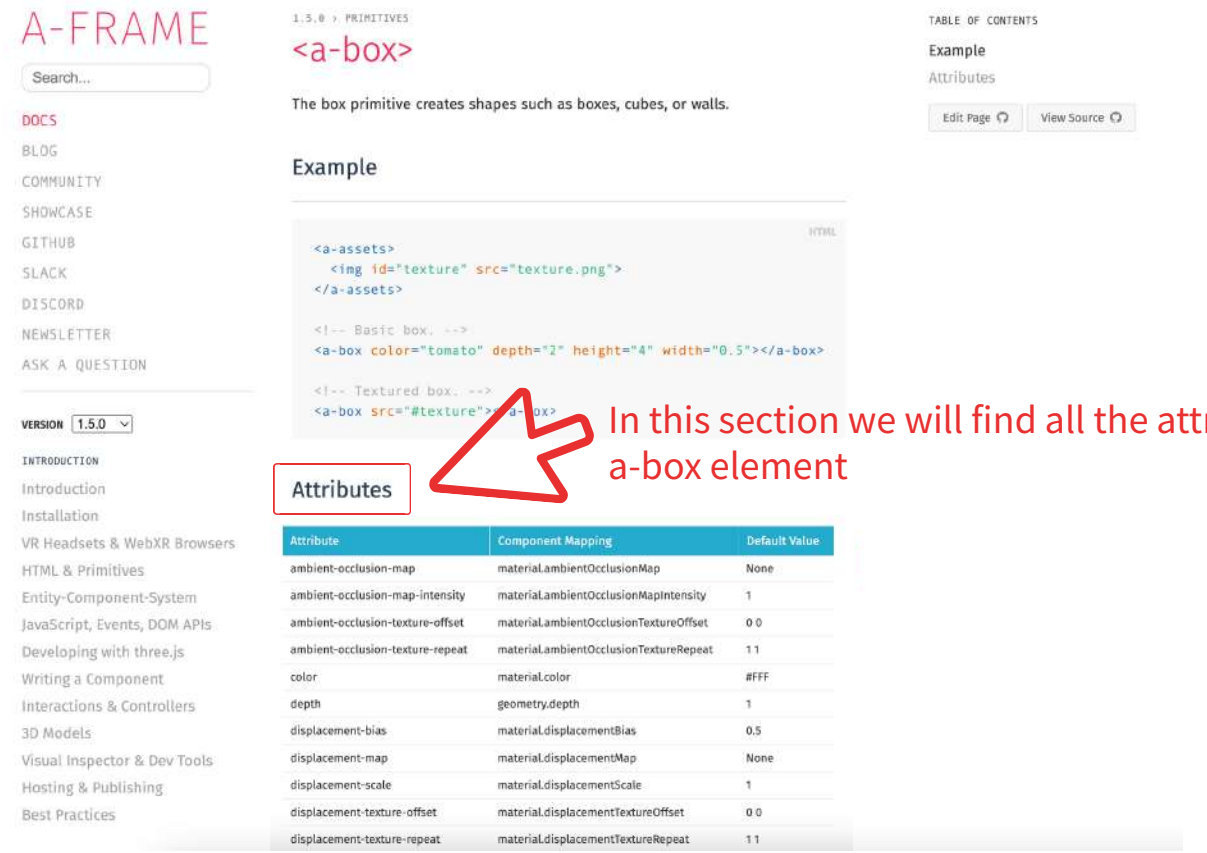
Basic line of code to add the box.we will copy this line of code to paste it in our project in Glitch.

With the box attributes changed, we will obtain the following result:



To remember how to find the documentation on how to add the basic a-box geometry in a-frame, see point **3. A-Frame. Get to know the platform**

Going back to the a-box documentation on the a-frame website, we can find all the attributes to work with the box element.



In this section we will find all the attributes that we can add to the a-box element

Next steps

In the following modules of the course we will learn how to work and create rooms using boxes.

In the following example you can see a preview of how to work with boxes for the creation of virtual rooms:

Bibliography

and useful resources

A-Frame (2015). A web framework for building 3D/AR/VR experiences. Disponible en: https://aframe.io/

Gill, A. (2017). AFrame: A domain specific language for virtual reality. In *Proceedings of the 2nd International Workshop on Real World Domain Specific Languages* 1-1.

Sánchez-Acedo, A. (2023). Guía básica para crear un entorno inmersivo con A-Frame. Disponible en: https://ciberimaginario.es/2023/02/17/guia-basica-para-crear-un-entorno-inmersivo-con-a-frame/

Creating virtual scenes with A-Frame | A-Frame Course I

Audiovisual and multimedia production: CIBERIMAGINARIO GRUPO DE INVESTIGACIÓN

This work was supported by the Autonomous Community of Madrid (Spain) with a grant for industrial doctorates (IND2022/SOC-23503) with the collaboration agreement with Prodigioso Volcán S.L; Universidad Rey Juan Carlos (ID 501100007511)