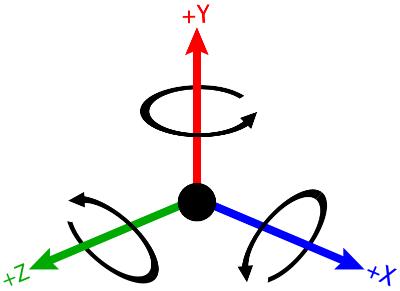
Physics

Adding simulated physics to a 3D environment is always exciting and the Accelerate Editor has physics built into it, we just need to turn it on! First of all make a new document.

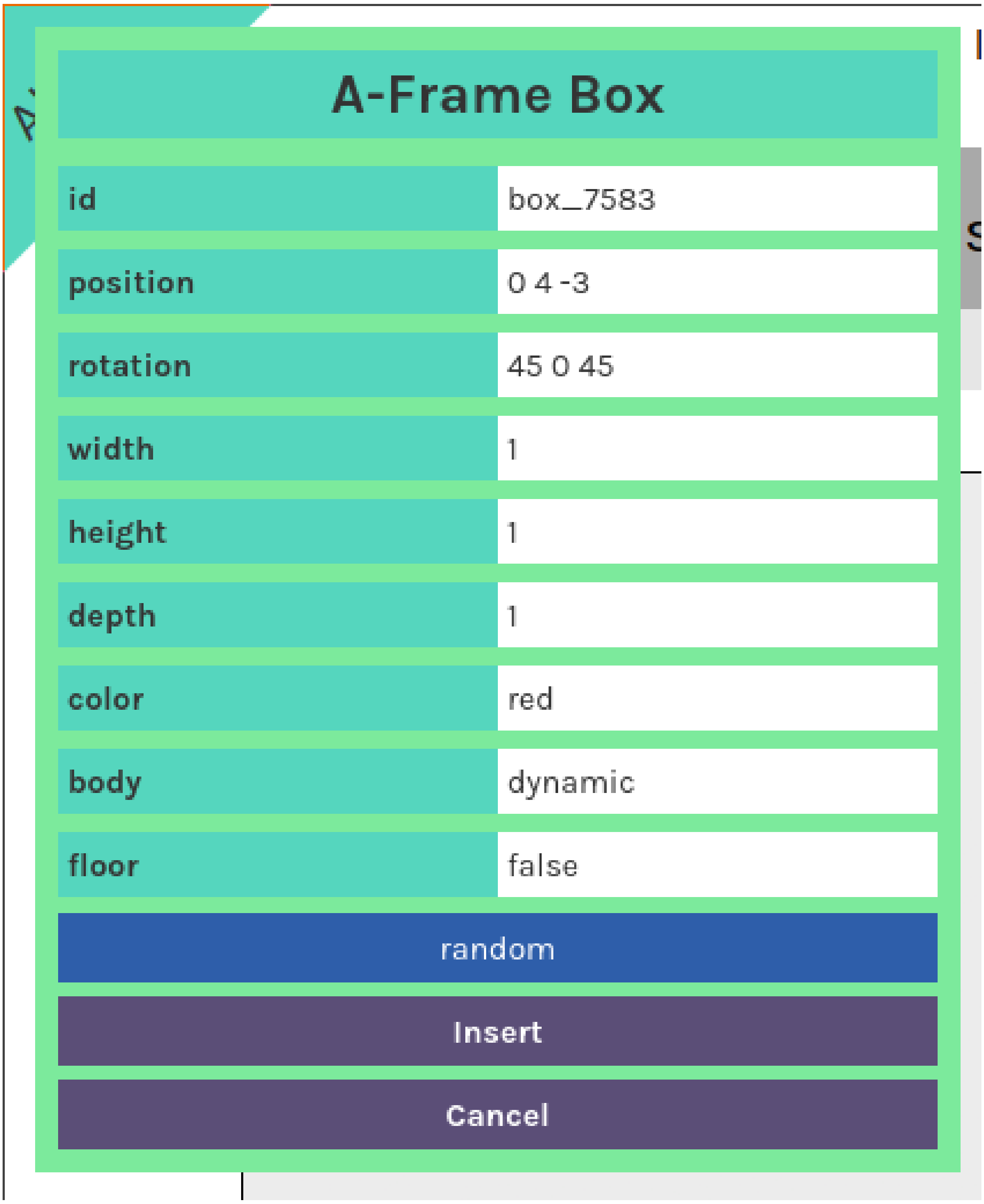
1. Adding a Falling Box

Open the snippet editor and select the **A-Frame Box**. In the snippet editor change the position to `0 4 -3` to raise it up a bit. Also change the rotation to something like `45 0 45` - this rotates *around* the **X** and **Z** axes.

> This can be a little tricky to get your head round at first: rotating in the **Y** axis rotates *around* the **Y** axis, like keeping a mug flat on a table, but rotating it. This diagram *might* help:



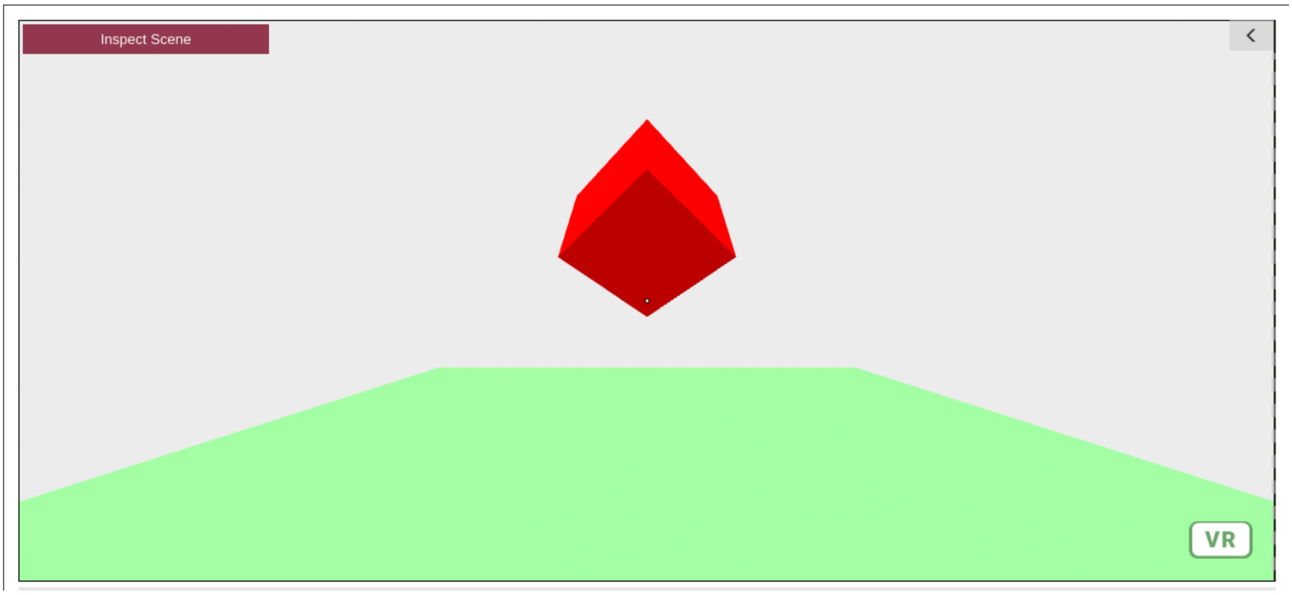
Finally change the **body** to **dynamic**. The full snippet is below:



A-Frame Box

id	box_7583
position	0 4 -3
rotation	45 0 45
width	1
height	1
depth	1
color	red
body	dynamic
floor	false

This will create a **dynamic** box which means it will behave with simulated physics. Press insert and watch your falling box fall into the scene! You can **Play / Pause** to re-run the scene.

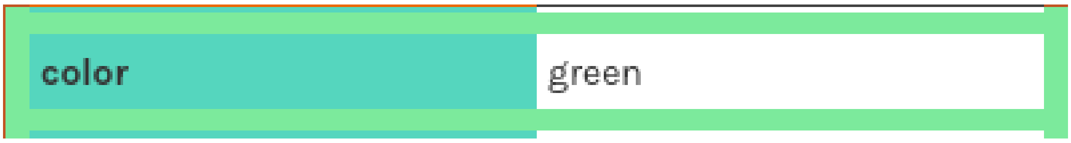


Dynamic elements will interact with each other too. Try adding more primitives through the snippet editor, or in code, and create a scene of falling objects.

2. Some Tips

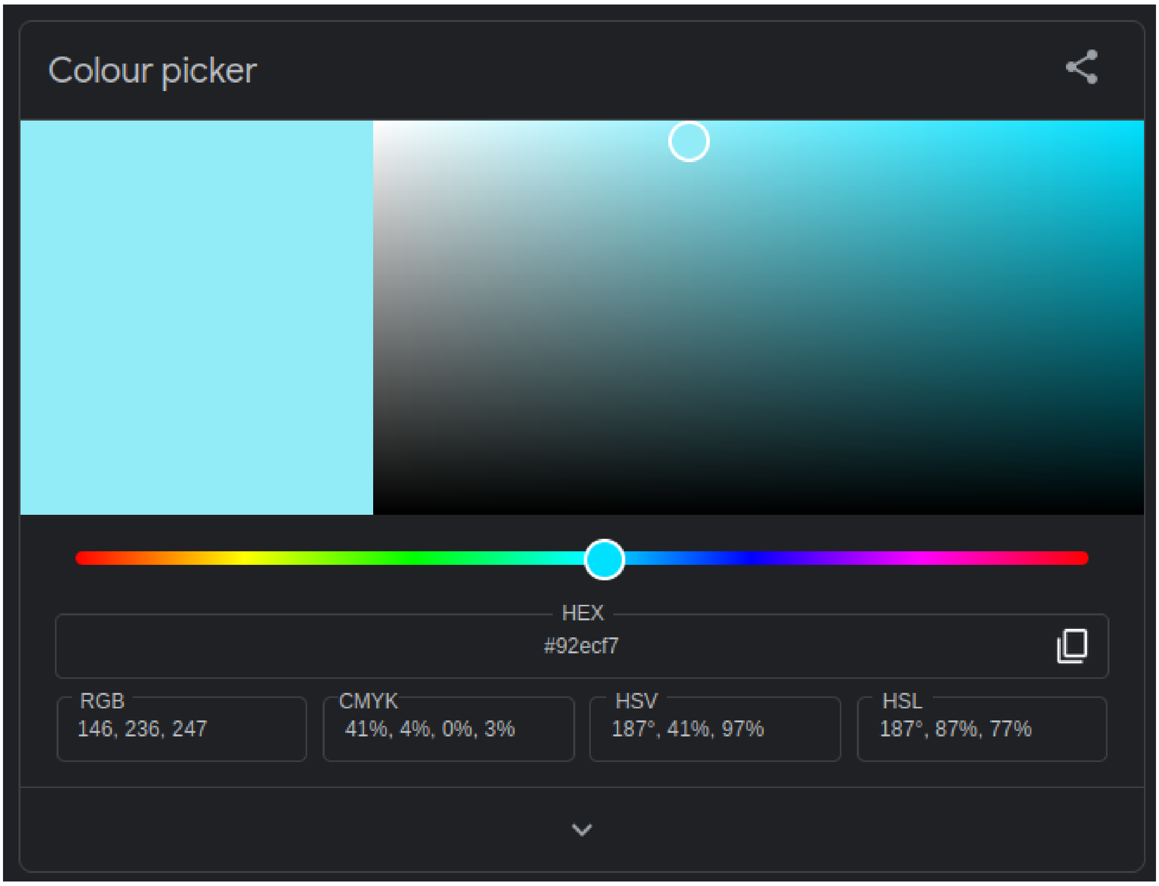
Making changes If you forget to change the **body** type in the snippet editor, remember you can edit values in code. Just find the object in the code (maybe **auto-format** to make it easier to read) and adjust the relevant value. Or maybe make use of the **Scene Inspector**! Just look for the line which looks like this `body="type:static;"` and change it to `body="type:dynamic;"`.

You may have noticed the **color** field in the snippet editor.



color green

The colours in the Accelerate Editor are **CSS Colors** which are 140 common colors which have been given a unique name which are understood by all browsers. This makes our life easier and we can just remember colours names we like, rather than complex hex codes or RGB values. Sometimes it's worth just trying a colour name to see if it works, like **lightblue** or **hotpink** - a full list of CSS colours is here. That said, if there is not a colour you like, you *can* also use a hex code if you prefer. Try using a colour picker like this one, choosing a colour and copying the hex code into the colour field of a snippet - **make sure to include the hash symbol like this "#92ecf7"**.


Colour picker

HEX
#92ecf7

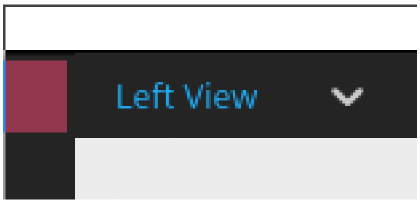
RGB
146, 236, 247

CMYK
41%, 4%, 0%, 3%

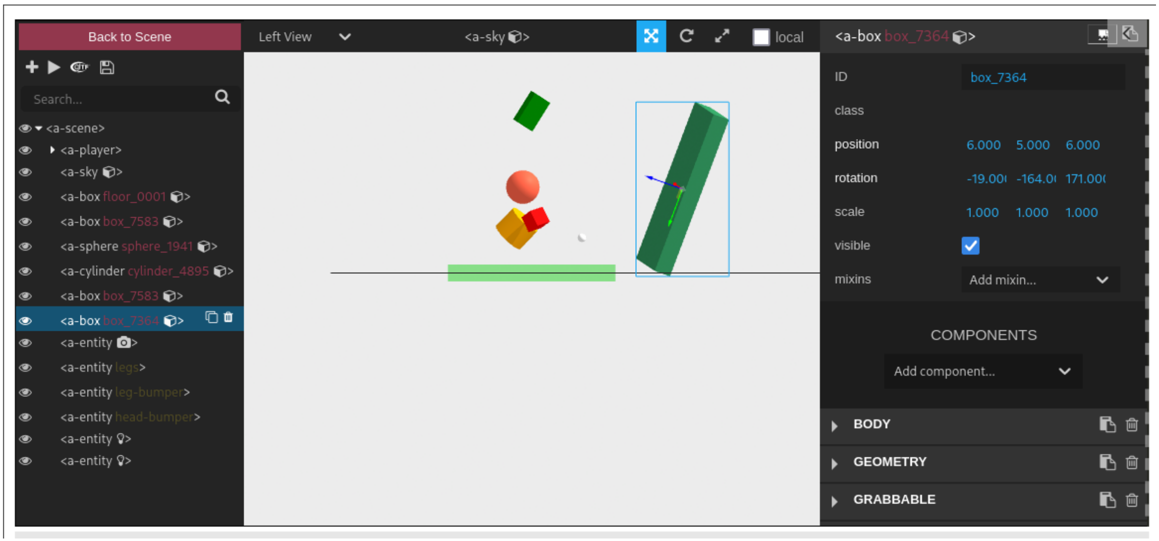
HSV
187°, 41%, 97%

HSL
187°, 87%, 77%

This is a good opportunity to play with the **Random** button in the snippet editor! The **Random** button will randomise the values within a reasonable range. It's an easy way to populate a scene with objects. You will need to keep and eye on the values to make sure they are not appearing below the floor, or behind the player (positive Z value). If you can't find an object you have added the **Scene Inspector** is your friend here. Open the scene inspector and zoom out by scrolling, or even change the view. You can see here that a random box was added with the position 6 5 6 which will appear behind the player. This is obvious when you choose the **Left View** or **Right View** in the scene inspector.



You can move the box around as you wish to get it into a better position.



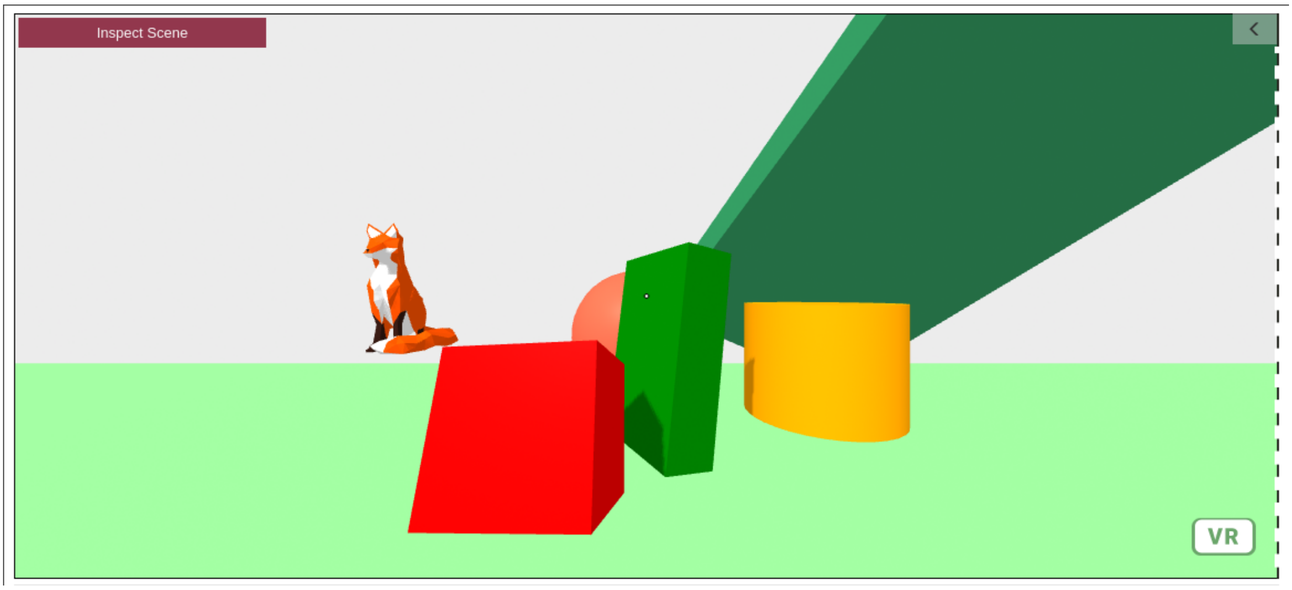
```
<a-scene>
  <a-player>
  <a-sky>
  <a-box floor_0001>
  <a-box box_7583>
  <a-sphere sphere_1941>
  <a-cylinder cylinder_4895>
  <a-box box_7583>
  <a-box box_7364>
  <a-entity>
  <a-entity legs>
  <a-entity leg-bumper>
  <a-entity head-bumper>
  <a-entity>
  <a-entity>
```

<a-box box_7364>

ID	box_7364		
class			
position	6.000	5.000	6.000
rotation	-19.00(	-164.0(	171.00(
scale	1.000	1.000	1.000
visible	☑		
mixins	Add mixin...		

- BODY
- GEOMETRY
- GRABBABLE

You can add physics to `.GLB` models too! Try adding a fox via the snippet editor, change the position and make it a **dynamic body**.



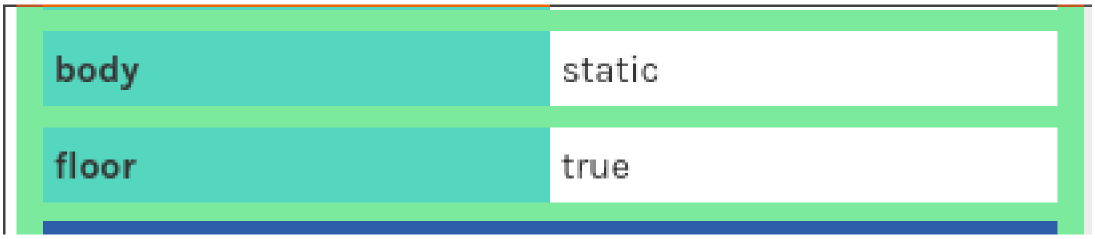
You might notice the fox falls in a slightly odd way. Sometimes `.GLB` files will have an unusual shape to them which makes it hard for the physics engine (the thing doing the hard simulation work) to calculate the exact shape of the object. **Simulating physics is a hard thing to do and so as the scene gets bigger and more complex you will find things slowing down. So try to use physics sparingly if you are creating a complex scene.**

3. The Floor

It's worth looking at the floor of our scene for a second. The default document will add this for you. If you look at the code you will notice it has the keyword `floor` in there, and a body type of `static`:

```
<a-box
  id="floor_0001"
  position="0 0 -3"
  rotation="0 0 0"
  color="lightgreen"
  width="10"
  height="1"
  depth="10"
  body="type:static;"
  grabbable="physics:true;"
  floor
></a-box>
```

These are important details and you will need to add these if you want to make other *floors* in your scene. Of course you can do this code, or you can specify these details in the snippet editor when you add an **A-Frame Box**:



body	static
floor	true