How to build a house in 3D

A short introduction to babylon.JS

@merelyChristina & @merelyAnna

Who are we?

CHRISTINA

merelyChristina

ANNA

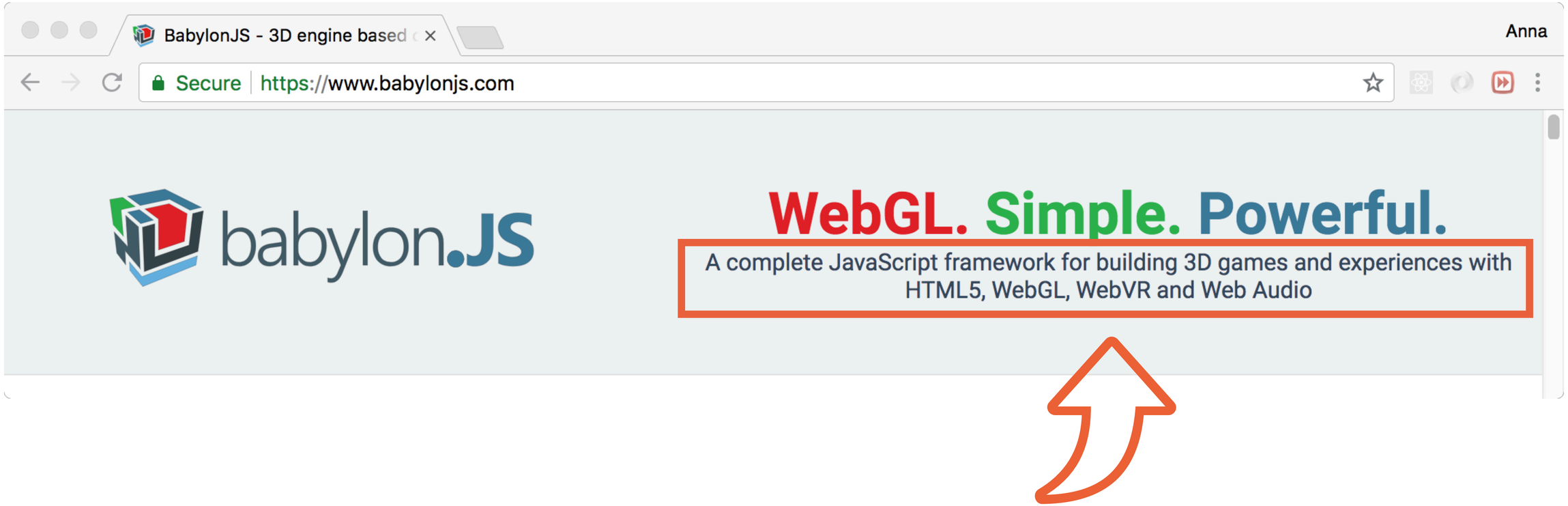
merelyAnna

codecentric

Have you ever thought about doing 3D stuff?

We are going to show you how to get started.

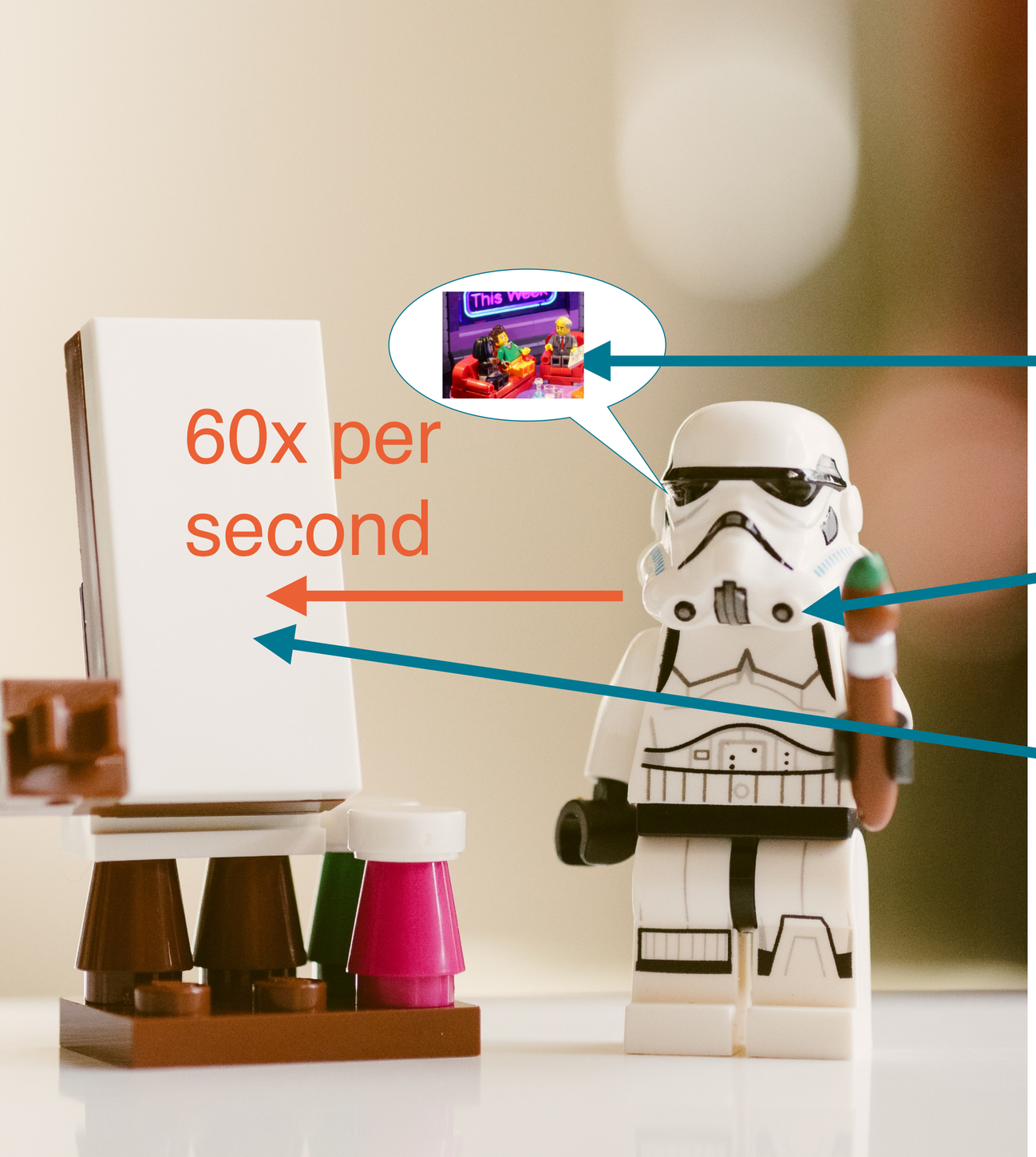
What is babylon.JS?



What is babylon.JS?

3D game engine
running in a browser

How does it work?



Scene

Engine

Canvas

Canvas

```
<html>
  <head> </head>

  <body style="overflow: hidden; margin: 0px">
    <canvas style="width: 100%; height: 100%;" id="babylon-canvas"></canvas>
  </body>
</html>
```

Getting the engine running

```
const initGame = () => {
  const canvas = document.getElementById("babylon-canvas");

  const engine = new Engine(canvas, true);
  const scene = new Scene(engine);

  engine.runRenderLoop(function() {
    scene.render();
  });
};

initGame();
```

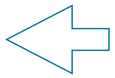

I want to see
something
on the canvas

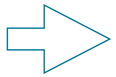
Parameters

```
const thing = new Thing("name", { }, scene);
```

class name options scene

Universal camera


Left

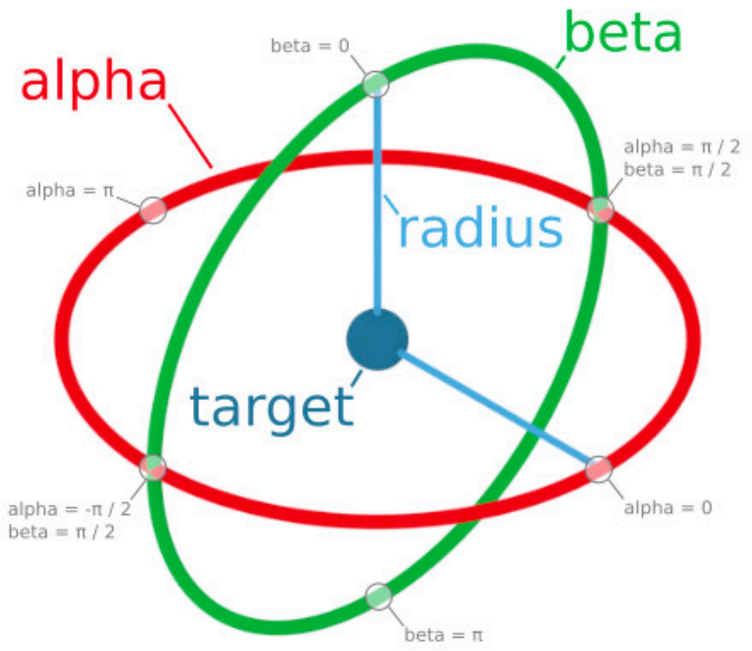

Right


Forward


Backward

```
const camera = new UniversalCamera("camera",
new Vector3(0, 0, -10), scene);

// This targets the camera to scene origin
camera.setTarget(Vector3.Zero());

// This attaches the camera to the canvas
camera.attachControl(canvas, true);
```

Arc rotate camera

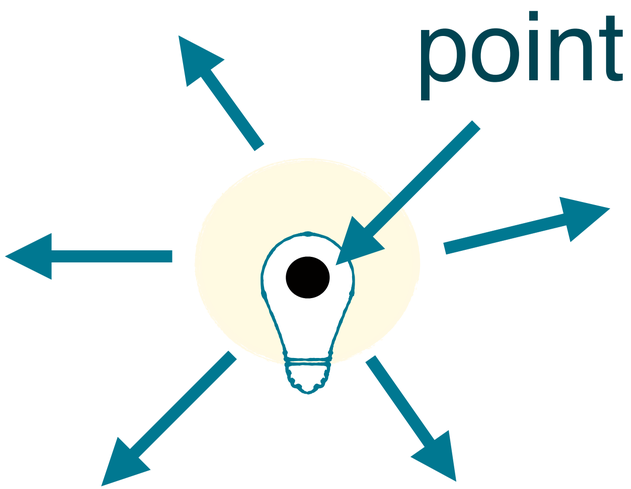



```
const camera = new ArcRotateCamera("camera", 0,
0, 20, new Vector3(0, 0, 0), scene);

// This attaches the camera to the canvas
camera.attachControl(canvas, true);
```

Image source: https://doc.babylonjs.com/babylon101/cameras#arc-rotate-camera

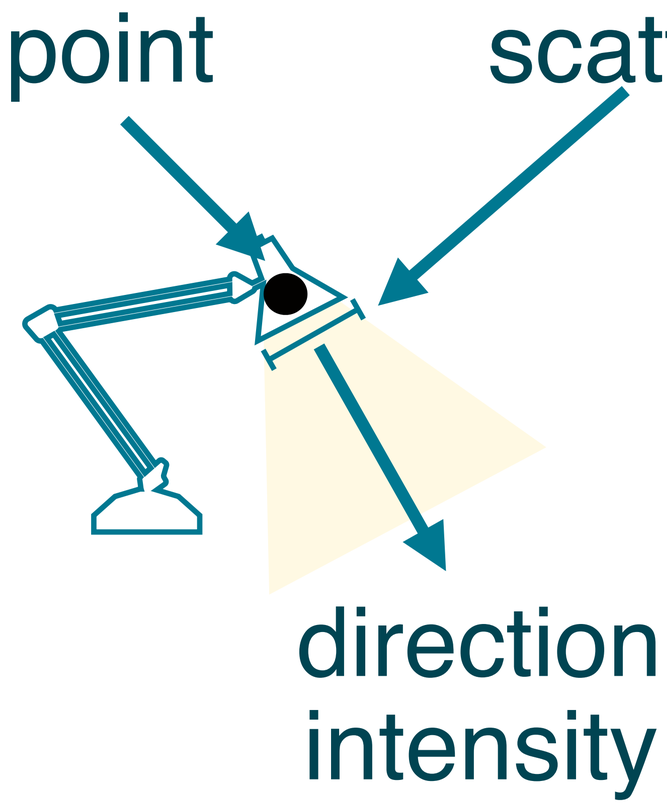
Without light
everything
is black

Point light

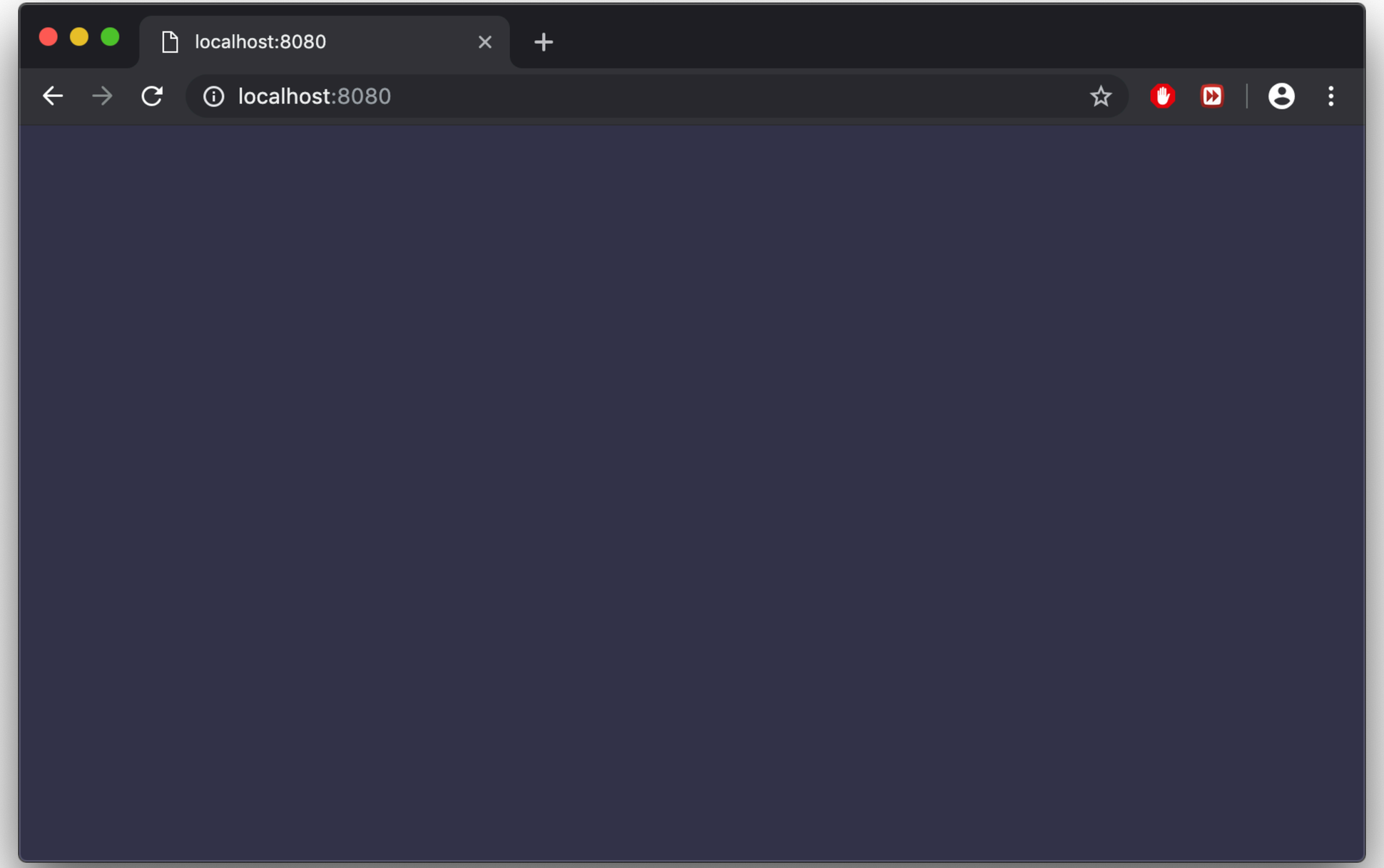



```
const light = new PointLight(
  "pointLight",
  new Vector3(0, 10, 0),
  scene
);
```

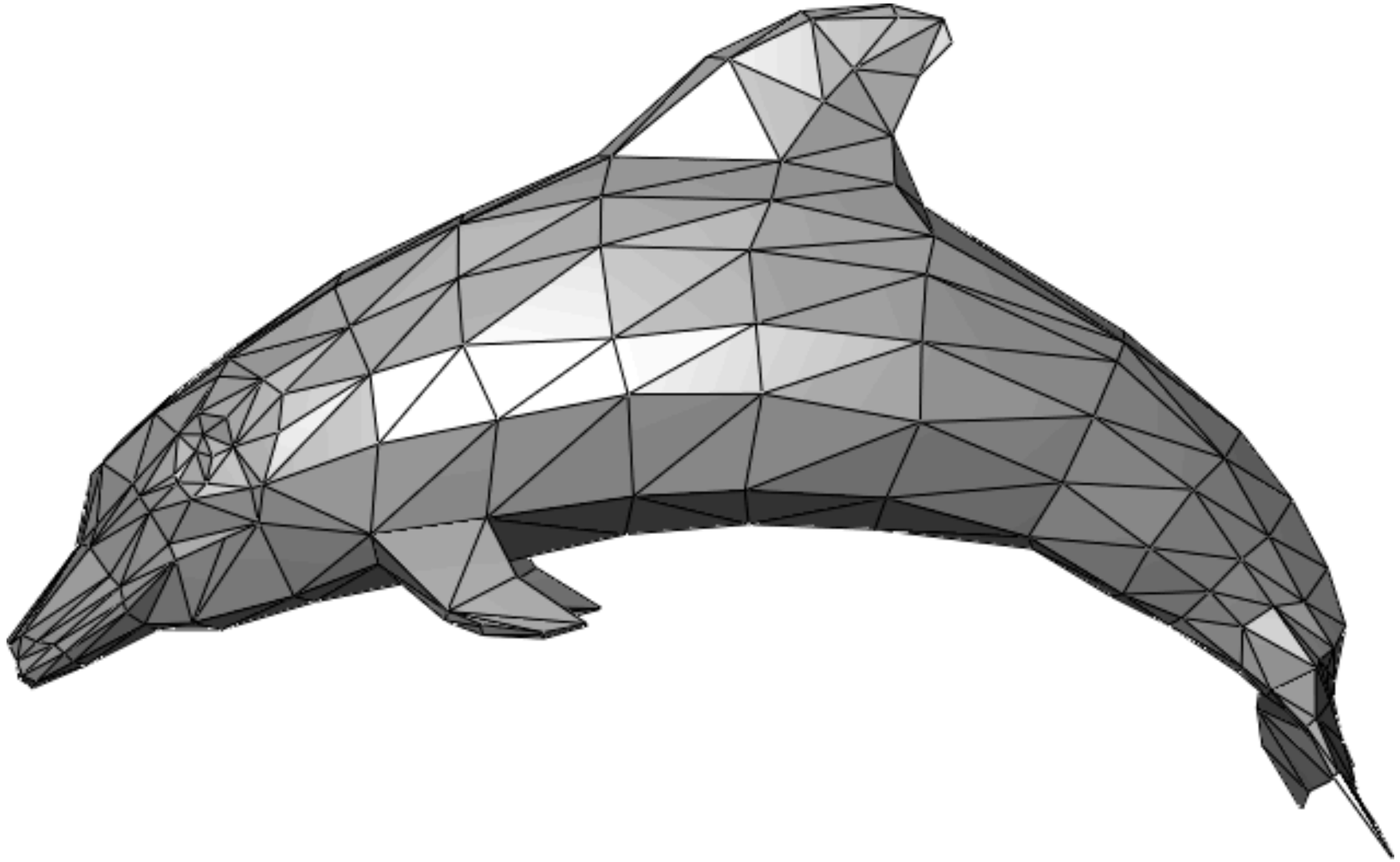
Spot light



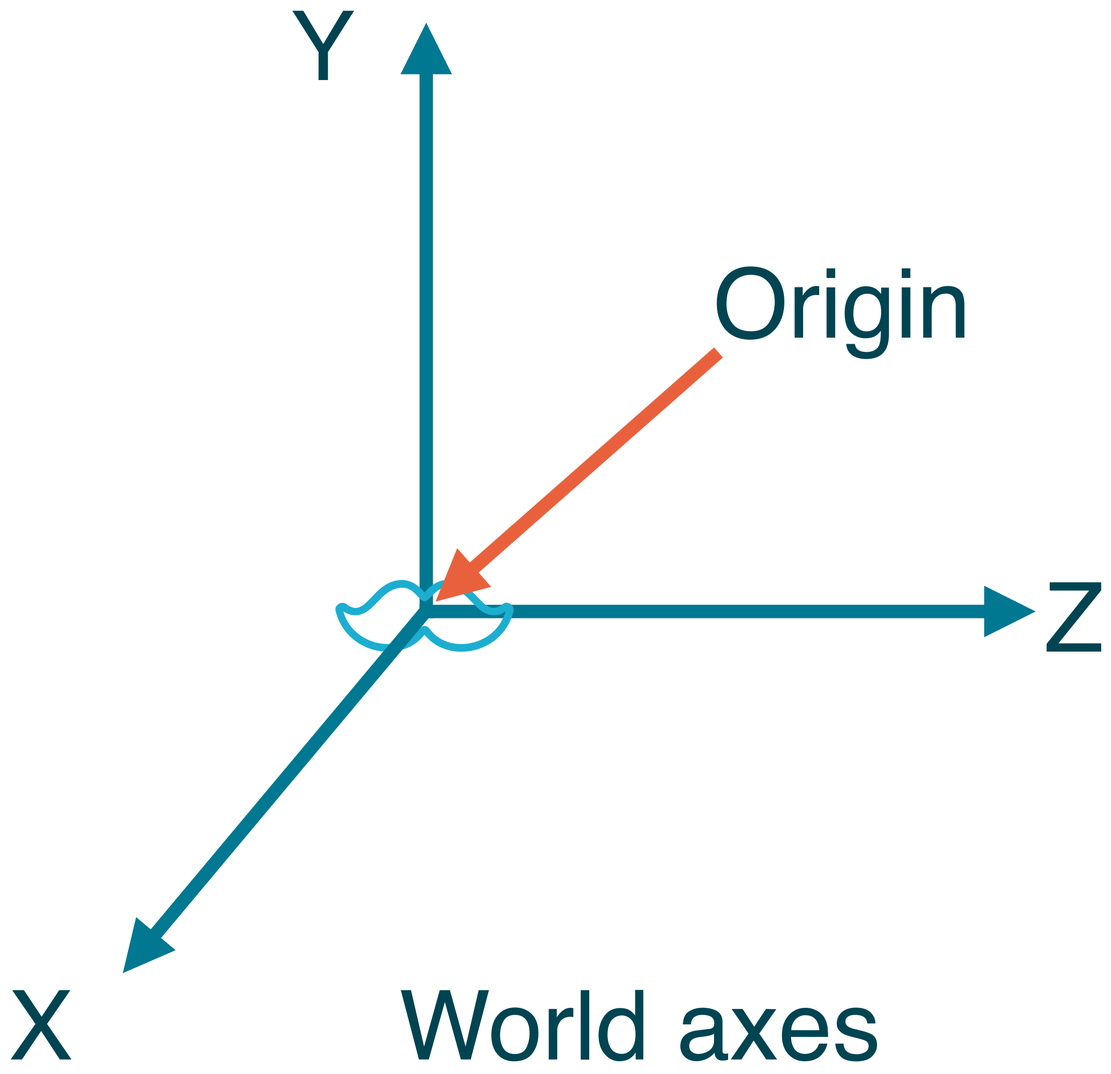

```
const light = new SpotLight(
  "spotLight",
  new Vector3(0, 10, 0),
  new Vector3(0, -1, 0),
  Math.PI / 2,
  10,
  scene
);
```


localhost:8080
localhost:8080

Mesh

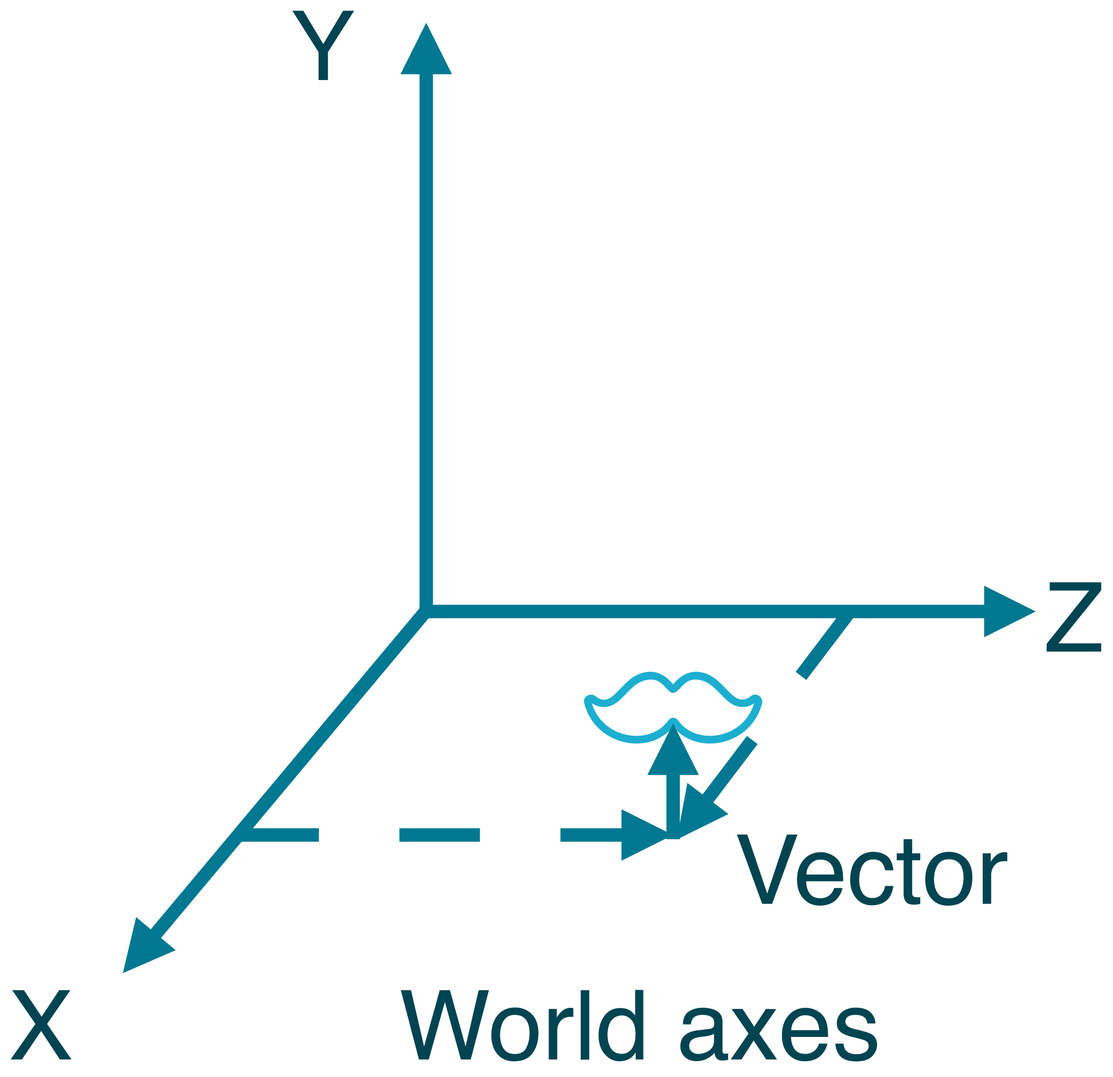


Positioning

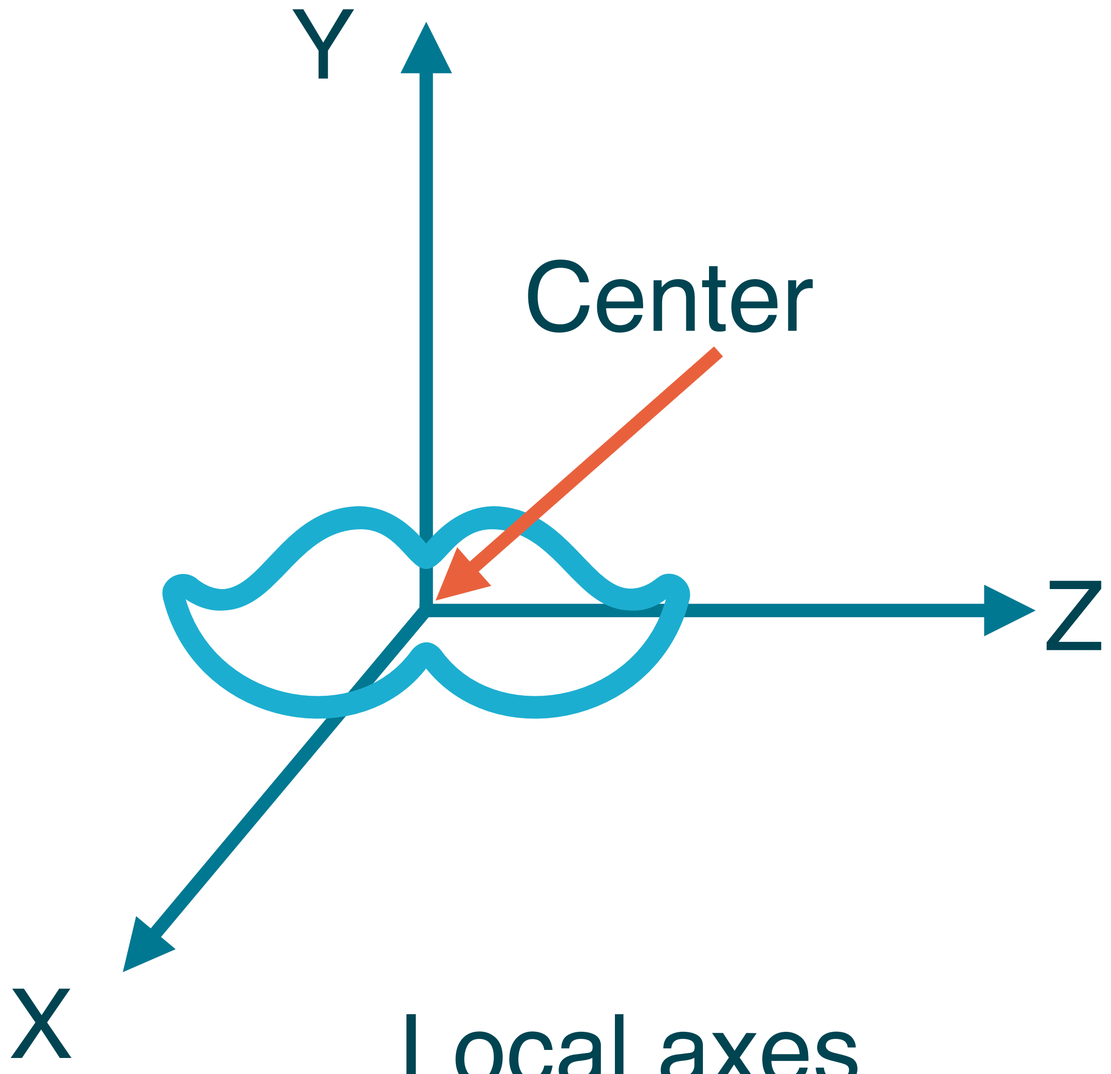


World axes

Positioning

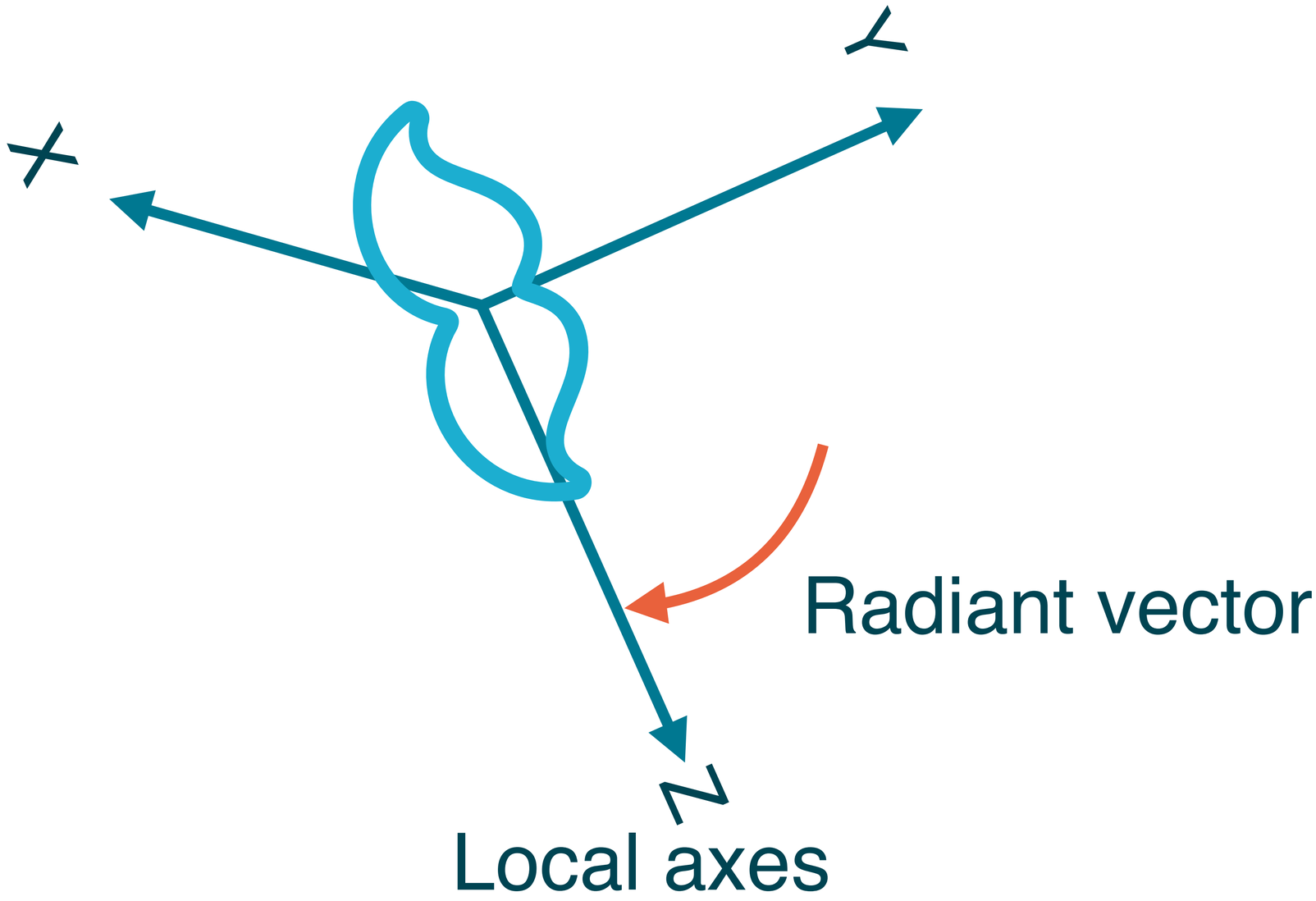


Rotation



Local axes

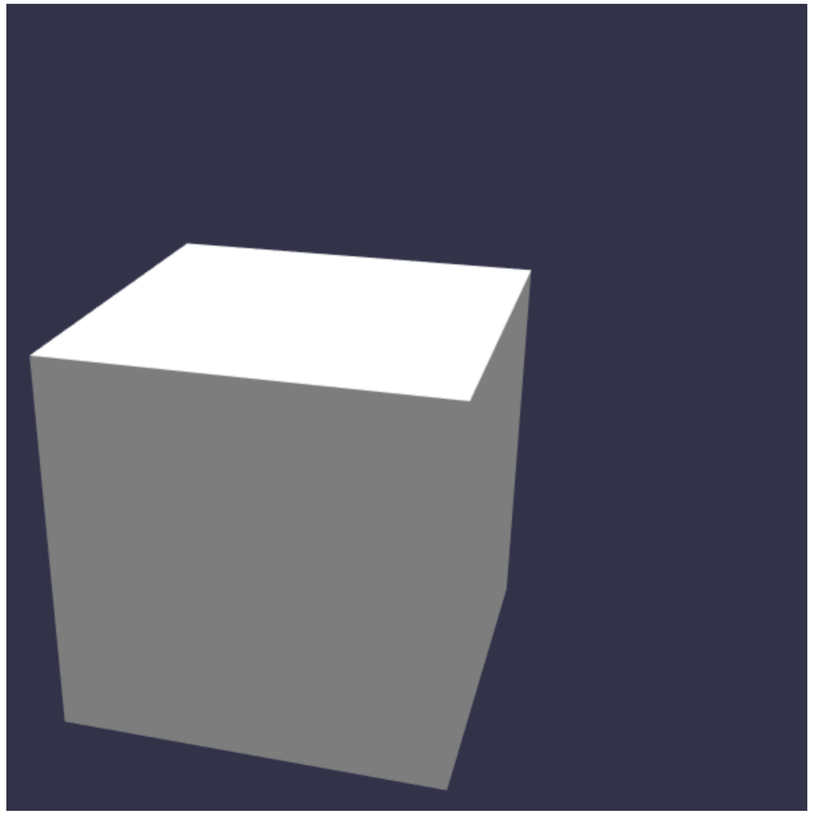
Rotation

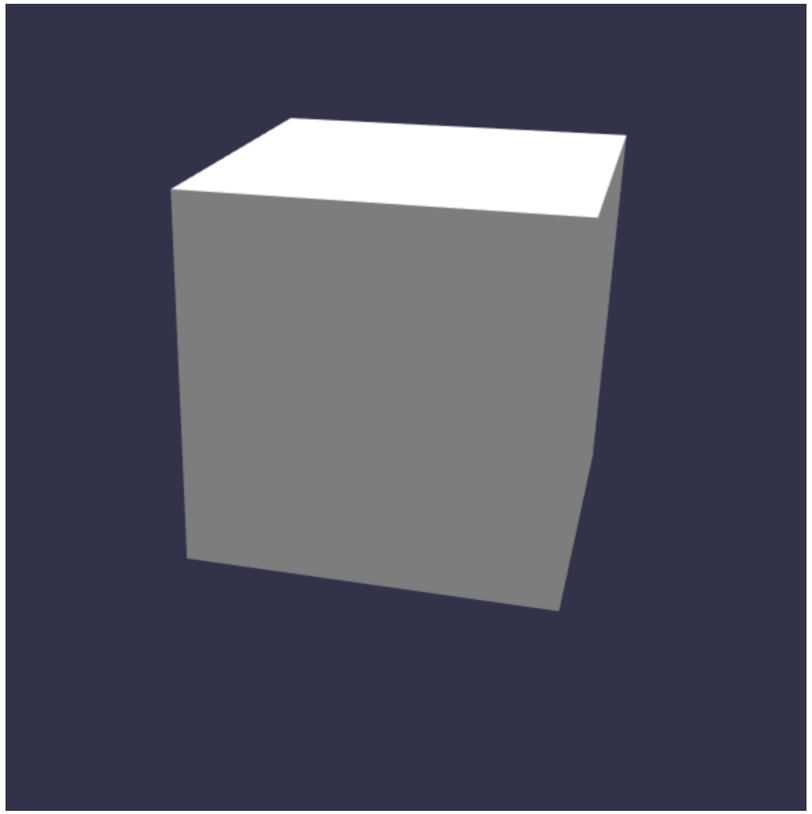


Example

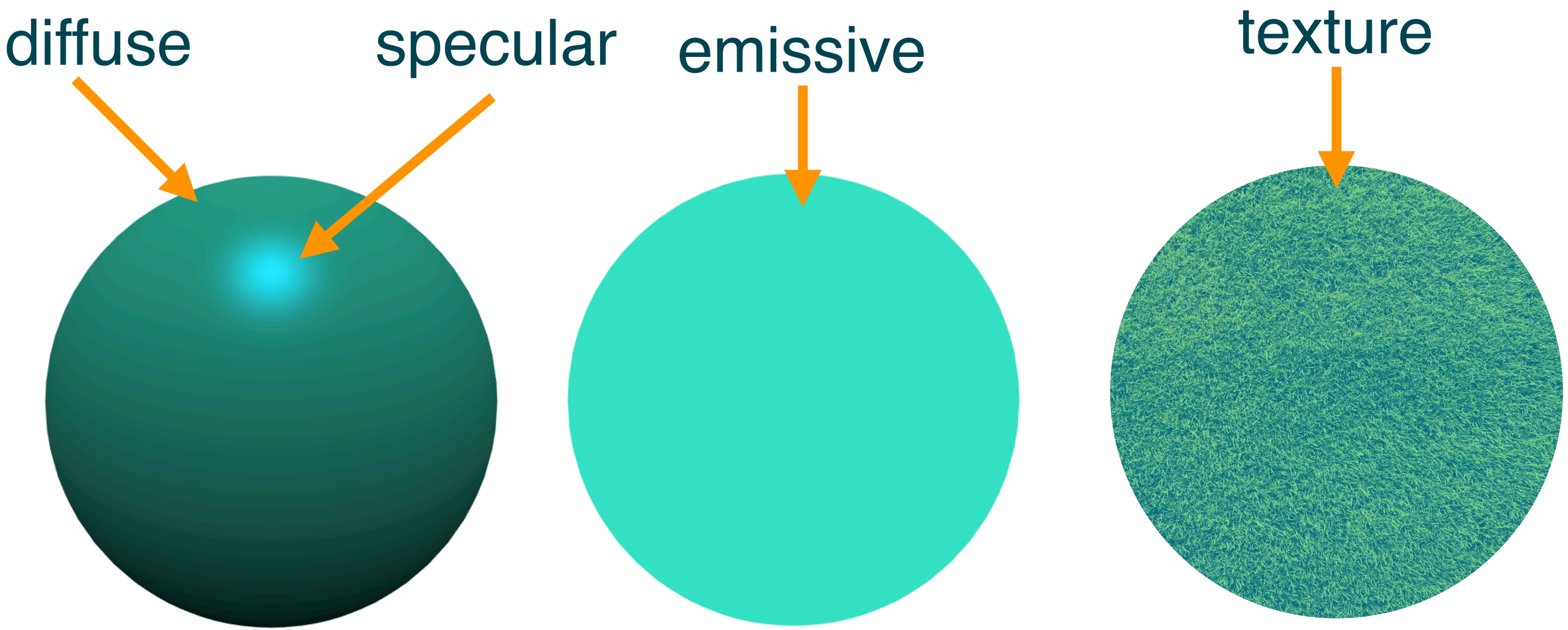
```
//Creating the box mesh
const box = MeshBuilder.CreateBox(
    "box",
    { width: 10, height: 10, depth: 10 },
    scene
);

// Positioning the box
box.position.x = 7;
box.position.y = 8;
box.position.z = 9;

// Rotating the box
box.rotation = new Vector3(2, 3, 4);
```

Example





Materials

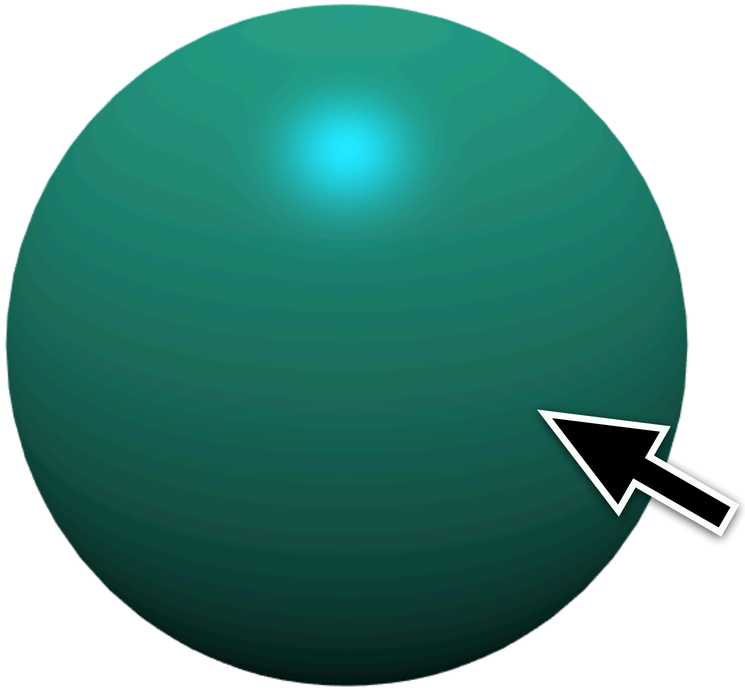


Materials

```
// Creating the material
const material = new StandardMaterial("material");
material.diffuseColor = new Color3(0.22, 0.89, 0.76);
material.specularColor = new Color3(0, 0.5, 0.76);

box.material = material;
```

User interaction

Action Manager



Actions

Actions Typ

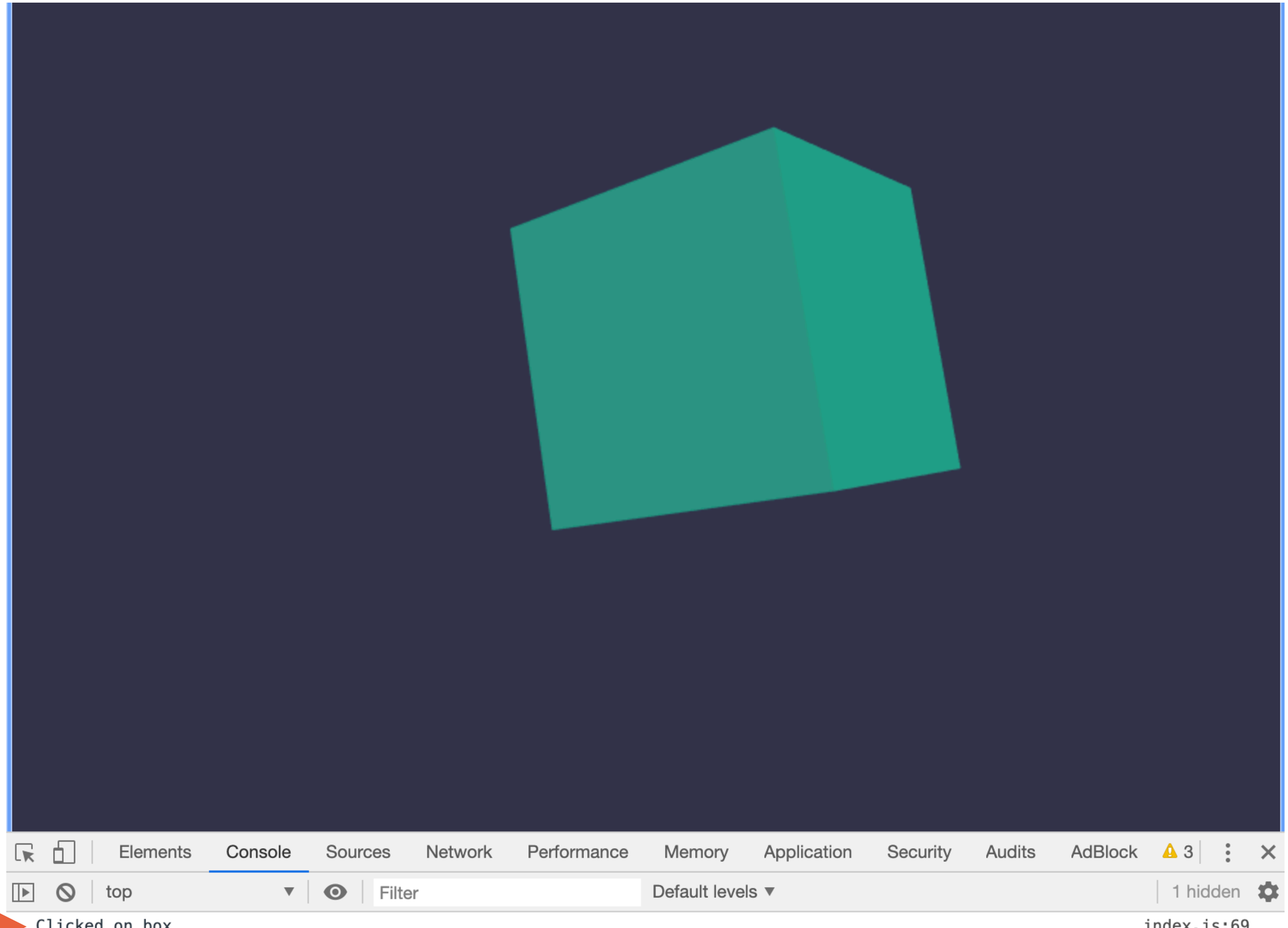
- Set value
- Play sound
- Execute code
- …

Trigger

- Click
- Double-click
- Press key
- …

Example

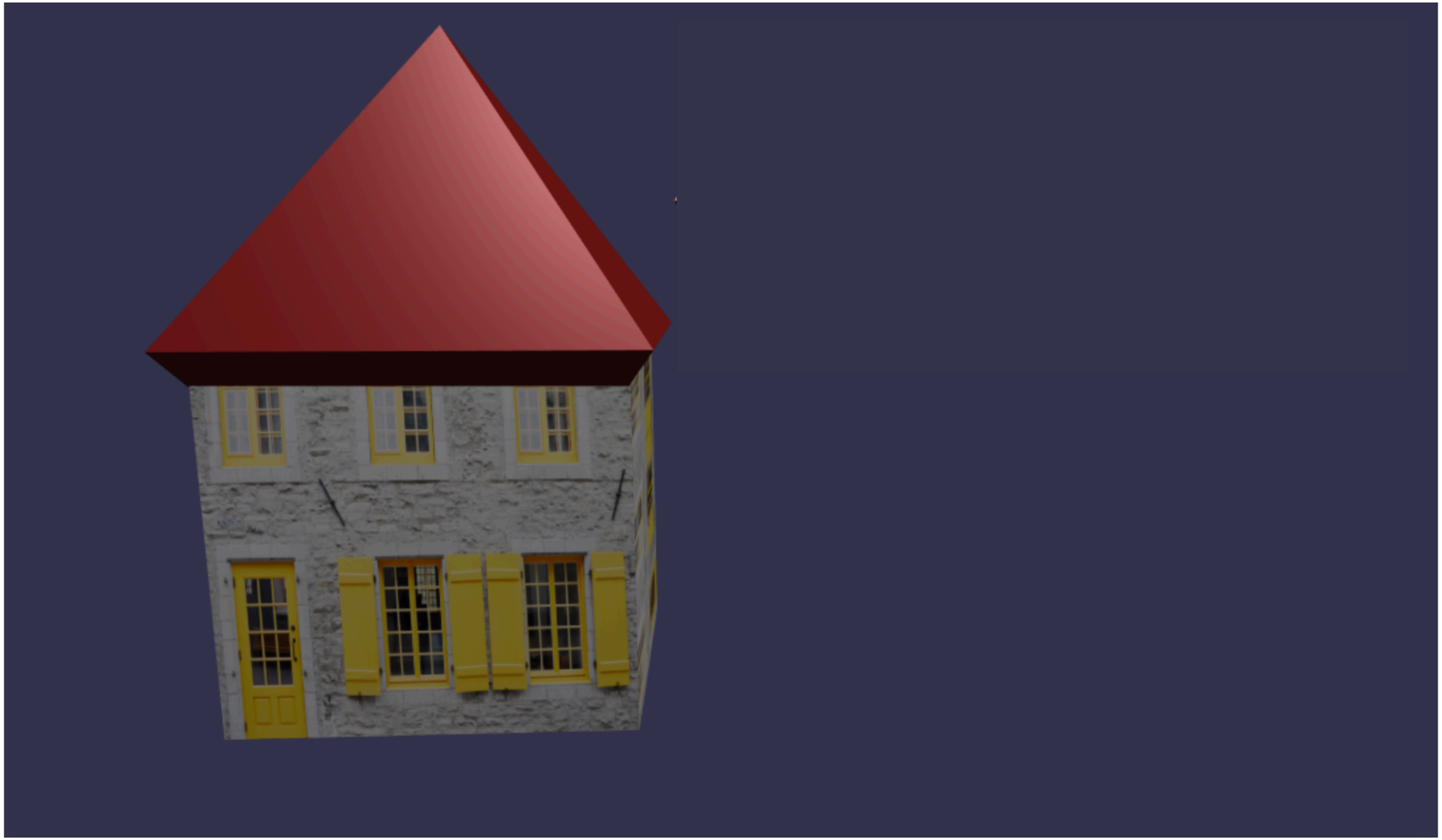
```
// Adding an action
box.actionManager = new ActionManager(scene);

box.actionManager.registerAction(
    new ExecuteCodeAction(ActionManager.OnPickTrigger, () =>
    {
        console.log("Clicked on box");
        box.material = material;
    })
);
```


Elements
Console
Sources
Network
Performance
Memory
Application
Security
Audits
AdBlock
3
top
Filter
Default levels
1 hidden
Clicked on box
index.js:69

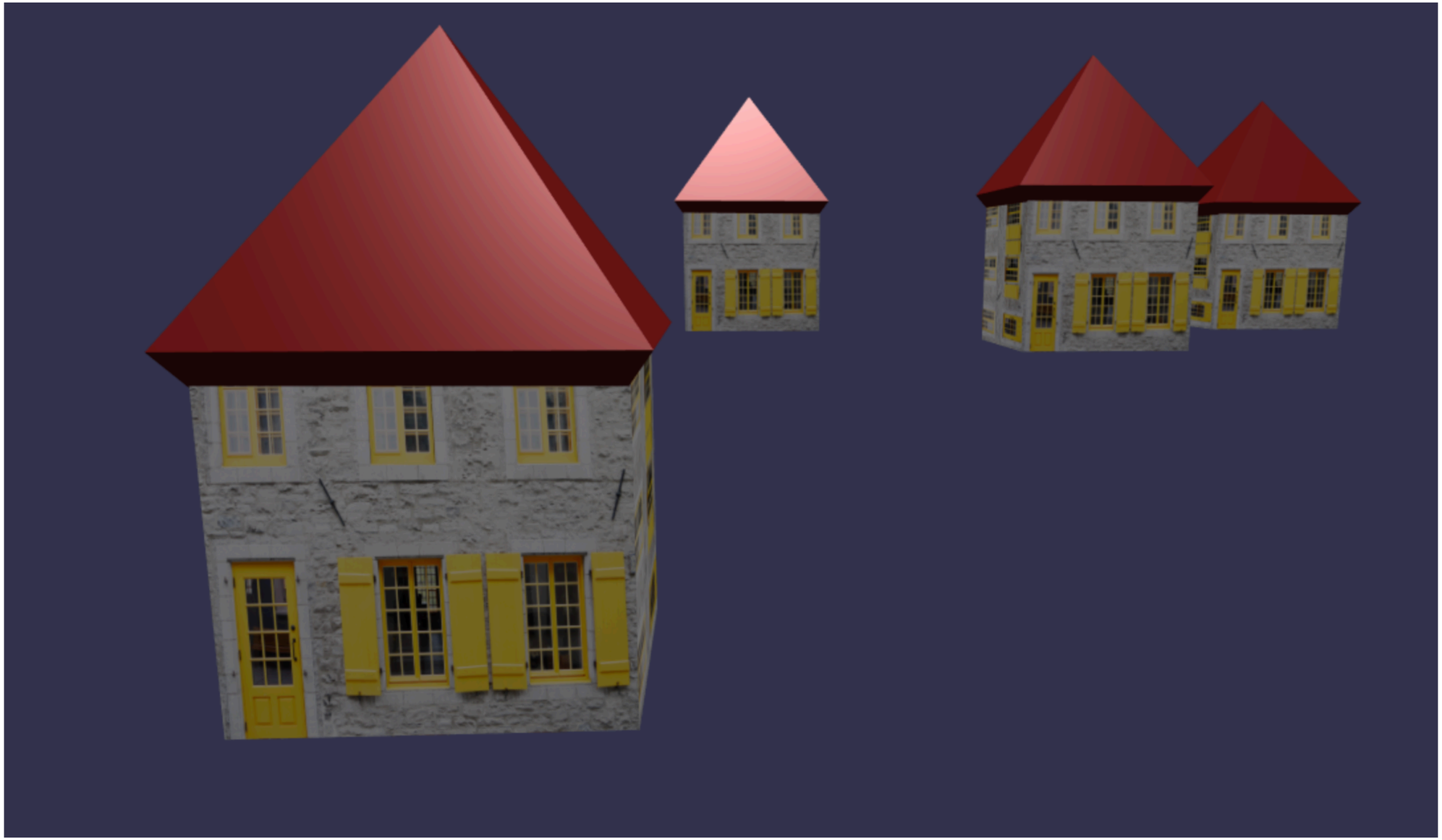

Isn’t the talk about building a house? 🤔

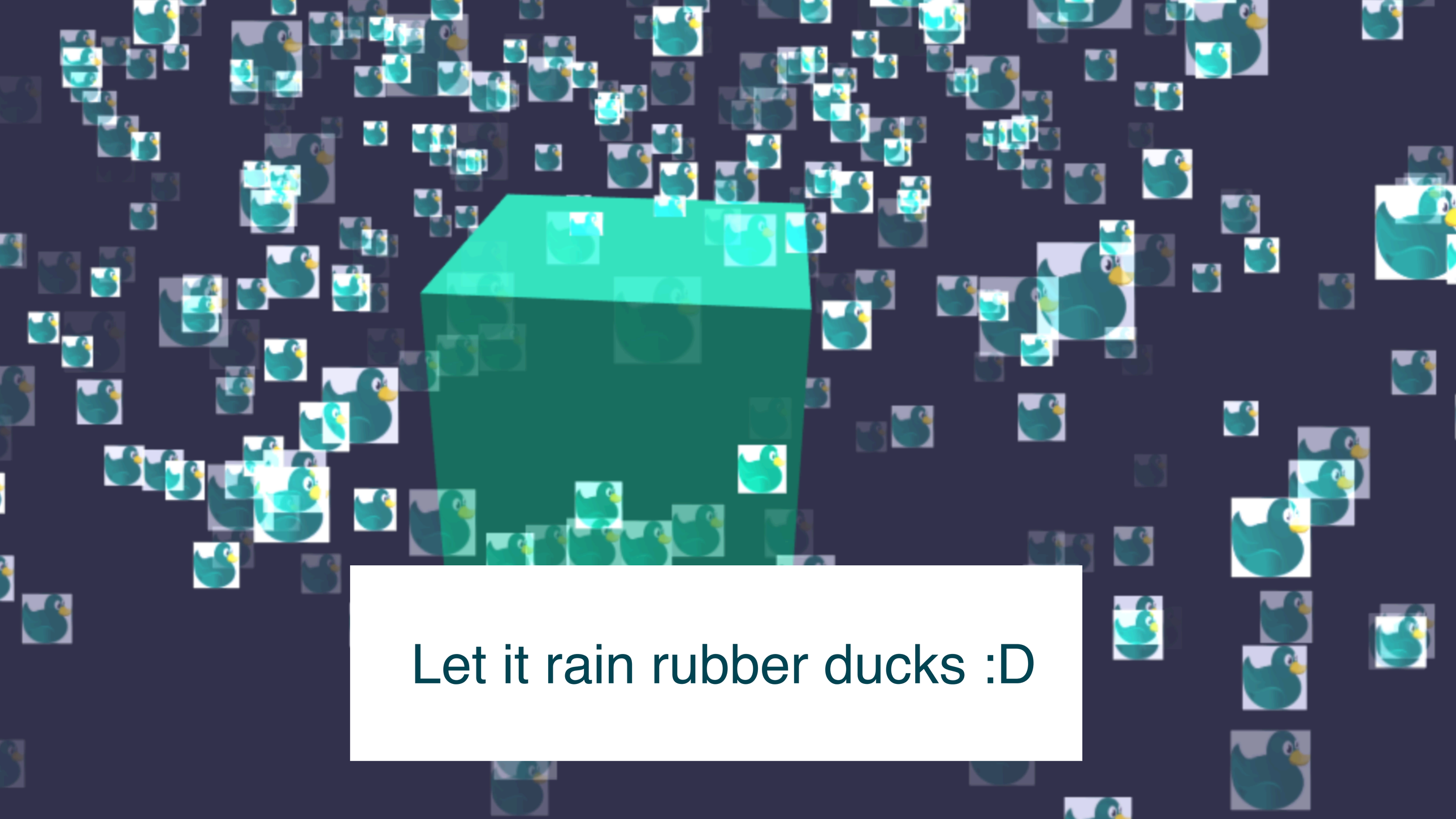
```
const house = MeshBuilder.CreateBox(
  "house",
  { height: 5, width: 5, depth: 5 },
  scene
);

const houseMaterail = new StandardMaterial();
houseMaterail.diffuseTexture = new Texture("/assets/facade.jpg", scene);
house.material = houseMaterail;

const roof = MeshBuilder.CreatePolyhedron(
  "roof",
  { type: 1, size: 2.8 },
  scene
);

const roofMaterial = new StandardMaterial();
roofMaterial.diffuseColor = new Color4(0.5, 0.1, 0.1, 1);
roof.material = roofMaterial;

roof.position.y += 2.2;
roof.rotation.y = 0.79;
```

The house

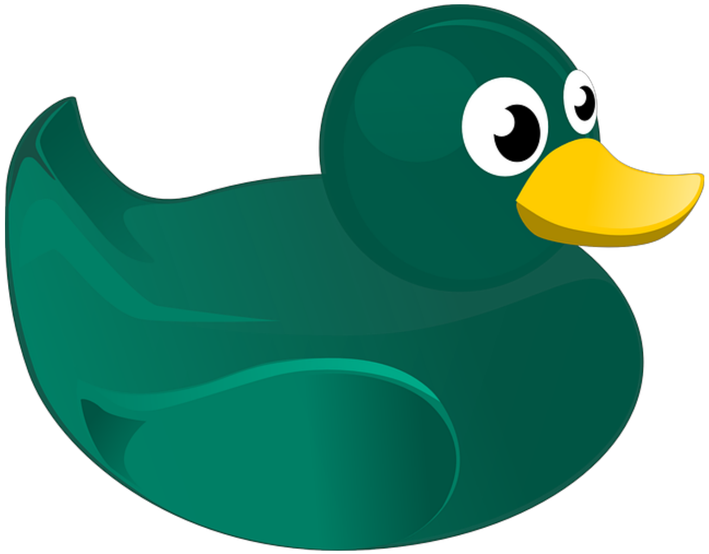


Multiple Houses :D




Let it rain rubber ducks :D

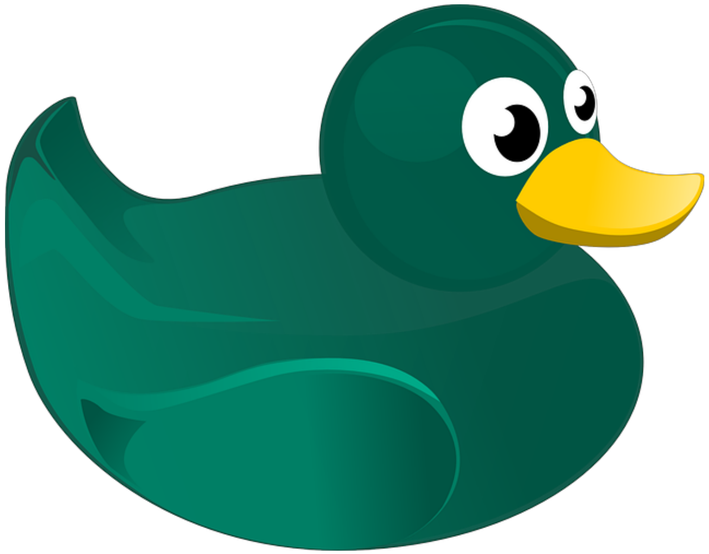
Particle System



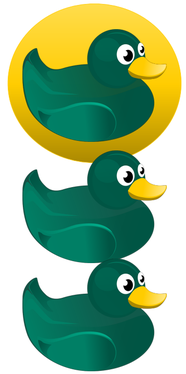
2D Sprite

Emitter

Particle System

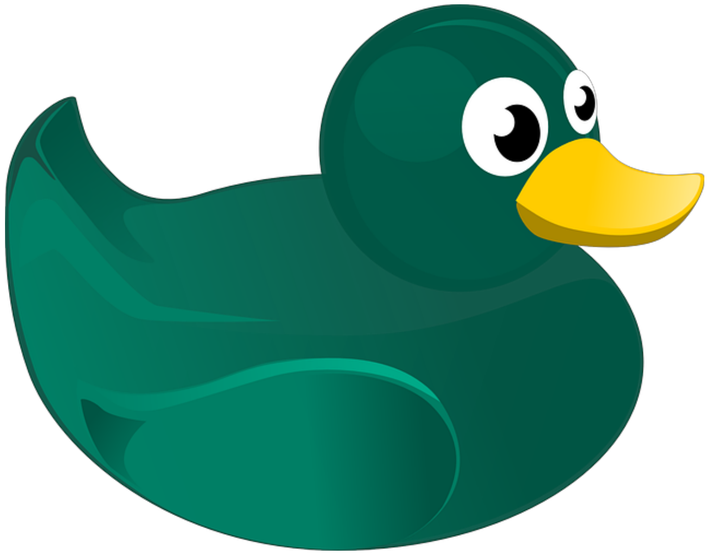


2D Sprite

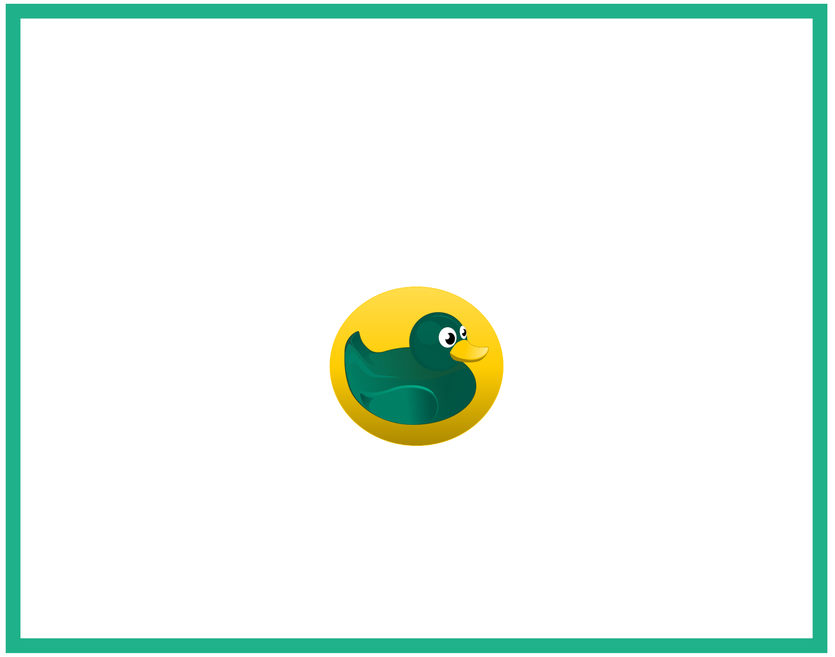


Emitter

Particle System

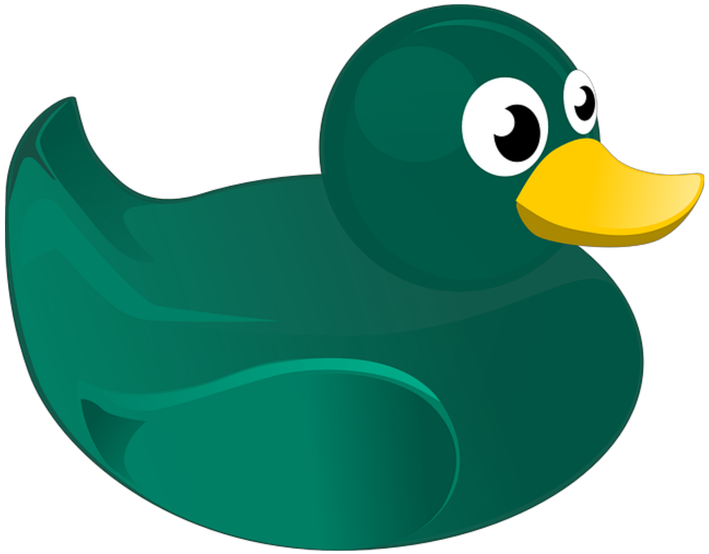


2D Sprite

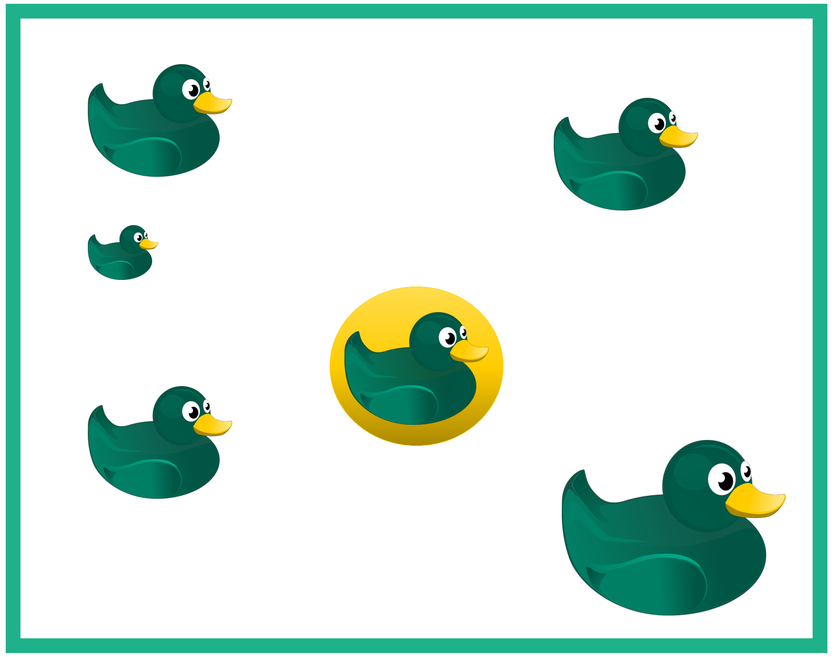


Emitter

Particle System



2D Sprite



Emitter

Particle Systems

```
const duckParticleSystem = new ParticleSystem(„ducks", 2000, scene);

duckParticleSystem.particleTexture = new Texture("/assets/duck.png", scene);
duckParticleSystem.emitter = new Vector3(0, 20, 0);
duckParticleSystem.minEmitBox = new Vector3(-20, -20, -20);
duckParticleSystem.maxEmitBox = new Vector3(20, 20, 20);
duckParticleSystem.gravity = new Vector3(0, -10, 0);
duckParticleSystem.emitRate = 999;

duckParticleSystem.start();
```

What else can it do?

Animation

Special FX

Debug Layer

Why use it?

Getting started

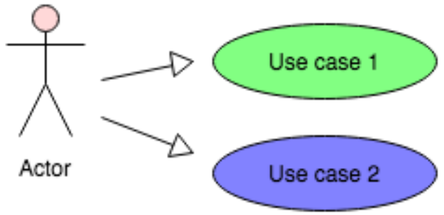
Runs in a browser

Extensive

Why not use it?

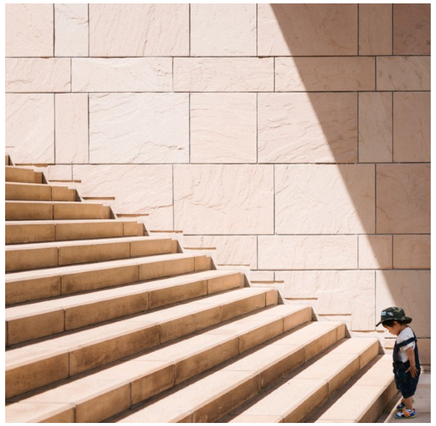
Extensive

Limits



Use Case

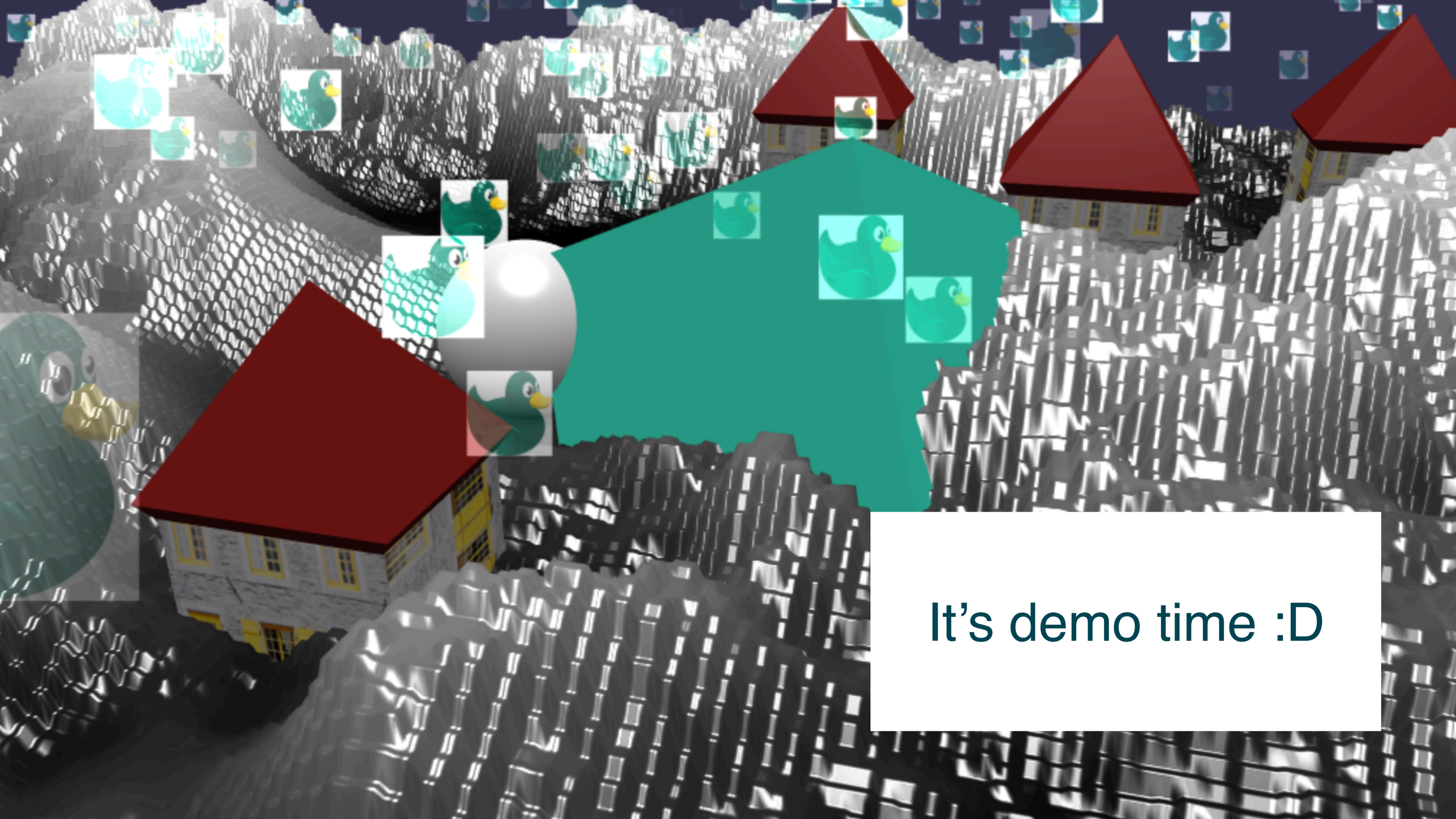
Our experience



Learning curve

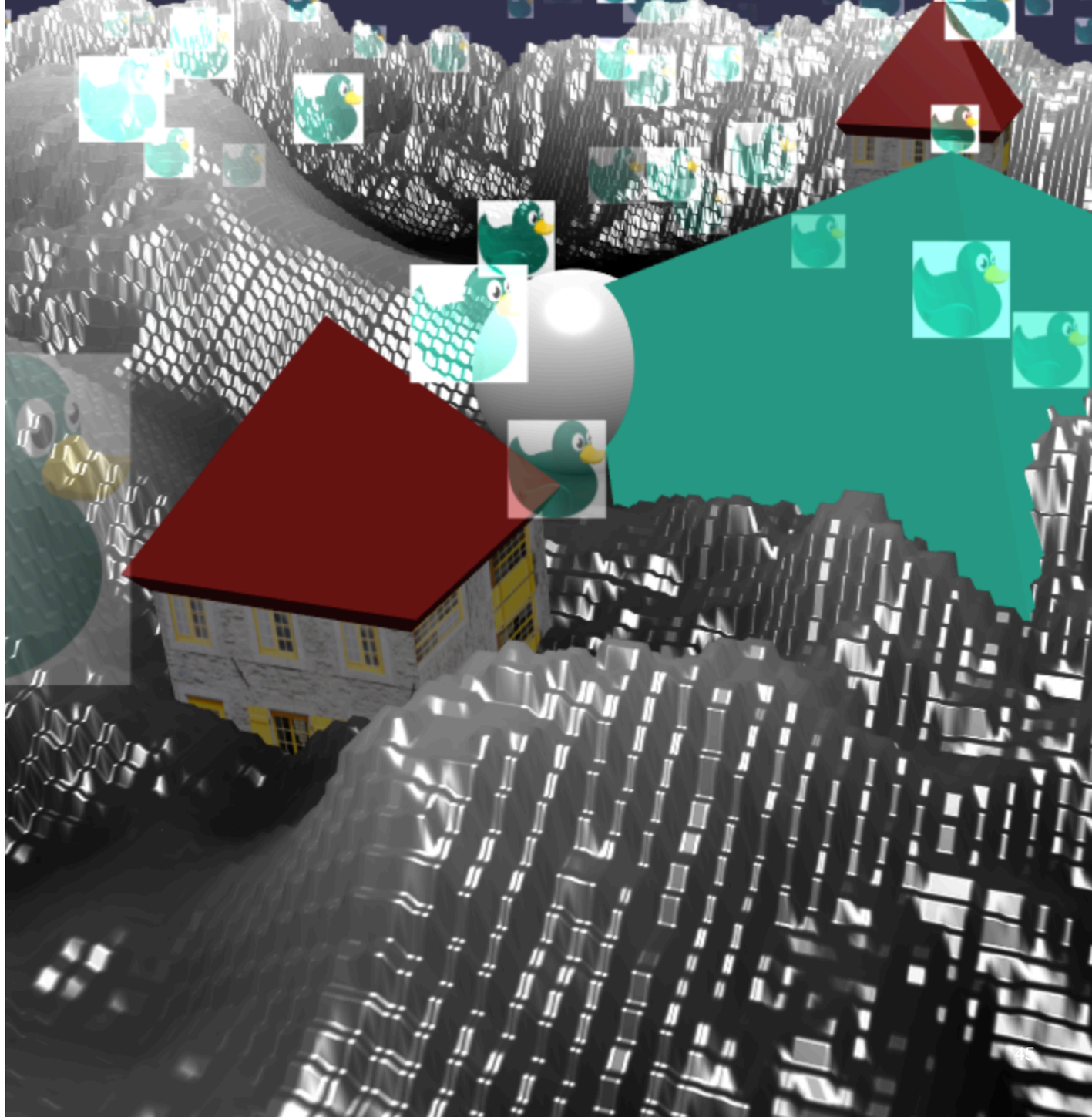
Performance

Community & Resources


It’s demo time :D

Give it a try!

Have fun,
go get crazy
with it
:)



QUESTIONS?

COME TALK TO US! :)

CHRISTINA

merelyChristina

ANNA

merelyAnna