

THREE.JS **17.04.2023.**
FMK_SUMMER_SEMESTER_2K23//[**GIT**](#)

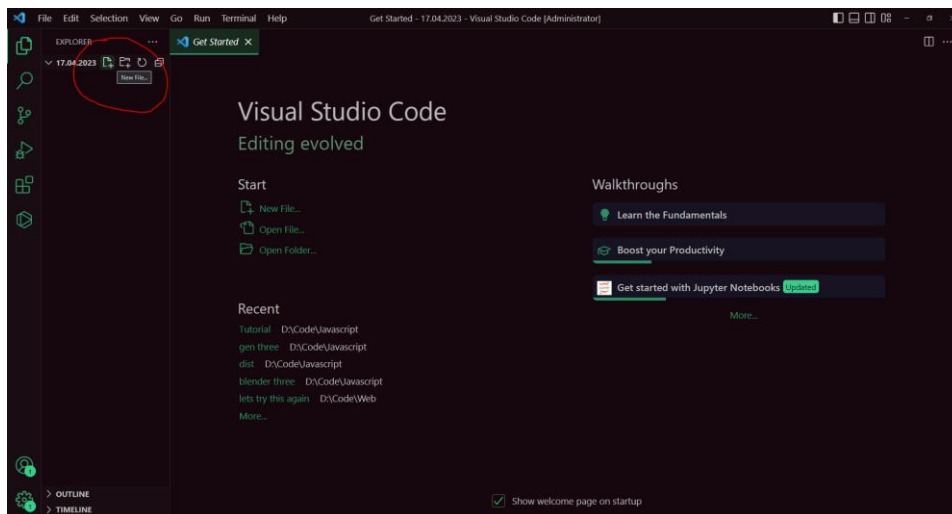
17.04.2023.

FMK_SUMMER_SEMESTER_2K23//GIT

[illegible]

PROJECT SETUP:

> Prvo je važno da napravimo naše html i javascript files.



> HTML file nazivamo index.html, dok javascript file nazivamo main.js

> U HTML file ispod `<body>` tag-a dodajemo sledeće dve skripte. (NAPOMENA *main.js* skripta treba da se zove onako kako ste nazvali vaš javascript file, dok druga skripta mora da bude identična).

```
<script src='https://cdnjs.cloudflare.com/ajax/libs/three.js/r83/three.min.js'></script>
```

```
<> index.html ● JS main.js
<> index.html > html > script
1 <!DOCTYPE html>
2 <html Lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <title>Document</title>
8 </head>
9 <body>
10
11 </body>
12     <script src="main.js"></script>
13     <script src='https://cdnjs.cloudflare.com/ajax/libs/three.js/r83/three.min.js'></script>
14 </html>
```

```
index.html JS main.js
index.html > html > body > canvas#bg
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10   <canvas id="bg"></canvas>
11 </body>
12   <script src="main.js"></script>
13   <script src="https://cdnjs.cloudflare.com/ajax/libs/three.js/r83/three.min.js"></script>
14 </html>
```

> Posto koristimo javascript „biblioteku“ moramo da je importujemo u js file.

[illegible]

- › Prvo moramo napraviti *scenu* u kojoj ćemo praviti i manipulirati 3D grafikom.
 - ›› To radimo tako što napravimo novu varijablu i pridajemo joj funkciju `new THREE.Scene()`;
 - ›› Možete promeniti pozadinu scene tako što vasa varijabli dodate atribut `.background.` (Boja mora biti u hexadecimalnoj vrednosti [primer sa slike je crna boja pozadine „0x000000“])
 - ›› [Ovde](#) možete da birate boje i da odmah dobijete njenu hexadecimalnu vrednost (samo zamenite # sa 0x i ctrl+V ostatak [primer #FF5633 → 0xff5633])

```
<> index.html JS main.js ●
JS main.js > ...
1 // imports
2 import * as THREE from "https://unpkg.com/three@0.127.0/build/three.module.js";
3
4 // setup
5 const scene = new THREE.Scene();
6 scene.background = new THREE.Color( 0x000000);
7
```

CAMERA:

>> Za one koje zanima: Kao i u blenderu postoje Perspective i Orthographic kamera.

```

<> index.html      JS main.js
JS main.js > ...
1  // imports
2  import * as THREE from "https://unpkg.com/three@0.127.0/build/three.module.js";
3
4  // setup
5  const scene = new THREE.Scene();
6  scene.background = new THREE.Color(0x000000);
7  const camera = new THREE.PerspectiveCamera(75, window.innerWidth / window.innerHeight, 0.1, 1000);
8  scene.add(camera);
9
10

```

> Menjanje pozicije kamere može se postići ili kroz korišćenje dimension-specific atributa, to jest da stavite za svaku osu novu liniju ili možete da uradite prosto `camera.position = new THREE.Vector3(x,y,z);` Mada, ako koristite starije verzije biblioteke može da se desi da interpreter nije u mogućnosti da renderuje kod. U tom slučaju koristite dimension-specific verziju.

[illegible]

RENDERER:

```
// renderer
const renderer = new THREE.WebGLRenderer({
  canvas: document.querySelector('#bg'), alpha: true
});
renderer.setPixelRatio(window.devicePixelRatio);
renderer.setSize(window.innerWidth, window.innerHeight);
renderer.render(scene, camera);
```

> Renderer renderuje scenu. Prekopirajte ovaj kod, ali pazite da u `.querySelector()` pravilno napišete ID vašeg canvasa (HTML canvas id iz project setup-a)

[illegible]

LIGHT:

> Na našoj sceni je mrkli mrak. Kada bismo dodali neku geometriju ništa ne bismo videli. Postoje bezbroj različitih „reflektora“ koje možete da koristite. Možete eksperimentisati i mešati razna svetla, ali nama je trenutno najvažnije da imamo ambijentalno svetlo koje će jednako osvetliti celu scenu.

```
22 // lights
23 const ambientLight = new THREE.AmbientLight(0xffffff);
24 scene.add(ambientLight);
25
```

► Boja svetla se može promeniti u vidu hexkoda (linija 23). Bela će naravno predstavljati boje kakve jesu, dok će crvena naravno naglasiti crvene tonove geometrije u sceni.

POINTLIGHT-u može da se manuelno podesi pozicija.

```
// lights
const ambientLight = new THREE.AmbientLight(0xffffff);
const pointLight = new THREE.PointLight(0xffaaff);
pointLight.position.set = (300,5,-500);
scene.add(ambientLight, pointLight);
```

[illegible]

GEOMETRY:

> Najzabavniji deo 3.js-a jeste pravljenje i manipulisanje 3D geometrijom. Jedan objekat se sastoji od određene vrste geometrije i materijala. Dokumentacija three.js-a ima listu

geometrija i sve parametre koji vam trebaju za rad. [GEOMETRIJA](#), [MATERIJALI](#).

```
29 // geometries
30 const geometry = new THREE.BoxGeometry(20,20,20);
31 const material = new THREE.MeshStandardMaterial({ color: 0xde7a3f,wireframe:false });
32 const pyro = new THREE.Mesh(geometry, material);
33 scene.add(pyro);
```

Za geometriju i materijal morate da napravite posebne varijable (slika iznad).

>> Line 30. je primer četvorougaoone geometrije gde imamo X,Y i Z parametre koje možemo menjati da dobijemo kocku ili pravougaonik.

>> Line 31. je primer standardnog materijala objekta. Uzima parametre boje (hexadecimalna vrednost) i wireframe-a (boolean vrednost (true ili false).

>> Line 32. je primer pravljenja 3D objekta uz pomoć geometrije i materijala. Ime varijable PYRO je samo primer. Varijabla može da se zove kako god vi želite (to se odnosi na većinu varijabli, ali radi konvencije geometry i material uglavnom ostaju isti osim, ako radite sa velikim brojem različitih objekata i geometrija)

>> Line 33. je najvažniji deo ovog snippeta, zato što ako ne napišemo ovu liniju koda nećemo dodati naš objekat na scenu. (Kao kada je ilustracija van artboarda u ilustratoru, a želimo da je eksportujemo u jpg/png/pdf etc.)

Kao što smo podesili poziciju kamere uz pomoć .position atributa, tako možemo da ga iskoristimo i za objekte. (primer dole)

```
35 pyro.position.x = 0;
36 pyro.position.y = 0;
37 pyro.position.z = -50;
```

Primer više oblika sa različitim geometrijama/materijalima:

```
29 // geometries
30 const geometry = new THREE.BoxGeometry(20,20,20);
31 const material = new THREE.MeshStandardMaterial({ color: 0xde7a3f,wireframe:false });
32 const pyro = new THREE.Mesh(geometry, material);
33 scene.add(pyro);
34
35 pyro.position.x = 0;
36 pyro.position.y = 0;
37 pyro.position.z = -50;
38
39 const geometry2 = new THREE.TorusGeometry(30,5,5,80);
40 const material2 = new THREE.MeshStandardMaterial({ color: 0xde7a3f,wireframe:false });
41 const pyro2 = new THREE.Mesh(geometry2, material2);
42 scene.add(pyro2);
43
44 pyro2.position.x = 0;
45 pyro2.position.y = 0;
46 pyro2.position.z = -50;
47
```

ANIMACIJA I „TRUE“ RENDERING:

```
48 // draw
49 function animate() {
50     requestAnimationFrame(animate);
51     |
52     renderer.render(scene, camera);
53 }
54
55 animate();
```

```

47
48 // draw
49 function animate() {
50     requestAnimationFrame(animate);
51     pyro.rotation.z += 0.01;
52     pyro.rotation.y += 0.02;
53
54     pyro2.rotation.x -= 0.01;
55     pyro2.rotation.y -= 0.02;
56     renderer.render(scene, camera);
57 }
58
59 animate();

```

[illegible]

Sada kada imamo završenu animaciju ili grafik koji želimo da se renderuje na sajtu, poželjno je da možemo da dodajemo tekst, slike i drugu html arhitekturu.

Prva stvar koju moramo da uradimo jeste da u html-u dodamo div van canvasa u kome ćemo izgraditi ostatak našeg sajta.

```
index.html # style.css JS main.js
```

```
# style.css > canvas
```

```
1  html, body {
2      width: 100%;
3      height: 100%;
4      margin: 0;
5  }
```

```

16
17     main {
18         width: 100vw;
19         color: ■ white;
20         z-index: 99;
21         position: absolute;
22         width: 100%;
23         margin: 0px auto;
24         padding: 120px 0px;
25         padding-bottom: 0px;
26         display: grid;
27         grid-template-columns: repeat(12, 1fr);
28     }
29
30     canvas {
31         position: fixed;
32

```

Happy designing !

CAMERA SCROLL:

FAQ:

THREE.JS **17.04.2023.**
FMK_SUMMER_SEMESTER_2K23//GIT