Kotlin a böngészőben

Szécsi László

3D Grafikus Rendszerek

2. előadás

HTML head

az oldal kitölti az ablakot

```
<!DOCTYPE html>
<html>
<head>
  <title>3gr</title>
  <meta name="viewport"
    content="width=device-width, initial-scale=1" />
  <meta http-equiv="Content-Type"
    content="text/html; charset=utf-8" />
  <link rel="stylesheet" type="text/css"
    href="css/style.css" />
</head>
```

elemek stílusai itt lesznek

folyt köv.

HTML body

ez a cimke (tag) egy szakaszt definiál az oldalon
ez egy HTML elem

```
<body>
  <div id="container">
    <canvas id="canvas"></canvas>
    <div id="overlay"></div>
  </div>

  <script src="kotlin.js"></script>
  <script src="js/WebGLMath.js"></script>
  <script src="output.js"></script>
</body>
</html>
```

vászon, erre rajzol a WebGL

ezt a HTML szakaszt szeretnénk a vászon fölé rajzoln
tartalmát a kódból módosítva
könnyen jeleníthetünk meg szöveget

JavaScript forráskód forrásfileok

CSS #container

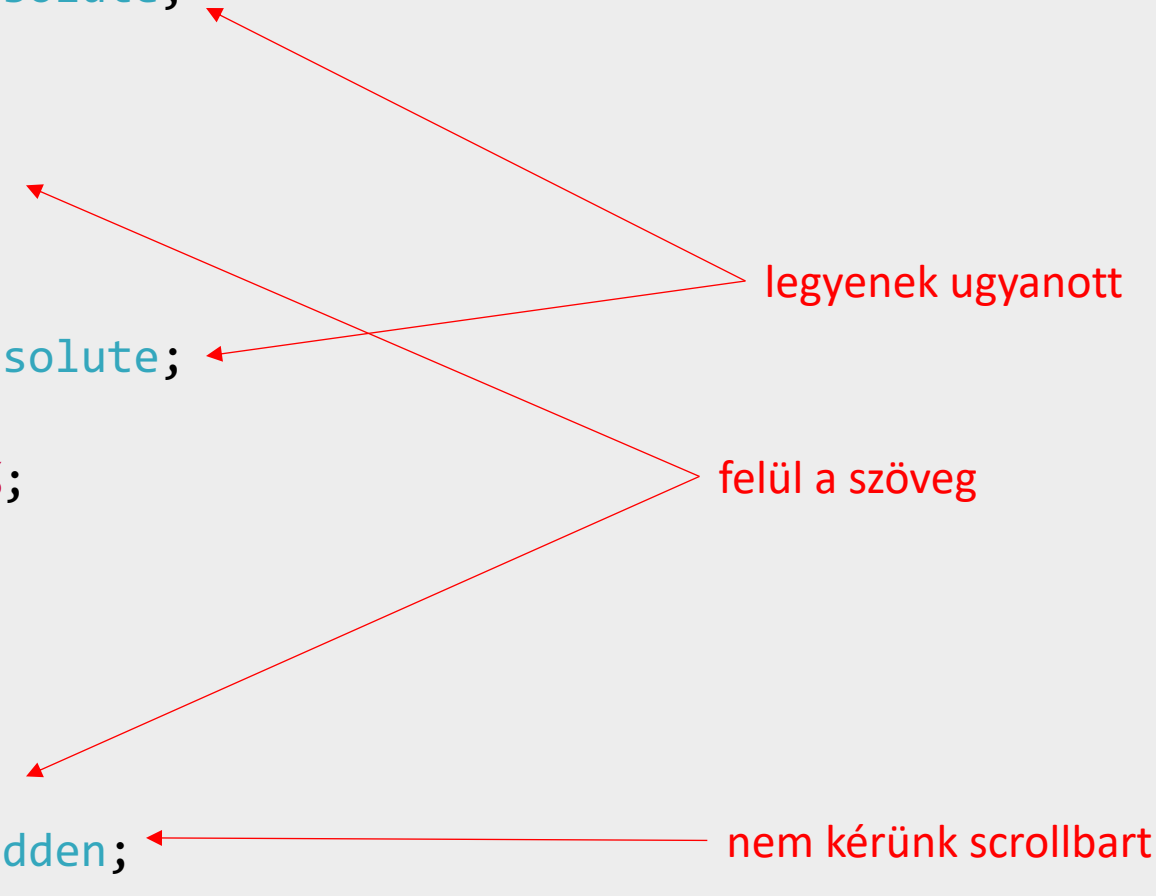
```
* {
  margin: 0;
  padding: 0;
}
#container {
  position: absolute;
  width: 100%;
  height: 100%;
  top: 0;
  left: 0;
  bottom: 0;
  right: 0;
  overflow: hidden;
}
```

töltse ki az oldalt

nem kérünk scrollbart

CSS #canvas, #overlay

```
#overlay {
  position: absolute;
  left: 10px;
  top: 10px;
  z-index: 20;
}
#canvas {
  position: absolute;
  width: 100%;
  height: 100%;
  top: 0;
  left: 0;
  bottom: 0;
  right: 0;
  z-index: 10;
  overflow: hidden;
}
```



DOM

- Document Object Model
- HTML címkék hierarchiája mint szülő-gyerek kapcsolatban álló objektumok rendszere

- győkér: `document`
  - **`getElementById`**

Window

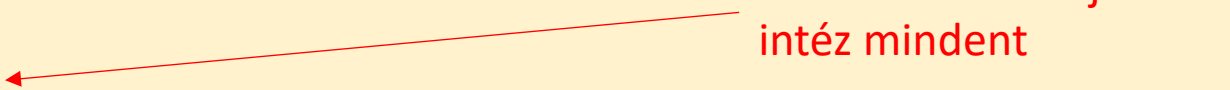
- `window`
  - globális objektum
    - ennek tulajdonsága minden globális
  - a dokumentumot megjelenítő ablak absztrakciója

```
> document.defaultView.document === document;
< true
```

Belépési pont

```
fun main() {
  val canvas = document.getElementById("canvas")
                                        as HTMLCanvasElement
  val overlay = document.getElementById("overlay")
                                        as HTMLDivElement
  overlay.innerHTML = """<font color="red">WebGL</font>"""

  try{
    val app = App(canvas, overlay)
    app.registerEventHandlers()
  } catch(e : Error) {
    console.error(e.message)
  }
}
```



App konstruktor, WebGL context

```
class App(
  val canvas : HTMLCanvasElement,
  val overlay : HTMLDivElement) {

  val gl = (
    canvas.getContext("webgl2") ?:
    throw Error("Browser does not support WebGL2")
    ) as WebGL2RenderingContext

  val scene = Scene(gl)
  init {
    resize()
  }
```

ez az objektum felel az erőforrások kezeléséért és kirajzolásáért

rajzolási felbontás beállítása
külön metódusban, mert ablakátméretezéskor is kell majd ugyanez

App::resize

rendering resolution

canvas size

```
fun resize() {
  canvas.width = canvas.clientWidth
  canvas.height = canvas.clientHeight
  scene.resize(gl, canvas)
}
```

Billentyűesemények feldolgozása

- nem a lenyomásra (typematic rate!) vagy felengedésre akarunk reagálni
- hanem minden frameben tudni, mi van épp lenyomva
- keysPressed objektum
  - stringek halmaza
  - elemek a gombok nevei
    - az S billentyű neve "S" legyen, ha lehet...
    - de az onkeydown eventjében keyCode van (S: 83???)
    - `keyNames`
      - elemek a string nevek, pl. "S", "DELETE", "SPACE", "F1"

Eseménykezelők

```
val keysPressed = HashSet<String>()


fun registerEventHandlers() {
  document.onkeydown =  {
    event : KeyboardEvent ->
    keysPressed.add( keyNames[event.keyCode] )
  }

  window.addEventListener("resize") { resize() }
  window.requestAnimationFrame { update() }
}
```

```
    "8", // [56]
    "9", // [57]
    "COLON", // [58]
    "SEMICOLON", // [59]
    "LESS_THAN", // [60]
    "EQUALS", // [61]
    "GREATER_THAN", // [62]
    "QUESTION_MARK", // [63]
    "AT", // [64]
    "A", // [65]
    "B", // [66]
    "C", // [67]
```

váltsunk ki egy rajzolást

ez a metódus fogja rajzolni a frame-et

App::update

```
fun update() {
    scene.update(gl, keysPressed)
    window.requestAnimationFrame { update() }
}
```

ütemezzük be a következő frame-et
ugyanezzel a kezelővel