

THREE.JS – INTERSEKCIA KURZORA

doc. Ing. Branislav Sobota, PhD.

Ing. Marián Hudák, Ing. Lenka Bubeňková

Katedra počítačov a informatiky, FEI TU v Košiciach

C 07

© 2024

CIELE CVIČENIA

- Three.js - objekty knižnice THREE Projector a THREE Raycaster.
- Three.js - deklarácia premenných a objektov za účelom pretínania trojrozmerných objektov kurzorom.
- Three.js - implementácia metód pre získavanie údajov o intersekcii kurzora s objektmi scény.
- Three.js - implementácia farebnej vizuálnej zmeny objektov pri pretnutí kurzorom.



1. *THREE.JS* – PRÍPRAVA BALÍČKA

- **Úloha:** Stiahnite si balíček „**Threejs_Intersekcia_kurzora.zip**“ z portálu Moodle KPI a predmetu Počítačova grafika.
- Obsah balíčka skopírujte do vášho projektu aby štruktúra vyzerala nasledovne:
 - WebGL_getStart
 - > css
 - > js
 - >> threejs
 - >> ThreeScene.js
 - models
 - > texture
 - index.html

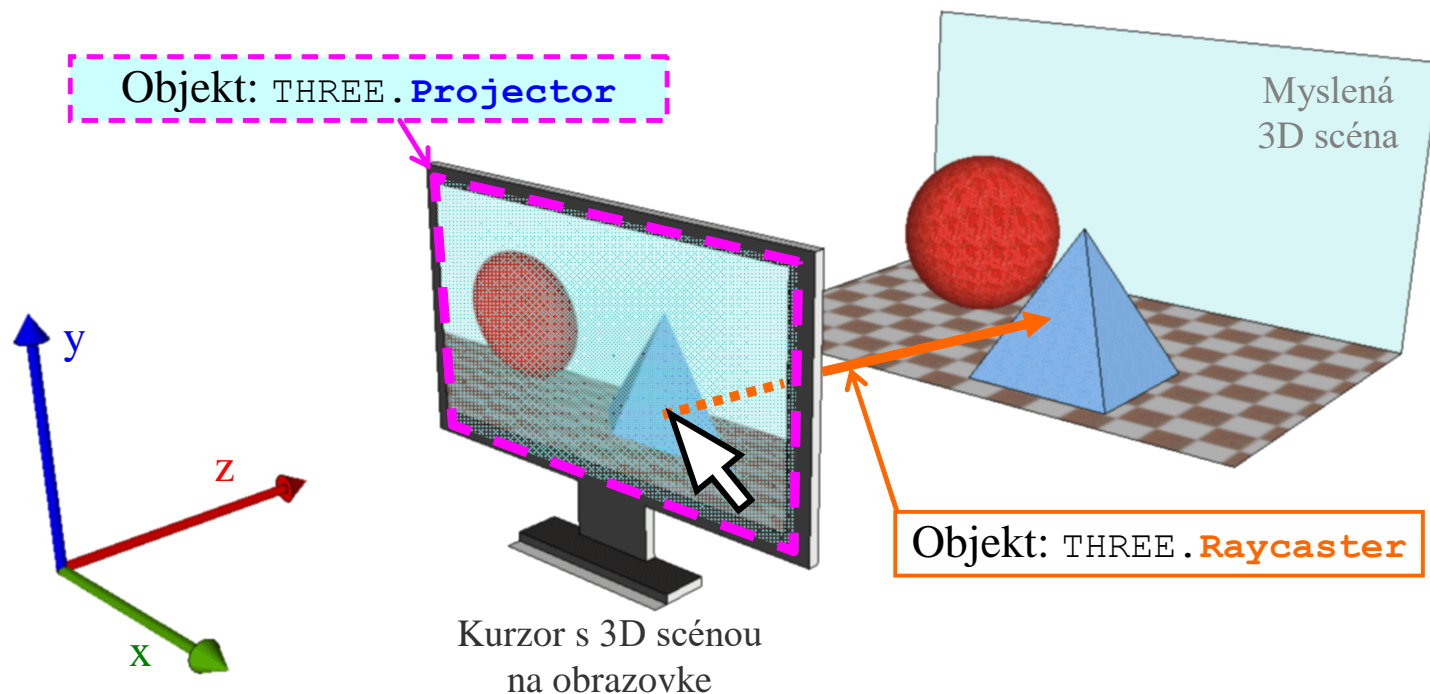
1. *THREE.JS* – SPUSTENIE BALÍČKA

1. Otvorte si skript „**ThreeScene.js**“ a html súbor „index.html“ vo vašom vývojovom prostredí.
2. Skontrolujte súbor „**index.html**“ .
Pokiaľ sa v elemente `<div id="canvas1"> . . . </div>` nenachádza skript referujúci na súbor „**ThreeScene.js**“, doplňte ho nasledujúcim riadkom kódu :

```
<script src="js/ThreeScene.js"></script>
```

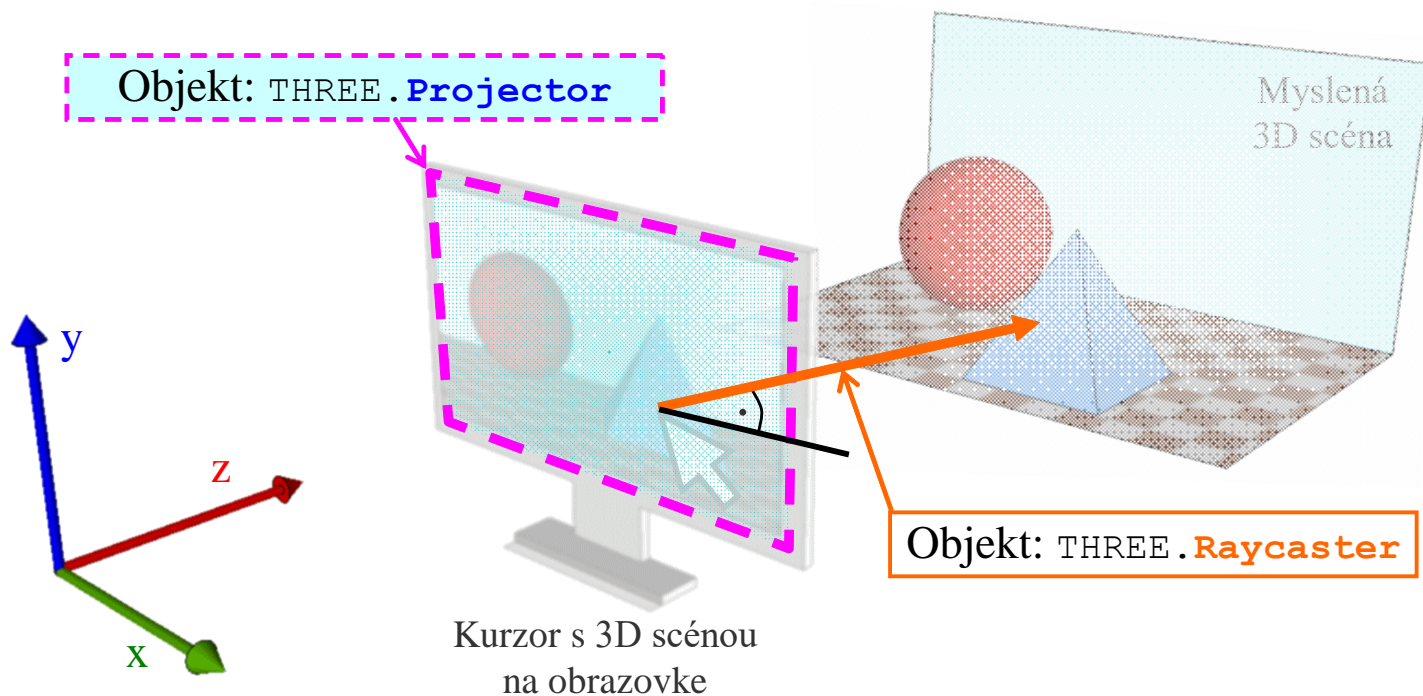
1. *THREE.JS* –*THREE.PROJECTOR* A *THREE.RAYCASTER*

- Objekty **Projector** a **Raycaster** sú do scény vkladané podľa nasledujúceho obrázku.



1. *THREE.JS* –*THREE.PROJECTOR* A *THREE.RAYCASTER*

- **Raycaster** vytvára myslenný lúč, vedený kolmo na obrazovku, v smere od kurzora do priestoru scény.
- **Projector** riadi prepočet medzi pozíciou kurzora na obrazovke a jeho umiestnením v trojrozmernej scéne.



2. THREE.JS - DEKLARÁCIA PREMENNÝCH A OBJEKTOV ZA ÚČELOM PRETÍNANIA TROJROZMERNÝCH OBJEKTOV KURZOROM.

- **Úloha:** doplňte premenné do skriptu „**ThreeScene.js**“
- V skripte „**ThreeScene.js**“ vložte nasledujúce **globálne premenné** :

```
var projector;
var mouse = { x: 0, y: 0 };
var INTERSECTED;
```



3. THREE.JS - IMPLEMENTÁCIA METÓD PRE ZÍSKAVANIE ÚDAJOV O INTERSEKCII KURZORA S OBJEKTMI SCÉNY.

- **Úloha:** rozšírte metódu **init()**

1. V skripte „**ThreeScene.js**“ doplňte na konci metódy **init()** nasledujúce riadky kódu :

```
projector = new THREE.Projector();
document.addEventListener( 'mousemove', onDocumentMouseMove, false );
```

- V metóde **init()** je dôležité vytvorenie objektu **THREE.Projector**, ktorého úlohou bude sledovanie pozície kurzora na obrazovke a následné vytvorenie projekcie potrebnej pre ovládanie lúča **THREE.Raycaster**.



3. THREE.JS - IMPLEMENTÁCIA METÓD PRE ZÍSKAVANIE ÚDAJOV O INTERSEKCII KURZORA S OBJEKTMI SCÉNY.

- **Úloha:** vytvorte metódu **onDocumentMouseMove(event)**:

1. V skripte „**ThreeScene.js**“ vytvorte metódu **onDocumentMouseMove(event)** :

```
function onDocumentMouseMove( event )
{
    mouse.x = ( event.clientX / window.innerWidth ) * 2 - 1;
    mouse.y = - ( event.clientY / window.innerHeight ) * 2 + 1;
}
```

- Táto metóda obsahuje vstupný parameter **event**, ktorý je odchytený počas kontinuálnej manipulácie s kurzorom.
- Následne sa získava pozícia kurzora v súradnicovej sústave webového prehliadača.



3. THREE.JS - IMPLEMENTÁCIA METÓD PRE ZÍSKAVANIE ÚDAJOV O INTERSEKCII KURZORA S OBJEKTMI SCÉNY.

- **Úloha:** rozšírte metódu **init()** o lokálne premenné.
- V skripte „**ThreeScene.js**“ doplňte na konci metódy update() nasledujúce riadky kódu:

```
var vector = new THREE.Vector3( mouse.x, mouse.y, 1 );
projector.unprojectVector( vector, camera );
var ray = new THREE.Raycaster( camera.position,
                               vector.sub(camera.position).normalize() );
var intersects = ray.intersectObjects( scene.children );
```

- **Opis premenných :**
 - **vector** – vytvorenie trojrozmerného vektora s počiatočnými súradnicami v mieste kurzora.
 - **ray** – vytvorenie lúča smerujúceho kolmo od kurzora do prostredia scény.
 - **intersects** – získavanie údajov o intersekcii objektu THREE.Raycaster s objektmi scény.

4. THREE.JS - IMPLEMENTÁCIA FAREBNEJ VIZUÁLNEJ ZMENY OBJEKTOV PRI PRETNUTÍ KURZOROM

- **Úloha:** rozšírte metódu **update()** o detekciu intersekcie.

1. V skripte „**ThreeScene.js**“ doplňte na konci metódy update() fragment kódu:

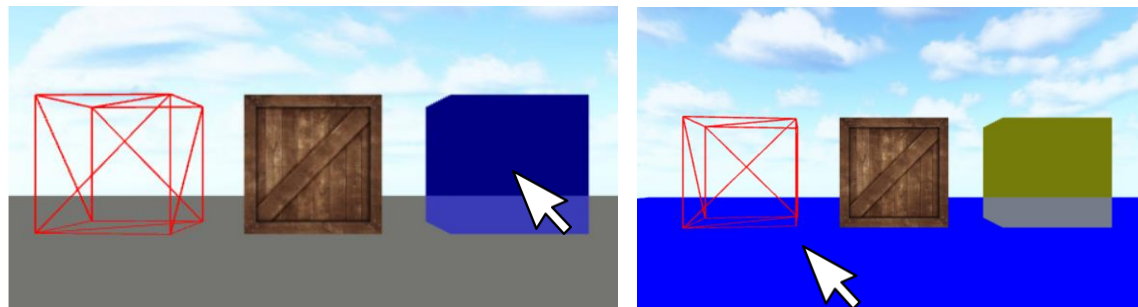
```
//sleduje sa intersekcia kurzora s okolitými objektmi
if ( intersects.length > 0 )
{ //ak je object detegovaný, zafarbi sa na zeleno, ostatne objekty majú
povodnu farbu
    if ( intersects[ 0 ].object !== INTERSECTED )
    {
        if ( INTERSECTED )
        INTERSECTED.material.color.setHex( INTERSECTED.currentHex );
        INTERSECTED = intersects[ 0 ].object;
        INTERSECTED.currentHex = INTERSECTED.material.color.getHex();
        INTERSECTED.material.color.setHex( 0x00FF00 );
    }
}
else
{ //po ukončení intersekcie sa objektu priradi povodna farba
    if ( INTERSECTED )
        INTERSECTED.material.color.setHex( INTERSECTED.currentHex );
    INTERSECTED = null;
}
```

4. THREE.JS - IMPLEMENTÁCIA FAREBNEJ VIZUÁLNEJ ZMENY OBJEKTOV PRI PRETNUTÍ KURZOROM

- Po vykonaní predošlej implementácie teraz otestujete jej správnosť. Spustíte si vizualizáciu a prechádzajte kurzorom cez objekty scény. Pokiaľ ste implementáciu zvládli úspešne, objekty sa budú zafarbovať na zeleno.



- Zmeňte implementáciu tak, aby sa objekty pri intersekcii s kurzorom zafarbili na modro.



4. THREE.JS – SPUSTENIE BALÍČKA

POZOR

Ak by po spustení nebola prítomná interakcia kurzora.
Vymažte alebo **zakomentujte** v súbore **index.html** nasledujúci odkaz :

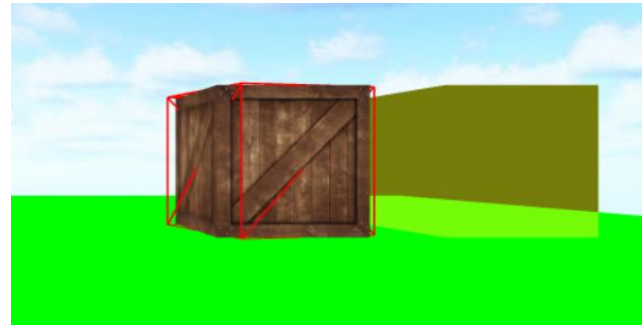
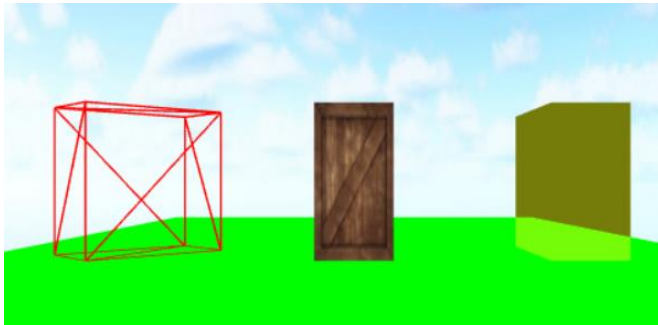
```
<link rel="stylesheet" href="styles.css">
```

a nezabudnite pomocou klávesu F5 vymazať cache pamäť
prehliadača

DOPLŇUJÚCE ÚLOHY

1. Vyskúšajte meniť niektoré z atribútov objektov pri intersekcii s kurzorom :

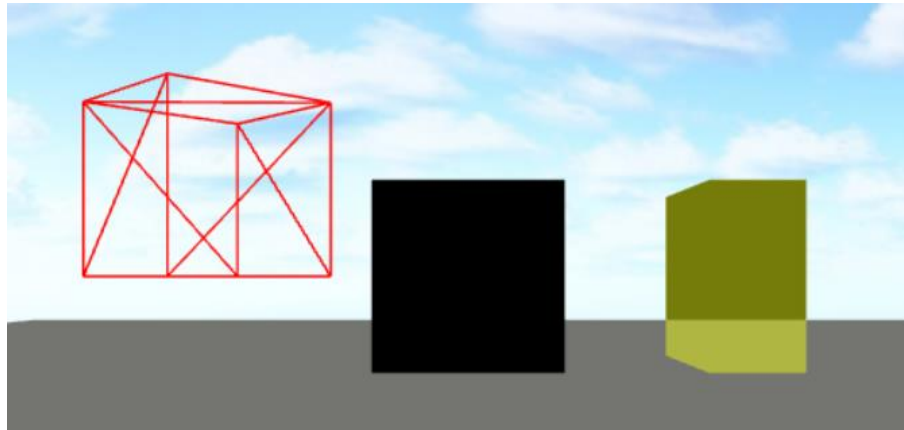
- **Zmena mierky** : `INTERSECTED.scale.x = 0.5;`
- **Posun**: `INTERSECTED.position.x += 1;`
- **Otočenie**: `INTERSECTED.rotation.x += 50;`



DOPLŇUJÚCE ÚLOHY

2. Na scéne sú umiestnené objekty: **cube1**, **cube2**, **cube3**. Vyskúšajte implementovať intersekciiu kurzorom tak, aby:

- Prenutie objektu **cube1** zmení jeho farbu na **čiernu**,
- Prenutie objektu **cube2** zmenší jeho mierku o **polovicu**,
- Prenutie objektu **cube3** posunie objekt o **0.5 jednotky smerom nahor**.



DOPLŇUJÚCE ÚLOHY

3. Zmeňte textúru objektu na lirkis.jpg pri intersekcii s kurzorom pre cube1, cube2 a cube3:



ÚLOHY NA SAMOSTATNÉ RIEŠENIE

- Na scéne sú umiestnené objekty: **cube1**, **cube2**, **cube3**.
 Vyskúšajte implementovať intersekciiu kurzorom tak, aby ste dokázali využiť alebo meniť vzťah *Rodič-Potomok*.



Q & A

branislav.sobota@tuke.sk
lenka.bubenkova@tuke.sk

Katedra počítačov a informatiky, FEI TU v Košiciach

© 2024