****



**عنوان: بازی پینگ پنگ با Three JS**

**ارائه کنندگان : سارا خالوباقری، زهرا اکبرزاده، سمانه بامداد**

**اساتید راهنما : دکتر مینوفام و سامعی**

**تاریخ ارائه: 11/3/1398**

**فهرست مطالب**

عنوان صفحه

1- چکیده .................................................................................................................................................................................... 4

2- توضیحات کد ....................................................................................................................................................................... 5

1-2- تابع init ................................................................................................................................................................... 5

2-2- تابع animate ........................................................................................................................................................ 7

3-2- تابع onDocumentKeyUp.............................................................................................................................. 7

4-2- تابع ballMovment............................................................................................................................................ 7

5-2- تابع collision....................................................................................................................................................... 9

3- مرجع .................................................................................................................................................................................... 10

4- نمرات ................................................................................................................................................................................... 11

**1- چکیده**

پروژه یک بازی پینگ پنگ ساده ی دو نفره است که با استفاده از فریمورک Three Js نوشته شده و با استفاده از دکمه های صفحه کلید دو بازیکن بر روی صفحه جا بجا می شوند تا از برخورد توپ با دیواری که در پشتشان قرار دارد جلوگیری شود و اگر توپ با دیواره پشت بازیکن برخورد کند برای حریف یک امتیاز ثبت می شود.

پینگ پنگ، بازی دونفره، Three JS

**2-** **توضیحات کد**

**1-2- تابع init**

ابتدا یک نور محیط با رنگ سفید تعریف کرده و به صحنه اضافه کردیم.

**var** ambientLight = **new THREE**.*AmbientLight*(0xffffff, 1);  
 ***scene***.add(ambientLight);

صحنه پس زمینه (میز پینگ پنگ) را با رنگ زرد و ابعاد 100 در 180 ایجاد کرده و به صحنه در مختصات (0و0و0) اضافه کردیم.

**var** surfaceGeometry = **new THREE**.*PlaneGeometry*(100, 180);  
 **var** surfaceMaterial = **new THREE**.*MeshPhongMaterial*({**color**: 0xc6b10a, **side**: **THREE**.DoubleSide});  
 ***surfaceMesh*** = **new THREE**.*Mesh*(surfaceGeometry, surfaceMaterial);  
 ***surfaceMesh***.**rotateX**(***Math***.**PI** / 2);  
 ***surfaceMesh***.**position**.**y** = 0;  
 ***surfaceMesh***.**position**.**x** = 0;  
 ***surfaceMesh***.**position**.**z** = 0;  
 ***scene***.add(***surfaceMesh***);

دیواره سمت راست را با رنگ آبی و ابعاد 3 در 180 و ضخامت 5 ایجاد کرده و به صحنه در مختصات (0و0و50) اضافه کردیم.

**var** RightWallGeometry = **new THREE**.*CubeGeometry*(3, 180, 5);  
 **var** RightWallMaterial = **new THREE**.*MeshPhongMaterial*({**color**: 0x0aaaa9, **side**: **THREE**.DoubleSide});  
 **var** RightWallMesh = **new THREE**.*Mesh*(RightWallGeometry, RightWallMaterial);  
 RightWallMesh.**rotateX**(***Math***.**PI** / 2);  
 RightWallMesh.**position**.**y** = 0;  
 RightWallMesh.**position**.**x** = 50;  
 RightWallMesh.**position**.**z** = 0;  
 ***scene***.add(RightWallMesh);

دیواره سمت چپ را با رنگ آبی و ابعاد 3 در 180 و ضخامت 5 ایجاد کرده و به صحنه در مختصات (0و0و50-) اضافه کردیم.

**var** LeftWallGeometry = **new THREE**.*CubeGeometry*(3, 180, 5);  
 **var** LeftWallMaterial = **new THREE**.*MeshPhongMaterial*({**color**: 0x0aaaa9, **side**: **THREE**.DoubleSide});  
 **var** LeftWallMesh = **new THREE**.*Mesh*(LeftWallGeometry, LeftWallMaterial);  
 LeftWallMesh.**rotateX**(***Math***.**PI** / 2);  
 LeftWallMesh.**position**.**y** = 0;  
 LeftWallMesh.**position**.**x** = -50;  
 LeftWallMesh.**position**.**z** = 0;  
 ***scene***.add(LeftWallMesh);

دیواره پشت بازیکن اول را با رنگ آبی و ابعاد 3 در 100 ایجاد کرده و به صحنه در مختصات (87-و0و0) اضافه کردیم.

**var** PlayersBackWall1Geometry = **new THREE**.*CubeGeometry*(100, 3, 20);  
 **var** PlayersBackWall1Material = **new THREE**.*MeshPhongMaterial*({**color**: 0x0aaaa9, **side**: **THREE**.DoubleSide});  
 ***PlayersBackWall1Mesh*** = **new THREE**.*Mesh*(PlayersBackWall1Geometry, PlayersBackWall1Material);  
 ***PlayersBackWall1Mesh***.**rotateX**(***Math***.**PI** / 2);  
 ***PlayersBackWall1Mesh***.**position**.**y** = 0;  
 ***PlayersBackWall1Mesh***.**position**.**x** = 0;  
 ***PlayersBackWall1Mesh***.**position**.**z** = -87;  
 ***PlayersBackWall1Mesh***.**name**=**"p1Wall"**;  
 ***collidableMeshList***.push(***PlayersBackWall1Mesh***);  
 ***scene***.add(***PlayersBackWall1Mesh***);

دیواره پشت بازیکن دوم را با رنگ آبی و ابعاد 3 در 100 ایجاد کرده و به صحنه در مختصات (87و0و0) اضافه کردیم.

**var** PlayersBackWall2Geometry = **new THREE**.*CubeGeometry*(100, 3, 20);  
 **var** PlayersBackWall2Material = **new THREE**.*MeshPhongMaterial*({**color**: 0x0aaaa9, **side**: **THREE**.DoubleSide});  
 ***PlayersBackWall2Mesh*** = **new THREE**.*Mesh*(PlayersBackWall2Geometry, PlayersBackWall2Material);  
 ***PlayersBackWall2Mesh***.**rotateX**(***Math***.**PI** / 2);  
 ***PlayersBackWall2Mesh***.**position**.**y** = 0;  
 ***PlayersBackWall2Mesh***.**position**.**x** = 0;  
 ***PlayersBackWall2Mesh***.**position**.**z** = 87;  
 ***PlayersBackWall2Mesh***.**name**=**"p2Wall"**;  
 ***collidableMeshList***.push(***PlayersBackWall2Mesh***);  
 ***scene***.add(***PlayersBackWall2Mesh***);

بازیکن اول را با رنگ آبی و ابعاد 3 در 25 ایجاد کرده و به صحنه در مختصات (82-و2و0) اضافه کردیم.

**var** Player1Geometry = **new THREE**.*CubeGeometry*(25, 3, 5);  
 **var** Player1Material = **new THREE**.*MeshPhongMaterial*({**color**: 0x0aaaa9, **side**: **THREE**.DoubleSide});  
 ***Player1Mesh*** = **new THREE**.*Mesh*(Player1Geometry, Player1Material);  
 ***Player1Mesh***.**rotateX**(***Math***.**PI** / 2);  
 ***Player1Mesh***.**position**.**y** = 2;  
 ***Player1Mesh***.**position**.**x** = 0;  
 ***Player1Mesh***.**position**.**z** = -82;  
 ***Player1Mesh***.**name**=**"P1"**;  
 ***collidableMeshList2***.push(***Player1Mesh***);  
 ***scene***.add(***Player1Mesh***);

بازیکن دوم را با رنگ آبی و ابعاد 3 در 25 ایجاد کرده و به صحنه در مختصات (82و2و0) اضافه کردیم.

**var** Player2Geometry = **new THREE**.*CubeGeometry*(25, 3, 5);  
 **var** Player2Material = **new THREE**.*MeshPhongMaterial*({**color**: 0x0aaaa9, **side**: **THREE**.DoubleSide});  
 ***Player2Mesh*** = **new THREE**.*Mesh*(Player2Geometry, Player2Material);  
 ***Player2Mesh***.**rotateX**(***Math***.**PI** / 2);  
 ***Player2Mesh***.**position**.**y** = 2;  
 ***Player2Mesh***.**position**.**x** = 0;  
 ***Player2Mesh***.**position**.**z** = 82;  
 ***Player2Mesh***.**name**=**"P2"**;  
 ***collidableMeshList2***.push(***Player2Mesh***);  
 ***scene***.add(***Player2Mesh***);

توپ را ایجاد کرده و به آن بافت دادیم و به صحنه در مختصات (78و5و0) اضافه کردیم.

**var** textureball = **new THREE**.*TextureLoader*().load(**'texture/perlin-512.png'**);  
 **var** ballGeo = **new THREE**.*SphereGeometry*(2.5, 16, 16);  
 **var** ballMat = **new THREE**.*MeshPhongMaterial*({**shading**: **THREE**.FlatShading, **shininess**: 65 , **map**:textureball});  
 ***ball*** = **new THREE**.*Mesh*(ballGeo, ballMat);  
 ***ball***.**position**.**x** = 0;  
 ***ball***.**position**.**y** = 5;  
 ***ball***.**position**.**z** = 78;  
 ***scene***.add(***ball***);

**2-2- تابع animate**

با بررسی مقدار متغییر stepplayer1 بازیکن اول را بر روی محور X به سمت راست و چپ حرکت می دهیم.

**if** (***stepplayer1*** == 3) {  
 **if** (***Player1Mesh***.**position**.**x** < 36)  
 ***Player1Mesh***.**position**.**x** += 4;  
 ***stepplayer1*** = 0;  
} **else if** (***stepplayer1*** == 2) {  
 **if** (***Player1Mesh***.**position**.**x** > -36)  
 ***Player1Mesh***.**position**.**x** -= 4;  
 ***stepplayer1*** = 0;  
}

با بررسی مقدار متغییر stepplayer2 بازیکن دوم را بر روی محور X به سمت راست و چپ حرکت می دهیم.

**if** (***stepplayer2*** == 3) {  
 **if** (***Player2Mesh***.**position**.**x** < 36)  
 ***Player2Mesh***.**position**.**x** += 4;  
 ***stepplayer2*** = 0;  
} **else if** (***stepplayer2*** == 2) {  
 **if** (***Player2Mesh***.**position**.**x** > -36)  
 ***Player2Mesh***.**position**.**x** -= 4;  
 ***stepplayer2*** = 0;  
}

**3-2- تابع onDocumentKeyUp**

با فشردن دکمه l و j از صفحه کلید مقدار متغییر stepplayer1 میکند؛ که در نهایت منجر به جا بجا شدن بازیکن اول میشود. ( دکمه l بازیکن را به سمت راست می برد و دکمهj بازیکن را به سمت چپ می برد.)

**if** (event.**keyCode** == 74) {  
 ***stepplayer1*** = 2;  
} **else if** (event.**keyCode** == 76) {  
 ***stepplayer1*** = 3;  
}

با فشردن دکمه f و s از صفحه کلید مقدار متغییر stepplayer2 میکند؛ که در نهایت منجر به جا بجا شدن بازیکن دوم میشود. ( دکمه f بازیکن را به سمت راست می برد و دکمهs بازیکن را به سمت چپ می برد.)

**if** (event.**keyCode** == 83) {  
 ***stepplayer2*** = 2;  
} **else if** (event.**keyCode** == 70) {  
 ***stepplayer2*** = 3;  
}

**4-2- تابع ballMovment**

برای حرکت توپ به صورت افقی و در محور x به طوری که از دیوارهای راست و چپ عبور نکند، دو محدوده و دو فلگ در نظر گرفتیم و بررسی کردیم تا زمانی که موقعیت توپ در محور x عددی کوچکتر یا مساوی 43 باشد و فلگ a ]که نشان دهنده آخرین برخورد توپ در محور افقی است، اگر با دیوار سمت چپ برخورد کرده باشد مقدار یک دارد در غیر این صورت مقدار صفر[ مقدار یک باشد پس توپ در راستای مثبت محور x حرکت کند. و با قرار گرفتن توپ در موقعیت 43 مقدار b یک و a صفر می شود.

**if** (***ball***.**position**.**x** <= 43 && ***a*** == 1) {  
 ***ball***.**position**.**x** += 1;  
 **if** (***ball***.**position**.**x** == 43) {  
 ***a*** = 0;  
 ***b*** = 1;  
 }

و در ادامه بررسی کردیم تا زمانی که موقعیت توپ در محور x عددی بزرگتر یا مساوی 43- باشد و فلگ b ]که نشان دهنده آخرین برخورد توپ در محور افقی است، اگر با دیوار سمت راست برخورد کرده باشد مقدار یک دارد در غیر این صورت مقدار صفر[ مقدار یک باشد پس توپ در راستای منفی محور x حرکت کند. و با قرار گرفتن توپ در موقعیت 43- مقدار a یک و b صفر می شود.

}  
**if** (***ball***.**position**.**x** >= -43 && ***b*** == 1) {  
 ***ball***.**position**.**x** -= 1;  
 **if** (***ball***.**position**.**x** == -43) {  
 ***a*** = 1;  
 ***b*** = 0;  
 }  
}

برای حرکت توپ به صورت عمودی و در محور z به طوری که از دیواره های پشت بازیکن اول و دوم عبور نکند، دو محدوده و دو فلگ در نظر گرفتیم و بررسی کردیم تا زمانی که موقعیت توپ در محور z عددی کوچکتر یا مساوی 85 باشد و فلگ d ]که نشان دهنده آخرین برخورد توپ در محور عمودی است، اگر با دیواره پشت بازیکن دوم برخورد کرده باشد مقدار یک دارد در غیر این صورت مقدار صفر[ مقدار یک باشد پس توپ در راستای مثبت محور z حرکت کند. و با قرار گرفتن توپ در موقعیت 85 مقدار c یک و d صفر می شود.

**if** (***ball***.**position**.**z** <= 85 && ***d*** == 1) {  
 ***ball***.**position**.**z** += 1;  
 **if** (***ball***.**position**.**z** == 85) {  
 ***c*** = 1;  
 ***d*** = 0;  
 }  
}

و در ادامه بررسی کردیم تا زمانی که موقعیت توپ در محور z عددی بزرگتر یا مساوی 85- باشد و فلگ c ]که نشان دهنده آخرین برخورد توپ در محور عمودی است، اگر با دیواره پشت بازیکن اول برخورد کرده باشد مقدار یک دارد در غیر این صورت مقدار صفر[ مقدار یک باشد پس توپ در راستای منفی محور z حرکت کند. و با قرار گرفتن توپ در موقعیت 85- مقدار d یک و c صفر می شود.

**if** (***ball***.**position**.**z** >= -85 && ***c*** == 1) {  
 ***ball***.**position**.**z** -= 1;  
 **if** (***ball***.**position**.**z** == -85) {  
 ***c*** = 0;  
 ***d*** = 1;  
 }  
}

**5-2- تابع collision**

بررسی می کند اگر برخورد با دیواره پشت بازیکن دوم اتفاق افتاده باشد، ابتدا صدای شکستن پخش می شود و بعد برای بازیکن اول یک امتیاز مثبت در نظر میگیرد و به امتیازهای قبلی او اضافه می شود.

**if**(wallP1.**length**>0){  
  
 ***listener*** = **new THREE**.*AudioListener*();  
 ***camera***.add(***listener***);  
 ***sound*** = **new THREE**.*Audio*(***listener***);  
 ***audioloader*** = **new THREE**.*AudioLoader*();  
 ***audioloader***.load(**'sound/3d.mp3'**, **function** (buffer) {  
 ***sound***.setBuffer(buffer);  
 ***sound***.setVolume(0.5);  
 ***sound***.play();  
 });  
  
 ***p1Score***++;  
}

بررسی می کند اگر برخورد با دیواره پشت بازیکن اول اتفاق افتاده باشد، ابتدا صدای شکستن پخش می شود و بعد برای بازیکن دوم یک امتیاز مثبت در نظر میگیرد و به امتیازهای قبلی او اضافه می شود.

**if**(wallP2.**length**>0) {  
  
 ***listener*** = **new THREE**.*AudioListener*();  
 ***camera***.add(***listener***);  
 ***sound*** = **new THREE**.*Audio*(***listener***);  
 ***audioloader*** = **new THREE**.*AudioLoader*();  
 ***audioloader***.load(**'sound/3d.mp3'**, **function** (buffer) {  
 ***sound***.setBuffer(buffer);  
 ***sound***.setVolume(0.5);  
 ***sound***.play();  
 });  
  
 ***p2Score***++;  
}

**6-2- تابع playerCollision**

این تابع بطور کلی چک می کند که آیا توپ با بازیکن ها برخورد کرده است یا خیر. برای این منظور بررسی کردیم تا زمانی که موقعیت توپ در محور z عددی بزرگتر یا مساوی 74 باشد و برخورد با بازیکن دوم صورت گرفته باشد، فلگ c و a مقدار یک می شود و فلگ b و d صفر می شود. (برای حرکت توپ در جهت مخالف) و در ادامه بررسی کردیم تا زمانی که موقعیت توپ در محور z عددی کوچکتر یا مساوی 74- باشد و برخورد با بازیکن اول صورت گرفته باشد، فلگ b و d مقدار یک می شود و فلگ c و a صفر می شود. (برای حرکت توپ در جهت مخالف)

**3- مرجع**

WWW.Threejs.org/docs

|  |  |  |  |
| --- | --- | --- | --- |
|  | سارا خالوباقری | زهرا اکبرزاده | سمانه بامداد |
| تمرین | 80% | 10% | 10% |
| پروژه | 70% | 20% | 10% |

**4- نمرات**