



به نام خدا

پروژه گرافیک کامپیوتری

گردآورندگان :

رضا جهانگیری ۹۵۱۱۱۳۰۲۳

امیر احمدی طامه ۹۵۱۱۱۳۰۰۷

سروش سیف پور ۹۵۱۱۱۳۰۵۰

استاد راهنما :

جناب آقای مینوفام

جناب آقای فراز سامعی

بهار ۹۸

## فهرست

۲	شرح پروژه .....
۲	قوانین بازی .....
۴	توضیحات لازم برای کد پروژه .....
۵	متدها (توابع): .....
۶	تابع INIT .....
۷	تابع INTCANNON .....
۸	کار توابع دیگر نوشته شده در کد .....
۸	تابع ONWINDOWRESIZE: .....
۸	تابع WELCOME: .....
۸	تابع UPDATEPHYSICS: .....
۸	تابع CHECKED: .....
۸	تابع REL: .....
۸	تابع RENDER: .....
۹	تابع ONDOCUMENTKEYUP: .....
۹	تابع BALLMOVE: .....
۱۰	تصاویری از محیط بازی .....

## شرح پروژه

پروژه مذکور Bottle Shoot نام دارد .

در این بازی ، ۳ توپ بر روی زمین روی محور  $x$  ها در حال حرکت است و روبروی توپها تعدادی بطری بر روی میزی قرار گرفته است.

بازیکن باید با زمانبندی مناسب توپها را به گونه ای پرتاب کند تا به بطری ها اصابت کند و امتیاز جمع کند.

## قوانین بازی

بازیکن ۳ توپ در اختیار دارد که توپ اول نزدیک به میز، توپ دوم کمی عقبتر و توپ سوم نیز عقبتر از توپ دوم قرار دارد.

توپ ها بر روی زمین در جهت محور  $x$  ها در حال رفت و برگشت می باشند و بازیکن باید با زمانبندی مناسب و زدن دکمه Space توپها را به سمت بطری ها پرتاب کند.

هر بطری که به زمین می افتد بازیکن ۱۰ امتیاز کسب میکند ولی اگر با توپ به بطری بزند ولی بطری روی میز بماند هیچ امتیازی ندارد.

در بازی یک بطری خاص (Bonus Bottle) وجود دارد که تشخیص آن کمی سخت است و اگر بازیکن آن بطری را پیدا کرده و بزند ۱۰۰ امتیاز میگیرد و برنده بازی میشود.

پیغام های بازی در صورت کسب امتیازهای مختلف :

امتیاز ۱۰۰ و بیشتر از ۱۰۰ : Excellent : You Shotted Bonus Bottle And Get 100 Scores

امتیاز ۴۰ و بیشتر از ۴۰ : Good : Your Score Is More Than 40

امتیاز کمتر از ۴۰ : Not Bad : Your Score Is Less Than 40

امتیاز صفر : Game Over

در صورت تکمیل پروژه میتوان به آن امکانات مختلفی اضافه کرد از جمله:

۱. در صورت کسب حدنسب امتیاز ، بازیکن به مرحله بعد برود.
۲. در مراحل بالاتر با اضافه کردن مانع و مه کار بازیکن را سخت تر کند.
۳. بازیکن با جمع کردن امتیاز بتواند توپ مورد پسند خود را از فروشگاه بازی خریداری کند.
۴. بازیکن بتواند با جمع کردن امتیاز به جای ۳ توپ، ۴ توپ در هر مرحله پرتاب کند.
۵. ...

## توضیحات لازم برای کد پروژه

```
<body onload="welcome()">
<!-- Baraye Namayesh Score va Welcome -->
138 <div id="myModal" class="modal">
139 <div class="modal-content">
140 <span class="close">&times;</span>
141 <h2 style="font-size:20px">Game Over!!!</h2>
142 </div>
143 </div>
144
145
146 <div id="myModalwc" class="modalwc">
147 <div class="modal-contentwc">
148 <span class="closewc">&times;</span>
149 <h2 style="font-size:20px">
150 Wellcome to Bottle Shoting!<br>
151 This game create by Reza,Amir,Sorouh.
152 </h2>
153 </div>
154 </div>
155
156 <div id="myModalwin" class="modal">
157 <div class="modal-content">
158 <span class="close">&times;</span>
159 <h2 style="font-size:20px" class="Score">** Excellent : You Shooted Bonus Bottle And Get 100 Scores **</h2>
160 </div>
161 </div>
162
163 <div id="myModal1" class="modal">
164 <div class="modal-content">
165 <span class="close">&times;</span>
166 <h2 style="font-size:20px" class="Score">** Good : Your Score Is More Than 40 **</h2>
167 </div>
168 </div>
169
170 <div id="myModal2" class="modal">
171 <div class="modal-content">
172 <span class="close">&times;</span>
173 <h2 style="font-size:20px" class="Score">** Not Bad : Your Score Is Less Than 40 **</h2>
174 </div>
175 </div>
176
177 <!-- Baraye Namayesh Score va Welcome -->
```

در این قسمت box پیغام امتیازات و نیز پیغام شروع بازی نوشته شده است که در اول و آخر بازی نمایش داده میشود.

متغیرهای global : معنای تمام متغیر ها درکد واضح است فقط قابل ذکر است که تمام متغیرهایی که برای بخش Physic اجسام موردنیاز است باید global بشوند چون در چند تابع جداگانه مورد استفاده قرار میگیرند.

متغیر countBall برای تعیین توپ های پرتاب شده و متغیر Score برای تعیین امتیاز بازیکن.

آرایه ها : آرایه balls برای ذخیره ۳ توپ ساخته شده در تابع `init`.

آرایه `ballsBodys` برای ذخیره فیزیک توپها در تابع `initcannon`.

آرایه `bottleArray` برای ذخیره کردن بطری های ساخته شده در تابع `init`.

آرایه `bottleArrayBody` برای ذخیره فیزیک بطری ها در تابع `initcannon`.

## متدها (توابع):

در برنامه ۳ تابع اصلی داریم که بازی با آنها اجرا میشود :

```
222     var modal ;
223
224     init();
225
226     initCannon();
227
228     animate();
229
```

```
230
231     animate();
```

## تابع init

در تابع init تمام متغیرهای پایه threejs یعنی scene و camera و renderer تعریف شده است.

همچنین تمام اجزا و عناصر صحنه در این تابع تعریف و نوشته میشوند از جمله :

کف صحنه، تمام بطری ها ، تمام توپ ها ، تمام دیوارها ، نورهایی که برای دنیای خود در نظر میگیریم، سایه اجسام، موقعیت اجسام، جعبه هایی که در صحنه قرار میدهیم و ... .

در تابع init برای گذاشتن تصویر یک گراند ، دنیای خود را به صورت یک کره بزرگ در نظر گرفته و برای کره تصویر ۳۶۰ درجه و خاصیت DoubleSide گذاشته ایم تا فضای اطراف ساخته شود.

```
245
246     var texture = THREE.ImageUtils.loadTexture('texture/skysphere3.jpg');
247
248     texture.magFilter = THREE.NearestFilter;
249     texture.minFilter = THREE.NearestFilter;
250
251     var sphereGeometry = new THREE.SphereGeometry(150,30,30);
252     var sphereMaterial = new THREE.MeshPhongMaterial( {
253         map:texture,
254         shading: THREE.SmoothShading,
255         side: THREE.DoubleSide
256     } );
257     var space = new THREE.Mesh(sphereGeometry, sphereMaterial);
258     space.position.set(0,0,-20);
259     scene.add(space);
260
```

نکته : در تابع init توپها را در آرایه balls و بطری ها را در آرایه bottleArray قرار داده ایم.

## تابع initCannon

برای نوشتن این تابع نیاز به کتابخانه ی cannon.js داریم.

در این تابع اول متغیر world را مقداردهی کرده ایم سپس برای جهان خود یک جاذبه به سمت

yهای منفی نوشته ایم تا دنیای بازی ما مانند دنیای واقعی دارای نیروی جاذبه زمین باشد.

سپس در این تابع به تعریف فیزیک اجسام پرداخته ایم و برای هر جسمی که در init تعریف

کرده ایم و میدانیم که به فیزیک نیاز دارد ، در اینجا فیزیک مورد نیازش را نوشته ایم.

برای مثال تعریف فیزیک برای یک بطری و یک توپ به صورت زیر است :

```
ballShape = new CANNON.Sphere(0.75);
ballBody = new CANNON.Body({mass: 0.5});
ballBody.addShape(ballShape);
ballBody.angularDamping = 0.5;
ballBody.position.set(-14,0,5);
ballsBodys.push(ballBody);

world.add(ballsBodys[0]);
```

```
bottle1Shape = new CANNON.Box(new CANNON.Vec3(0.5,1.0, 0.5));
bottle1Body = new CANNON.Body({mass: 0.5});
bottle1Body.addShape(bottle1Shape);

bottle1Body.angularDamping = 0.75;
bottle1Body.position.set(-18.25,5.0,-40.0);
world.add(bottle1Body);
bottleArrayBody.push(bottle1Body);
```

پس از تعریف فیزیک اجسام، برای مطابقت دادن هر جسم تعریف شده در init به فیزیک مربوط به آن ،تابع

updatePhysics توسط تابع animate صدا زده میشود .



## کار توابع دیگر نوشته شده در کد

### تابع `onWindowResize` :

این تابع برای موقعی است که کاربر اندازه صفحه مرورگر خود را تغییر میدهد.

### تابع `welcome` :

این تابع برای نمایش پیغام `welcome` در هنگام شروع بازی میباشد.

### تابع `updatePhysics` :

پس از تعریف فیزیک اجسام، برای مطابقت دادن هر جسم تعریف شده در `init` به فیزیک مربوط به آن، تابع `updatePhysics` توسط تابع `animate` صدا زده میشود .

### تابع `checked` :

این تابع برای چک کردن این است که آیا بطری های روی میز بر روی زمین افتاده اند یا نه . به ازای هر بطری ای که به زمین افتاده است ۱۰ امتیاز به `score` اضافه میکند و اینکه اگر بطری `bonus` روی زمین افتاده باشد ۱۰۰ امتیاز به `score` اضافه میکند.

### تابع `rel` :

برای `reload` کردن صفحه .

### تابع `render` :

برای ترسیم دنیا.

## تابع onDocumentKeyUp :

این تابع هنگامی که کلیدی زده بشود فراخوانی میشود. در این تابع چک میکند که آیا کلید زده شده space است یا نه. اگر space بود و ۳ تا توپ پرتاب نشده بود، یک توپ پرتاب میکند. اگر ۳ تا توپ پرتاب شده بود، تابع checked را صدا میزند تا امتیاز و پیغام مناسب برای بازیکن نمایش داده شود. کد window.setTimeout (Rel, 4000) یعنی تابع rel را ۴۰۰۰ میلی ثانیه دیگر فراخوانی کن.

## تابع ballMove :

برای حرکت توپها بر روی محور Xها به سمت چپ و راست.

نام اعضای گروه	درصد مشارکت در تمرینات	درصد مشارکت در پروژه
رضا جهانگیری	٪۳۴	٪۳۴
امیر احمدی	٪۳۳	٪۳۳
سروش سیف پور	٪۳۳	٪۳۳

در ادامه تصاویری از محیط بازی قرار داده شده است.

## تصاویری از محیط بازی

