



بسم الله الرحمن الرحيم

وزارت علوم، تحقیقات و فناوری



نام پروژه :

Space Galaxy

نام برنامه نویسان :

محمد امین پورمرادیان

مهتاب شهرتی

فاطمه کاهه ء

نام اساتید:

سید امیر هادی مینوفام

فراز سامعی

بهار ۹۸



## مقدمه:

بازی GCA Crash که در آن یک جت در سمت چپ و تعدادی مگس در سمت راست وجود دارد را باید طوری هدایت کنید که جت از بین موانع (مگس ها) با موفقیت عبور کند.

## ابزار ها و زبان های برنامه نویسی استفاده شده

### HTML:

عبارت HTML مخفف Hyper Text Markup Language به معنی زبان نشانه گذاری فوق متن است. Html زبان استاندارد طراحی صفحات وب است و کلیه کدهای صفحه اعم از طرف سرور و طرف مشتری در نهایت به کدهای HTML تبدیل شده و توسط مرورگر نمایش داده می شوند.

کامپایلر های زبان های برنامه نویسی سروری در نهایت کد های خود را برای نمایش به کد اچ تی ام ال تبدیل میکنند و برای مرورگر میفرستند تا به کاربران نمایش داده شود.

HTML یک زبان نشانه گذاری است ، به این معنی که بخش های مختلف توسط اجزایی به نام تگ از هم جدا شده ، که هر کدام دارای کاربرد و خواص مربوط خود هستند . این تگ ها به مرورگر اعلام می کنند که هر بخش از صفحه چه نوع عنصری است و باید به چه صورت نمایش داده شود .



مرورگرهای وب، که قادر به درک و تفسیر برچسب‌های HTML هستند، تک‌تک آن‌ها را از داخل سند HTML خوانده و سپس محتوای آن صفحه را نمایان‌سازی (Render) می‌کنند.

HTML زبان برنامه‌نویسی نیست، بلکه زبانی برای نشانه‌گذاری ابرمتن است و اساساً برای ساخت‌مند کردن اطلاعات و جدایش اجزای منطقی یک نوشتار به کار می‌رود. از سوی دیگر، HTML را نباید به عنوان زبانی برای صفحه‌آرایی یا نقاشی صفحات وب به کار بُرد؛ این وظیفه اکنون بر دوش فناوری‌های دیگری همچون CSS است.

## Java Script:

Java Script یک زبان اسکریپت نویسی است که بیشتر با کدهای HTML در ارتباط است و دقیقاً همانند کدهای HTML روی پلت فرم‌های مختلف اجرا می‌شود یا به عبارتی به وسیله مرورگرهای وب interpret می‌شود.

اگرچه نت اسکریپت سازنده Java Script است اما در واقع Java Script به سیستم عامل یا Platform خاصی وابسته نیست و با اکثر مرورگرهای پیشرفته از جمله Internet Explorer4 و نسخه‌های بعدی مایکروسافت کار می‌کند.

البته مرورگر IE مستقیماً Java Script را پشتیبانی نمی‌کند و دارای زبان اسکریپت نویسی مخصوص به خود "Jscript" است که اکثر ویژگی‌های Java Script را پشتیبانی می‌کند.

## 1-امکانات و قابلیت های جاوا اسکریپت:



طراحان صفحات وب می توانند با استفاده از Function ها و Object های آماده و از پیش تعریف شده جاوا اسکریپت قابلیت های زیادی را برای صفحات وب ایجاد کنند.

-می توان با استفاده از کدهای گرافیکی انیمیشن ایجاد کرد و همچنین صفحاتی را طراحی کرد که کاربر به دلخواه قادر به جابجایی یا تغییر تصاویر گرافیکی باشد.

## ۲- تفاوت جاوا و جاوا اسکریپت:

- shock: در عین حال که جاوا اسکریپت توانایی های بسیاری در زمینه ایجاد و طراحی صفحات وب دارد به علت وجود بعضی از محدودیت ها در آن , تنها برای نوشتن برنامه های کوچک و ساده در صفحات وب بکار می رود.

-بر خلاف جاوا که برنامه های آن قبل از اجرا باید کامپایل شود و به بایت کد تبدیل شود برنامه های جاوا اسکریپت نیازی به کامپایل برای اجرا ندارند و در همان لحظه اجرا بوسیله مرورگر خوانده شده و interpret می شوند.

-گرچه می توان بوسیله جاوا اسکریپت یک پرسشنامه یا فرم را به server فرستاد اما جاوا اسکریپت قدرت ایجاد ارتباط متقابل بین server و client را به اندازه جاوا ندارد.

## مفهوم گرافیک کامپیوتری :

گرافیک کامپیوتری به تصاویر و فیلم هایی اطلاق می شود که با استفاده از کامپیوتر ساخته می شوند.

گرافیک کامپیوتر یک حوزه گسترده و جدید در علوم کامپیوتر است.



برخی از مباحث مهم در گرافیک کامپیوتری عبارتند از طراحی رابط کاربری، عناصر گرافیکی، گرافیک برداری، مدلسازی سه بعدی، سایه سازی، طراحی GPU، تصویر سازی سطوح ضمنی به روش رهگیری پرتو، و همچنین بینایی کامپیوتری است. اساس گرافیک کامپیوتری به علوم پایه در زمینه های هندسه، نور و فیزیک وابسته است.

گرافیک کامپیوتری وظیفه نمایش طرح و تصویر را به طور مؤثر و قابل فهم برای کاربر به عهده دارد. همچنین برای پردازش داده های تصویری دریافت شده از دنیای فیزیکی استفاده میشود. ابداع گرافیک کامپیوتری، تأثیر بسزایی بر انواع رسانه ها داشته و بطور کلی موجب تحول انیمیشن، فیلم، تبلیغات، بازی های ویدئویی و طراحی گرافیکی شده است.

## ۱- مفاهیم و اصول

تصاویر معمولاً توسط دستگاه هایی همچون دوربینها، آینه ها، لنزها، تلسکوپها، میکروسکوپها و غیره ایجاد میشوند. تصاویر دیجیتال شامل تصاویر برداری و تصاویر رستر هستند، اما تصاویر رستر متداولتر میباشند.

## ۲- پیکسل

در تصویر سازی دیجیتال، پیکسل (یا عنصر تصویر) یک نقطه از تصویر رستر است. پیکسلها در یک شبکه منظم دو بعدی قرار گرفته اند، و اغلب به وسیله نقاط یا مربعها توصیف میشوند. هر پیکسل، یک نمونه از تصویر اصلی است، و طبیعتاً نمونه های بیشتر، تصویر دقیقتری از عکس اصلی به دست میدهند.

## ۳- رندرینگ

رندرینگ به معنای تولید یک تصویر دو بعدی از یک مدل سه بعدی توسط برنامه های کامپیوتری است. یک فایل صحنه شامل اشیاء تعریف شده توسط یک زبان یا ساختاری از داده است؛ این فایل میتواند شامل هندسه، زاویه دید، اطلاعات مربوط به بافت، نورپردازی و سایه پردازی برای توصیف یک صحنه مجازی باشد. داده های داخل فایل صحنه سپس به یک برنامه رندرینگ ارسال میشوند تا پردازش شده و به یک تصویر



دیجیتال یا فایل گرافیک رستر تبدیل شود. برنامه رندرینگ معمولاً به صورت داخلی در نرم افزار گرافیک کامپیوتری قرار دارد و در موارد دیگر به شکل پلاگین یا نرم افزارهای کاملاً مجزا عرضه میشود. واژه "رندرینگ" ممکن است به مفهوم "پردازش هنری" یک صحنه باشد. اگرچه جزئیات فنی روشهای رندرینگ، متفاوت هستند چالشهای تولید یک تصویر دو بعدی از مدل سه بعدی ذخیره شده در فایل صحنه، به صورت یک نقشه کلی با عنوان "خط لوله گرافیکی" در کنار یک ابزار رندرینگ، نظیر GPU ارائه میشوند GPU. ابزاری است که میتواند به CPU در محاسبات کمک کند. اگر نیاز باشد یک صحنه تحت نورپردازی مجازی واقعی به نظر رسد لازم است نرم افزار رندرینگ، معادله رندرینگ را حل کند. اگرچه معادله رندرینگ تمام پدیده های نوری را در نظر نمیگیرد، اما یک مدل نورپردازی کلی برای تصاویر پردازشی کامپیوتری (CGI) محسوب می شود. واژه رندرینگ همچنین برای توصیف فرایند محاسبات مربوط به افکتهای در ویرایش یک ویدیو برای تولید ویدئوی خروجی نهایی به کار میرود.

### مفهوم three.js:

یکی از جذابیت هایی که می تواند در یک صفحه وب وجود داشته باشد، ایجاد احوام سه بعدی و اصولاً قرار دادن کاربر در یک محیط سه بعدی است. اما در تولید صفحات وب یکی از چالش های بزرگ، حجم زیاد احوام و هم چنین فرایند پیچیده تولید احوام در نرم افزارهای سه بعدی و وارد نمودن آن ها در صفحات وب می باشد.

Three.js یک کتابخانه جاوا اسکریپت می باشد که بر پایه WebGL بنا شده و با استفاده از آن می توان احوام سه بعدی را به راحتی در صفحات وب لود نمود و بدین ترتیب تجربه ای متفاوت را به کاربر منتقل کرد.



### چه زمانی بازی را بردید؟

اگر جت توانست بعد از اینکه از همه مگس ها عبور کرد و با مانع برخورد نکرد یعنی برنده شدید . و پس از برد ، صفحه refresh میشود تا دوباره بتوان بازی جدید را از سر گرفت.

### چه زمانی بازی را باختید؟

اگر جت در حین اینکه از مگس ها عبور می کرد با مانعی برخوردی داشت ، collision رخ داده و بازنده محسوب میشوید .  
و پیغام you lose score را دریافت میکنید.  
و پس از باخت ، صفحه refresh میشود تا دوباره بتوان بازی جدید را از سر گرفت.







**بازی:** در این پروژه ما ۳ تا **folder** داریم:

(1

**:Lib**

کتابخانه ها در آن نگهداری میشود:

در این بازی از کتابخانه گرافیکی Three.js برای نمایش و پردازش های گرافیکی کاربرد دارد.

از کتابخانه OrbitControls.js برای کنترل بخش های مختلف بازی که مثلا با موس کار میکند جهت debug کاربرد دارد که بعنوان تست در برنامه استفاده کردیم. ( به طور پیش فرض غیر فعال است)  
از کتابخانه Detector.js برای برخورد ها یا همان Collision استفاده شده.

(2

**:Models** ( object ها را از سایت [clara.io](http://clara.io) دانلود کردیم)

مدل های ما در آن نگهداری میشود.

مثلا aliens که حاوی مدل مگس میباشد که برنامه نویس میتواند هر تعداد که خواست اضافه کند در قالب آرایه و سطح بازی را سخت تر کند.

و mainjet که حاوی مدل هواپیمای ماست.

(۳

**:Textures**

حاوی عکس back ground ماست.



پروژه ی ما بصورت functional هست. هدف ما از این کار ، جلوگیری از زدن کدهای تکراری و نیز مدیریت و تسلط بهتر و پروژه را بخش های کوچکی می شکنیم .  
شامل:

### Function init()

کل پروژه اینجا انجام میشود. یعنی هرچی ما نوشتیم در نهایت برمیگرده به init

در این بخش تمام تنظیمات مربوط به دوربین و صحنه انجام می شود و همچنین تمام مدل های پس از بارگذاری کامل در این بخش صدا زده می شوند و به صحنه اضافه می شوند.

### Function animate()

مربوط میشه به threejs و مربوط به بخش انیمیشن میباشد.

نحوه عملکرد درون بازی به این صورت است که در یک حلقه Animation در هر ثانیه ۶۰ بار اجرا میشود که وضعیت بازی را بررسی میکند که چهار وضعیت کلی شروع، بازی، نتیجه برد و نتیجه باخت میباشد.

در این بخش حرکت مگس ها به سمت جت انجام می شود و همچنین فانکشن Update در این بخش فراخوانی می شود که در هر لحظه وضعیت برخورد بررسی شود.

### Function jetmodel()

در این بخش ما مدل جت خود را فراخوانی میکنیم و تنظیمات اولیه مانند چرخش در زاویه مناسب را انجام می دهیم. سپس در function init آن را فراخوانی میکنیم.



## Function aliensmodel ()

این متد دارای ۱ ورودی به اسم count می باشد. این مقدار برابر با تعداد مگس هایی می باشد که قرار است به صحنه اضافه شود. هرچه تعداد مگس ها بیشتر باشد بازی طولانی تر و سخت تر می شود.

سپس در متد init آن را فراخوانی میکنیم و به صحنه اضافه می کنیم.

## Function update()

مربوط به بخش Collision میباشد که اگر جت با مانعی برخورد کرد اینجا بررسی و کنترل شود.

ما برای انجام برخورد، ابتدا به تمام مدل های خودمان یک Mesh اضافه کردیم تا بتوانیم از برخورد این mesh ها ، collision را تشخیص دهیم.

البته این mesh ها کاملاً از دید کاربر مخفی هستند.

## keyDownTextField

در این بخش ما جت خود را کنترل میکنیم. در واقع با دریافت رویداد صفحه کلید ، keycode ورودی را دریافت و بر اساس آن جت را کنترل میکنیم.

با دکمه W به سمت بالا و S به سمت پایین حرکت میکند.

## امتیاز دهی:

در mytime تعیین میشود.

نحوه ی امتیاز دهی به این صورت است که به ازای هر ۱ ثانیه بازی جت ما ۱۰۰۰ امتیاز دریافت میکند.

در واقع یک تایمر داریم که این زمان را کنترل می کند. در صورت برد و باخت امتیاز دریافت شده به کاربر نمایش داده می شود.



`getRndInteger(min, max)`

این متد دو ورودی دریافت می کند و یک عدد تصادفی از مقدار min تا مقدار max تولید می کند.

هدف از استفاده این متد ایجاد یک مکان تصادفی برای هر کدام از مدل های ایجاد شده است.