

آشنایی با موتورهای بازی سازی و ساختار آنها

ارائه دهنده : امیرحسین موسوی

amirhosein.avagames@gmail.com

- بوجود آمدن موتورهای بازی سازی
- چرا از موتور بازی سازی استفاده کنیم
- بخش های اصلی موتورهای بازی سازی
- انتخاب موتور بازی سازی
- پرسش و پاسخ

- بوجود آمدن موتورهای بازی سازی

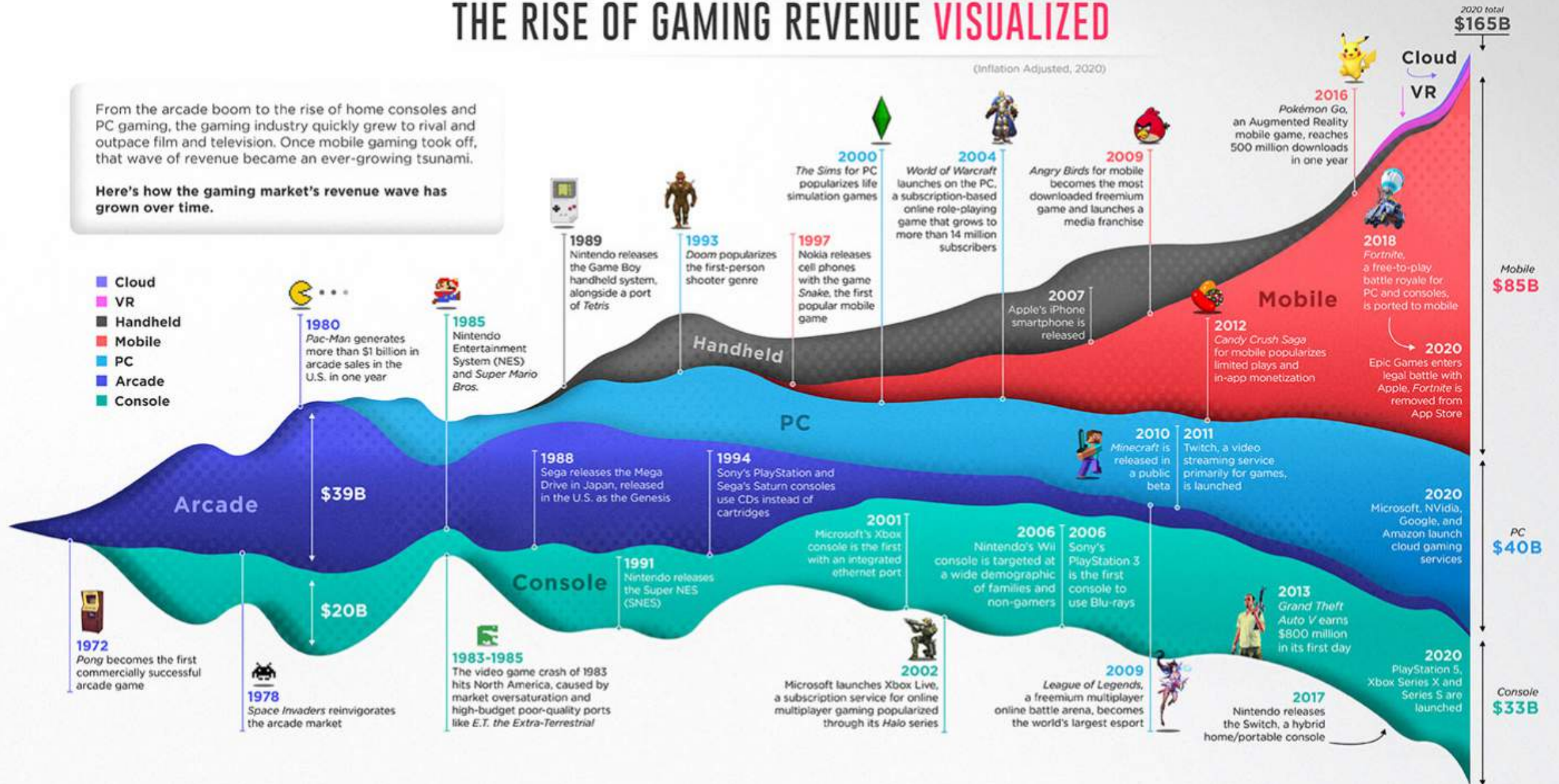
مرور تاریخچه ی صنعت ویدیو گیم و تاثیر آن بر بوجود آمدن موتور بازی سازی

THE RISE OF GAMING REVENUE VISUALIZED

(Inflation Adjusted, 2020)

From the arcade boom to the rise of home consoles and PC gaming, the gaming industry quickly grew to rival and outpace film and television. Once mobile gaming took off, that wave of revenue became an ever-growing tsunami.

Here's how the gaming market's revenue wave has grown over time.

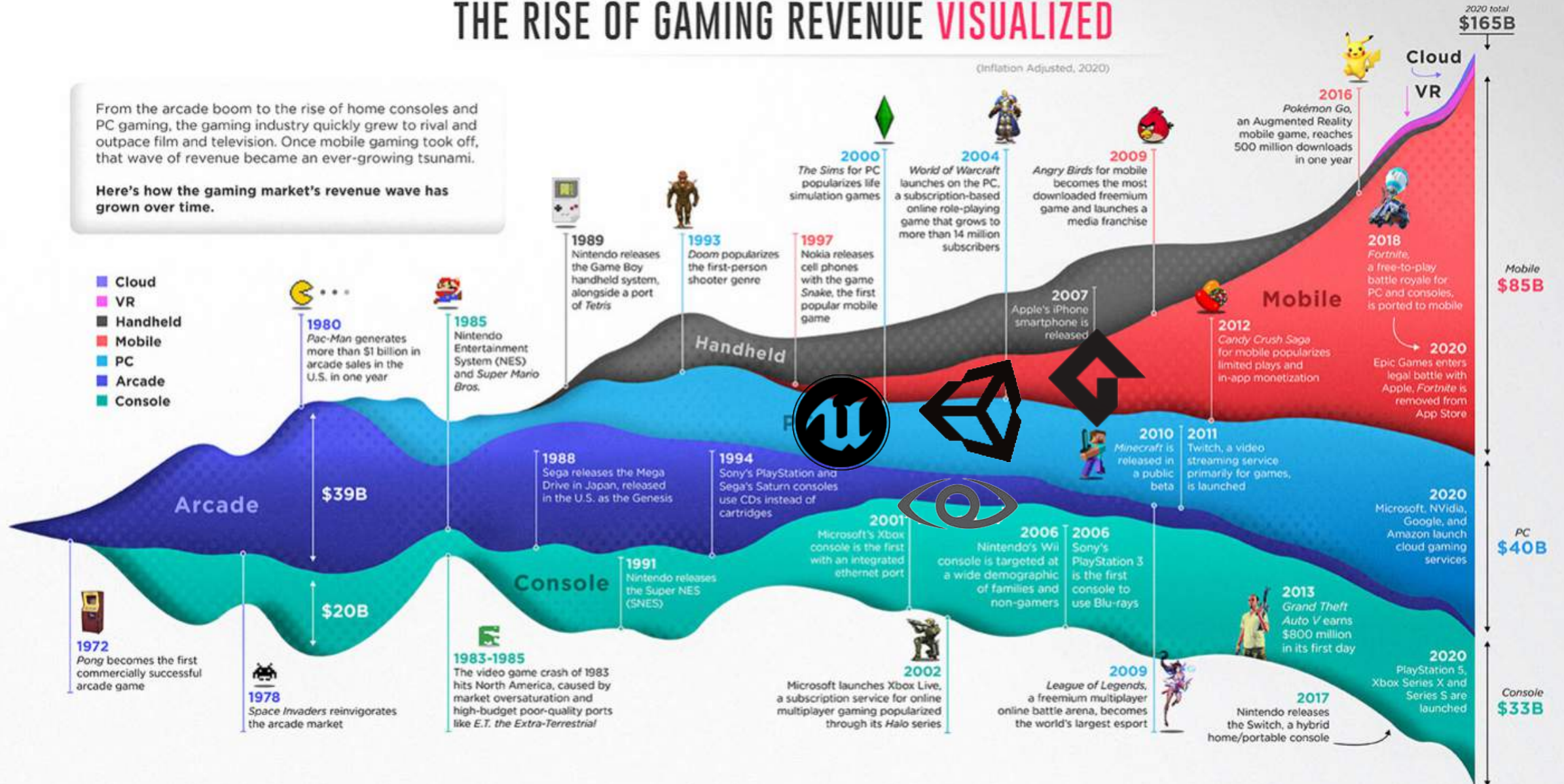


THE RISE OF GAMING REVENUE VISUALIZED

(Inflation Adjusted, 2020)

From the arcade boom to the rise of home consoles and PC gaming, the gaming industry quickly grew to rival and outpace film and television. Once mobile gaming took off, that wave of revenue became an ever-growing tsunami.

Here's how the gaming market's revenue wave has grown over time.



دسته بندی از نظر روند تولید :

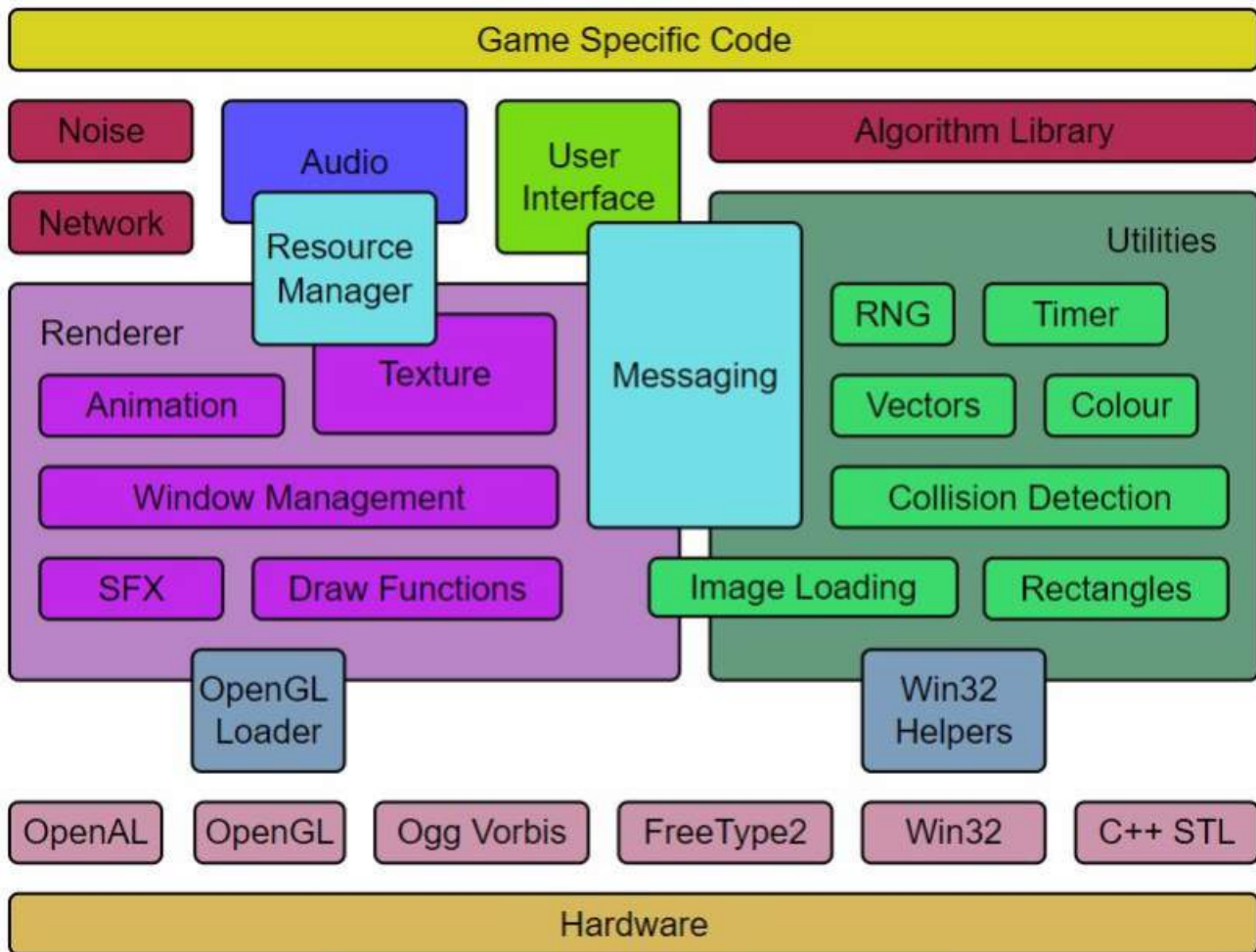
- تولید شده برای استفاده عمومی
..., unity, unreal, game maker

- برای تولید محصولات خود شرکت
RE Engine(capcom), Frostbyte(dice), Source(valve)o

- ابتدا برای محصولات خود شرکت و سپس انتشار عمومی
CryEngine

سوال ما: موتور خودمون یا موتور آماده؟

ساخت موتور بازی سازی



مواد لازم برای ساخت موتور بازی سازی:

- تسلط به علوم پایه (ریاضی، جبر، ماتریس ها و بردارها، مثلثات، فیزیک، مکانیک و حرکت، هندسه و...)
- یادگیری کامل برنامه نویسی ++c - javascript
- ساخت موتور ریاضی
- یادگیری گرافیک کامپیوتر و رندرینگ و OpenGL و DirectX
- ساخت موتور رندرینگ
- پیاده سازی موتور شبیه سازی فیزیک
- تشخیص برخورد
- موتور صدا و DirectSound-OpenAL
- سیستم مدیریت حافظه و منابع
- رابط کاربری
- شبکه
- محاسبات زمان
- انیمیشن
- api های پلتفرمهای مختلف
- تست و دیباگ و توسعه ...

موتور بازی سازی چیست؟
وسیله ای برای جلوگیری از اختراع دوباره چرخ!

کامپوننت های اصلی یک موتور بازی سازی:

- رندرینگ
- منطق بازی
- سیستم ورودی
- سیستم صدا و موسیقی
- مدیریت حافظه و منابع
- شبیه سازی فیزیک
- محیط طراحی فضا و مرحله
- خروجی سازگار با پلتفرم های مختلف

امکانات پیشرفته:

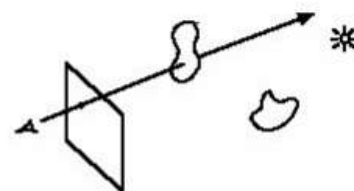
نور پردازی - شیدر نویسی - پردازش تصویر - مدلینگ و انیمیشن - هوش مصنوعی - پارتیکل سیستم
شبکه - دوربین - دیباگ

رندرینگ Rendering

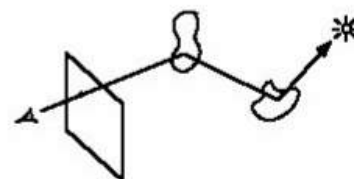
ساخت یک تصویر از مجموعه ای از اجسام ۲بعدی و ۳بعدی

Casting versus tracing

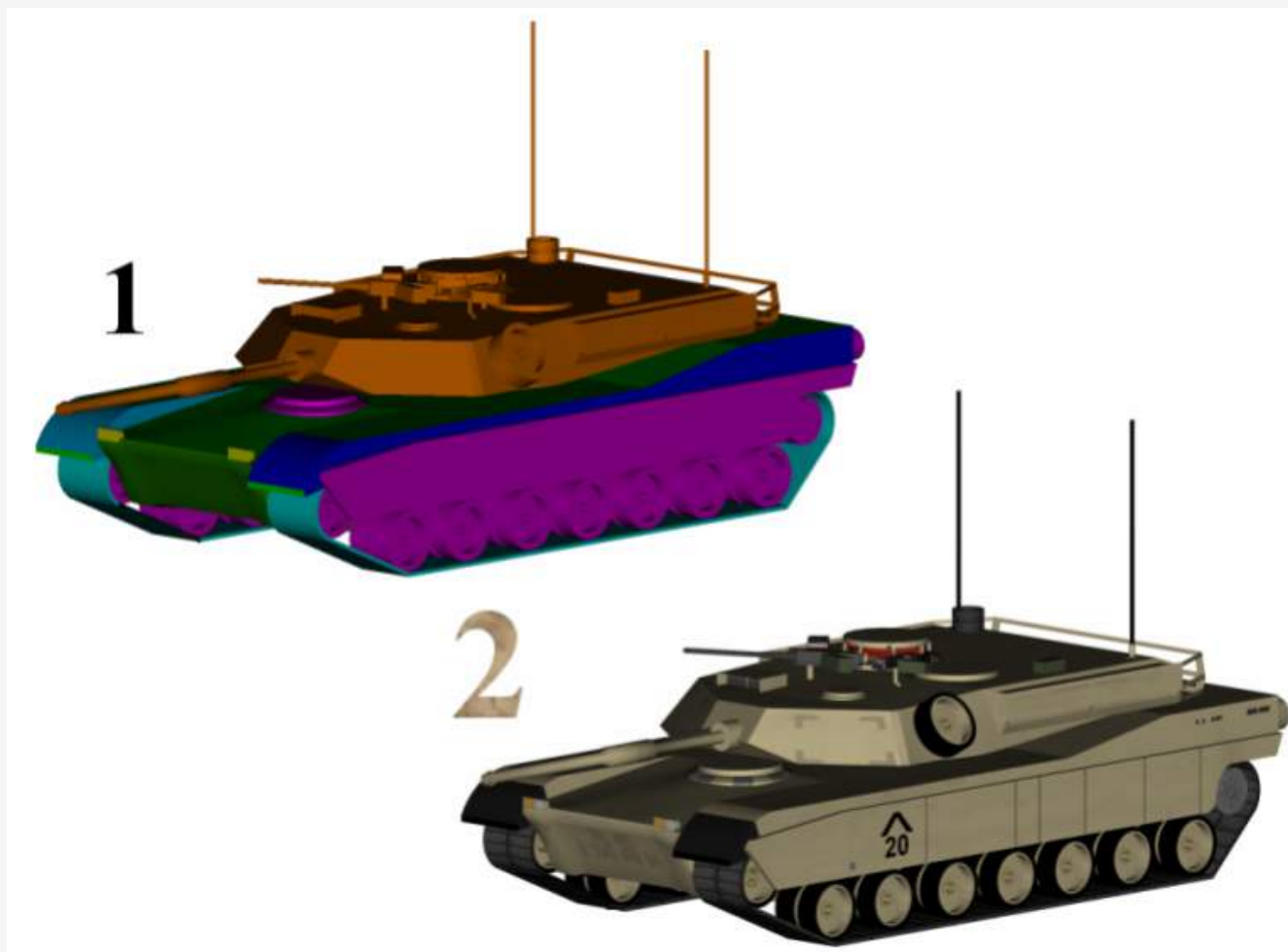
1st path = ray casting



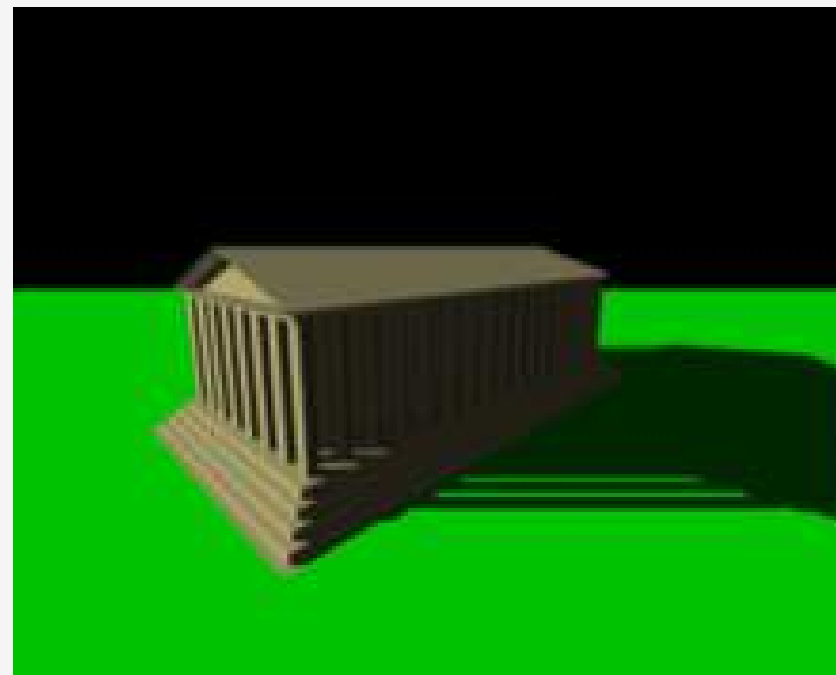
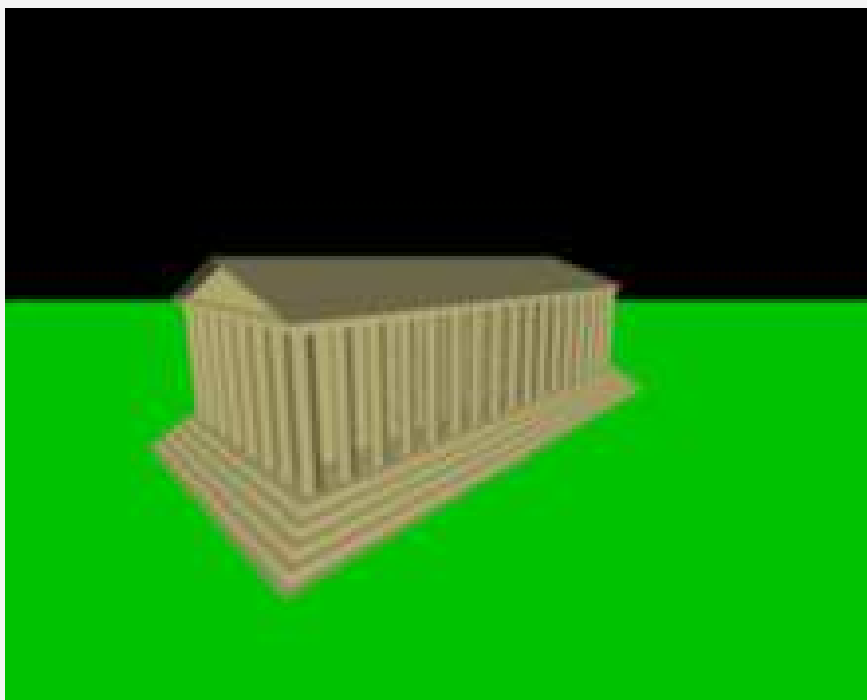
recursive = ray tracing



تکسچر Texture



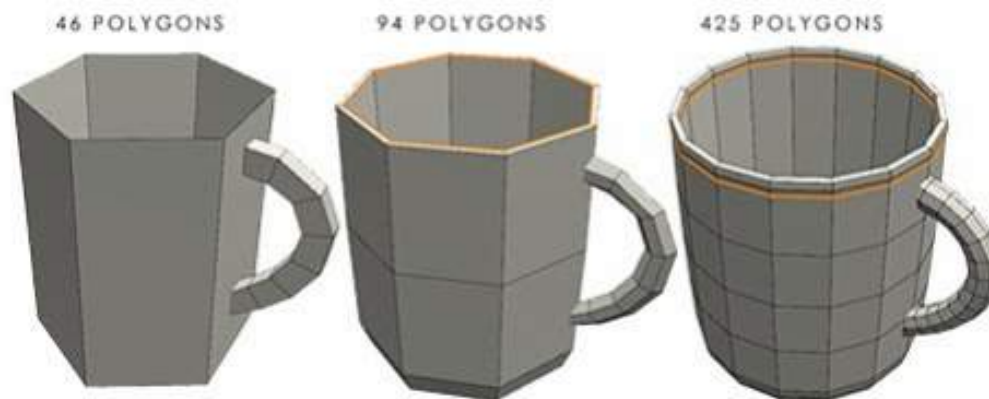
نورپردازی و سایه Lighting & Shadowing



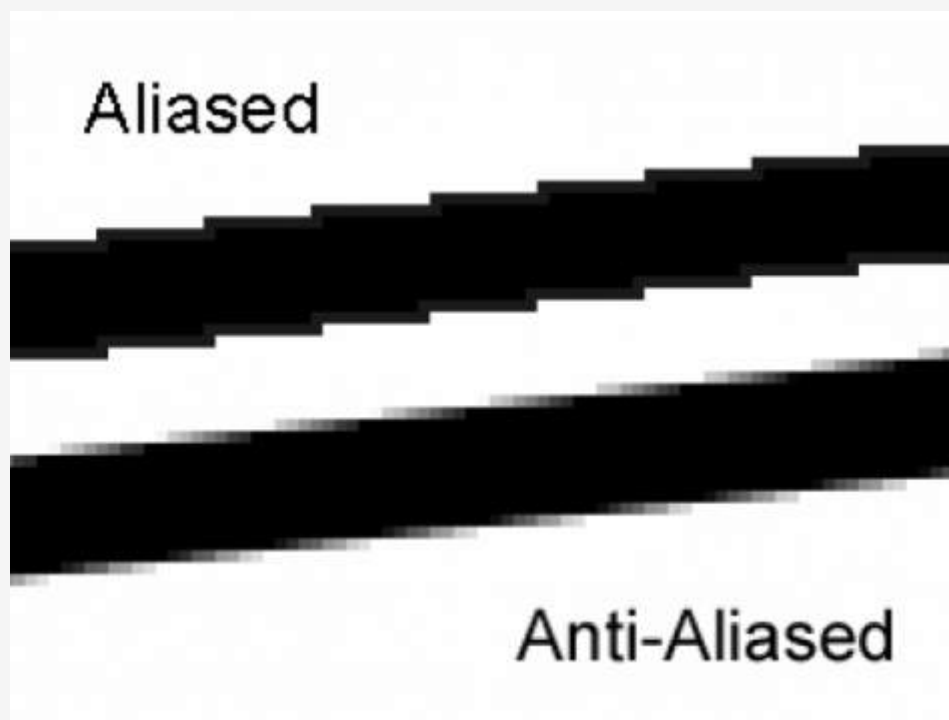
Level of Details - LOD

کاهش جزئیات با فاصله گرفتن، برای کاهش بار رندرینگ

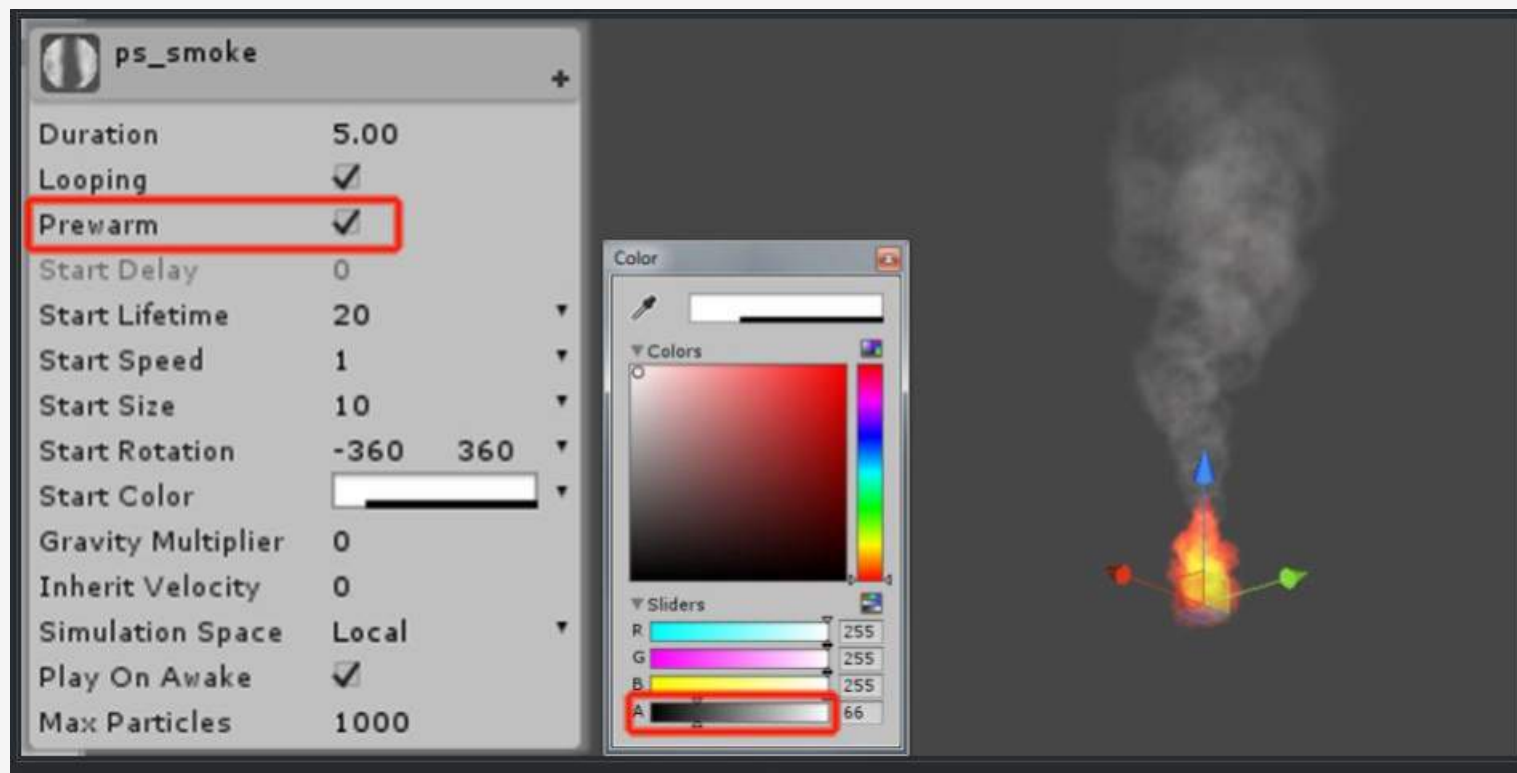
Setting up LODS in Unity



Anti-Aliasing



Particle System - VFX



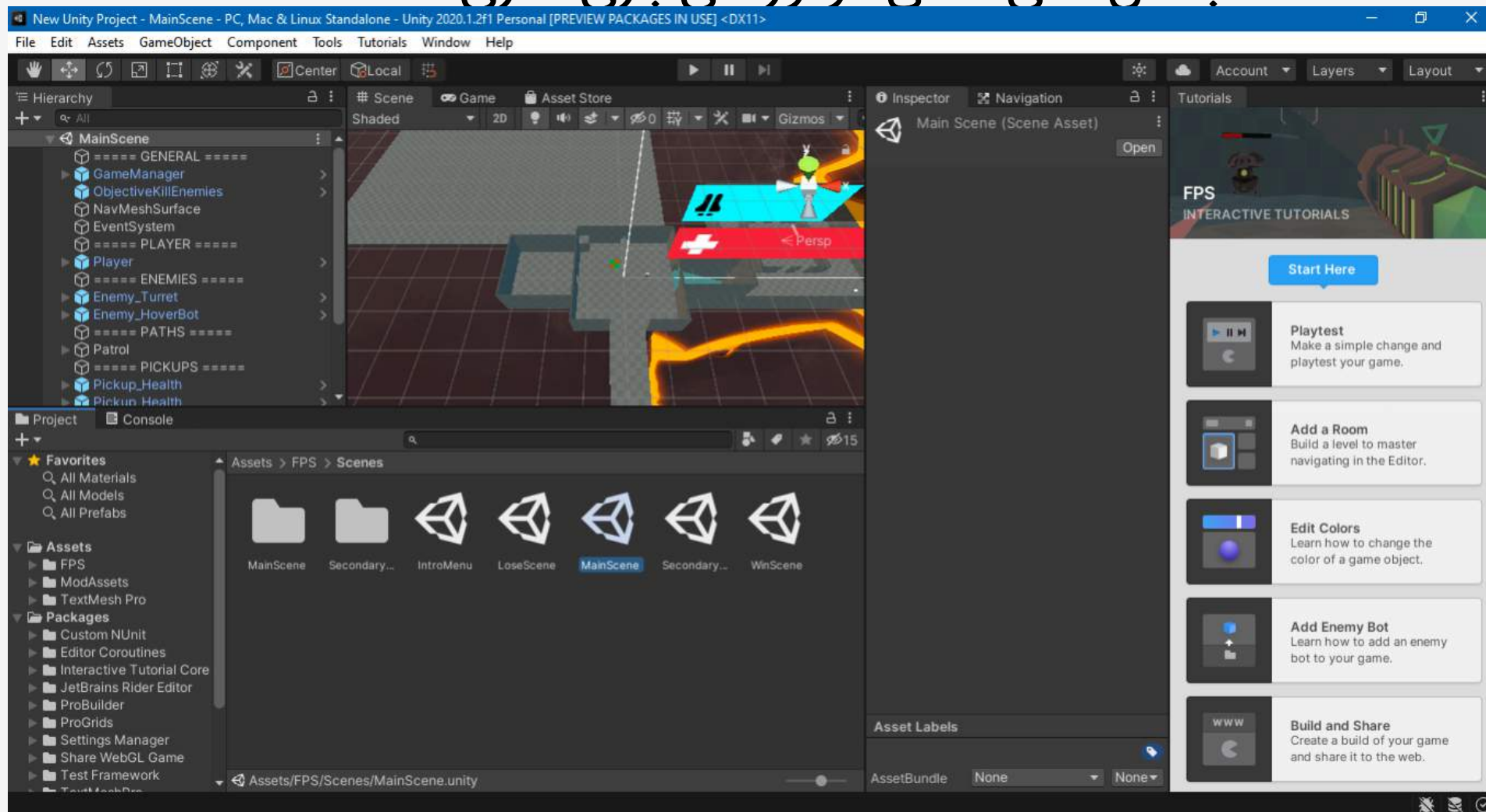
Fogging



Path Finding

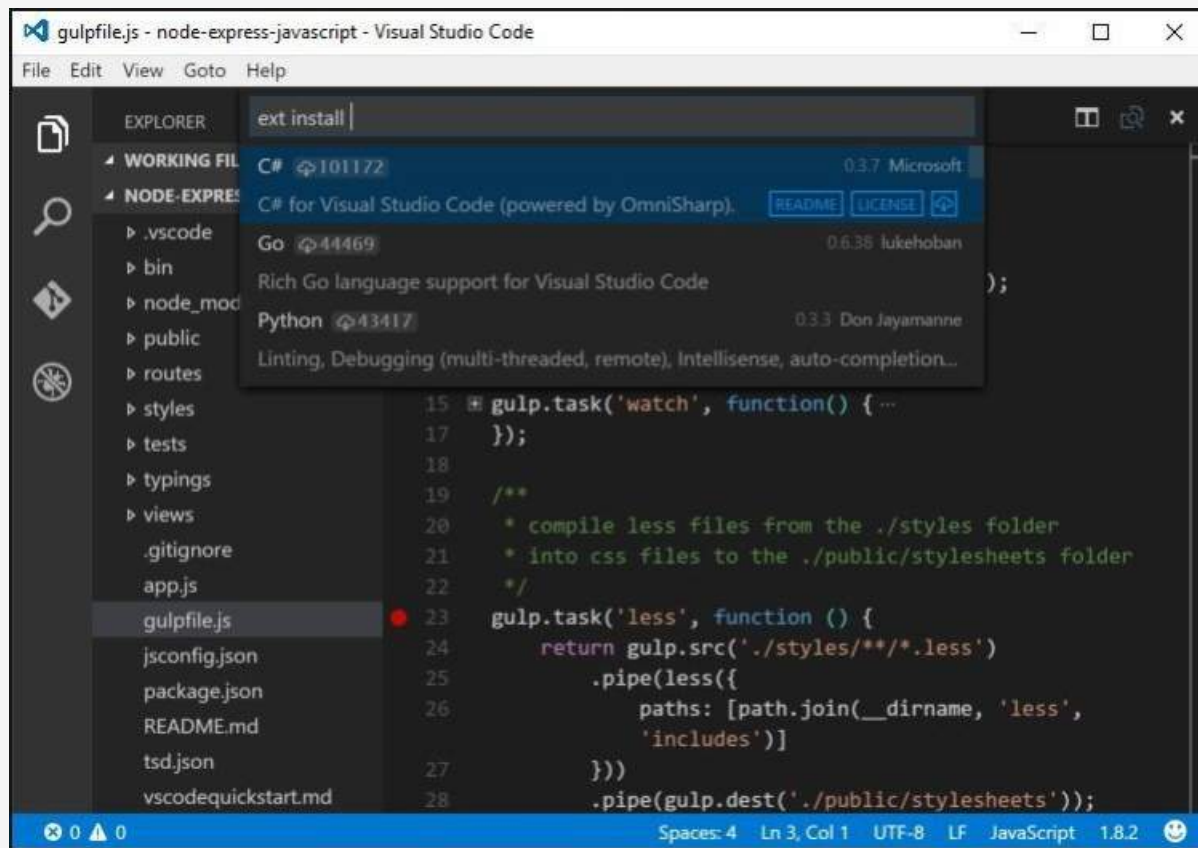


بخش های اصلی موتورهای بازی سازی - Editor

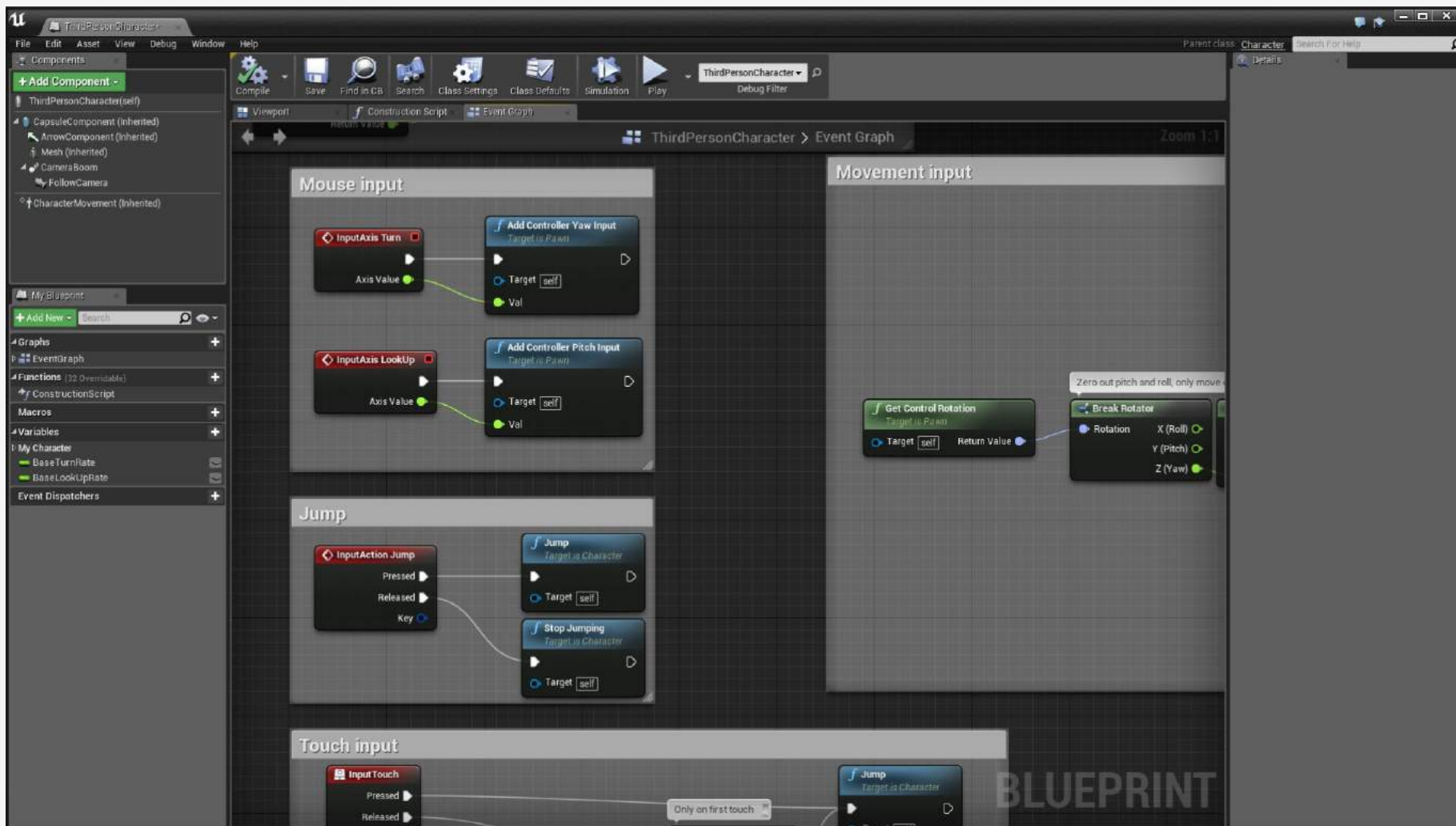


بخش های اصلی موتورهای بازی سازی - code > Scripting IDE

Visual Studio
Visual Studio Code
Jetbrains Rider
Monodevelop




بخش های اصلی موتورهای بازی سازی - visual > Scripting IDE



eCredits
eEffects
ePlayer
eMusic
eMenus
eMain
eLevelObjects

▼ Collectibles

A little trick to make a "wave" motion on collectibles, using the sine behavior

→ ⚙ System	On start of layout	🔥 Collectibles	Set  Sine cycle position to (Self.X % 128) ÷ 128
		Add action	Add...

Grab a collectible

→ 🟡 Player_Base	On collision with 🔥 Collectibles	Add action Add...	
-----------------	---	-------------------	--

Beetle (+1)

🔥 Collectibles	Is animation "Beetle" playing	📖 Dictionary	Add key "Beetles" with value <i>clamp(Self.Get("Beetles") + 1, 0, 99)</i>
		🔊 Audio	Play Collectible_1 not looping at volume 0 dB (tag "")
		Add action	Add...

Gold Beetle (+10)

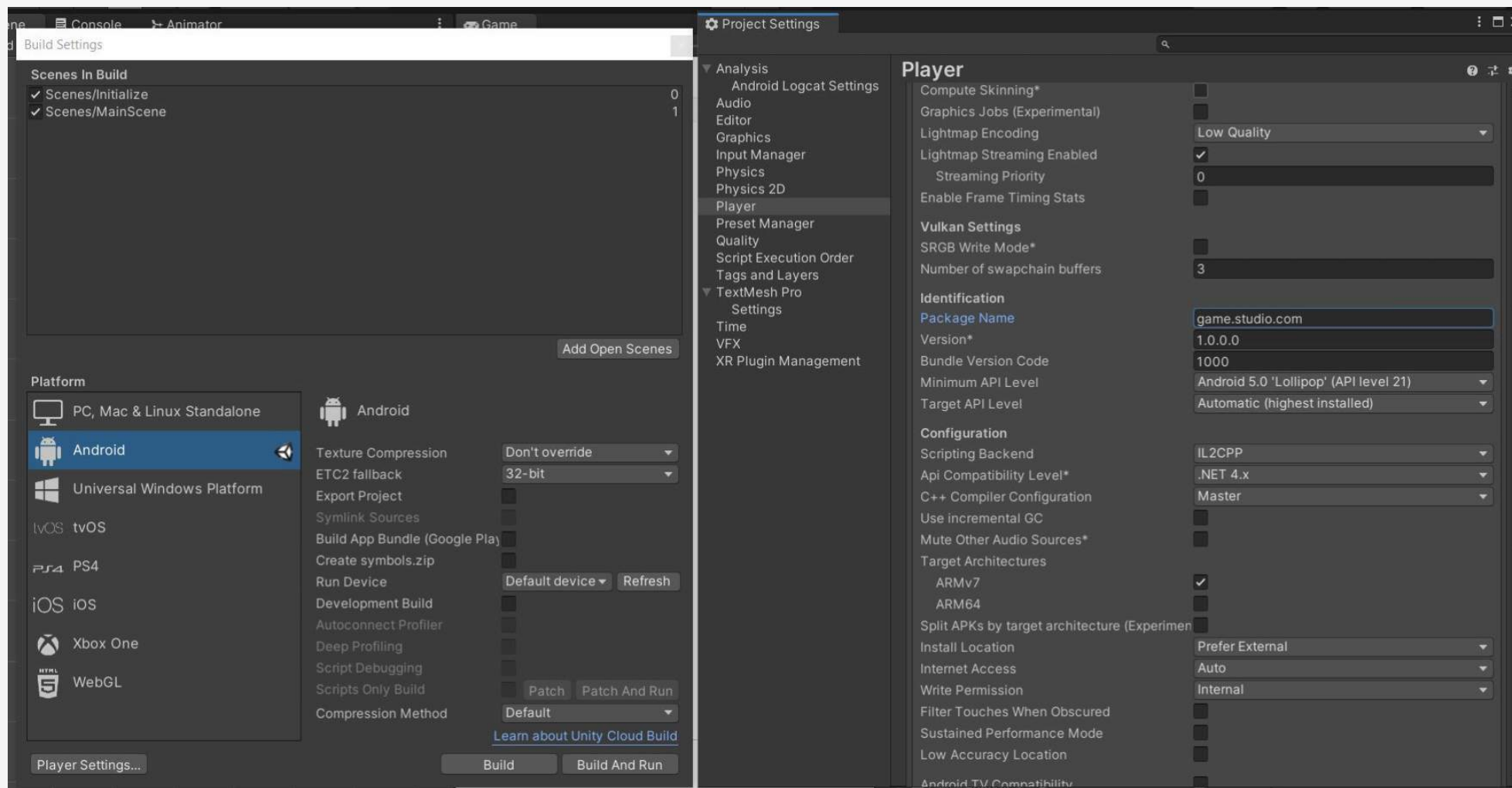
🔥 Collectibles	Is animation "GoldBeetle" playing	📖 Dictionary	Add key "Beetles" with value <i>clamp(Self.Get("Beetles") + 10, 0, 99)</i>
		🔊 Audio	Play Collectible_2 not looping at volume 0 dB (tag "")
		Add action	Add...

Health (+1)

🔥 Collectibles	Is animation "Kiwi" playing	📖 Dictionary	Add key "Health" with value <i>clamp(Self.Get("Health") + 1, 0, 2)</i>
		🔊 Audio	Play Collectible_1 not looping at volume 0 dB (tag "")
		Add action	Add...

Sparkle effects

بخش های اصلی موتورهای بازی سازی - Build System



معرفی و مقایسه چند موتور بازی سازی معروف و پرکاربرد

پارامتر های تاثیر گذار:

امکانات
پلتفرم های سازگار
محتوای آموزشی، راهنما و مستندات
جامعه کاربران بین المللی و محلی
انعطاف پذیری برای توسعه
قیمت گذاری
پروژه های انجام شده

معرفی و مقایسه چند موتور بازی سازی : Unreal Engine



سازنده : Epic Games
۲D & ۳D

قدرت بالای رندرینگ

حمایت در فروشگاه Epic Games

رایگان (۵درصد از درآمد بالای ۳هزار دلار)
خروجی: Win-Mac-Mobile-Console

بازی های ساخته شده:

Gears of war - Batman Arkham

Mass Effect

برنامه نویسی: visual و c++

جامعه کاربری و مستندات خیلی خوب
Asset Store +

معرفی و مقایسه چند موتور بازی سازی : Unity



سازنده : Unity Technologies
۲D & ۳D

سازگاری بالا با پلتفرم موبایل
جامعه کاربران زیاد در ایران
مناسب برای تازه کاران

رایگان (نسخه pro امکانات اضافه دارد)
خروجی: Win-Mac-Mobile-Console

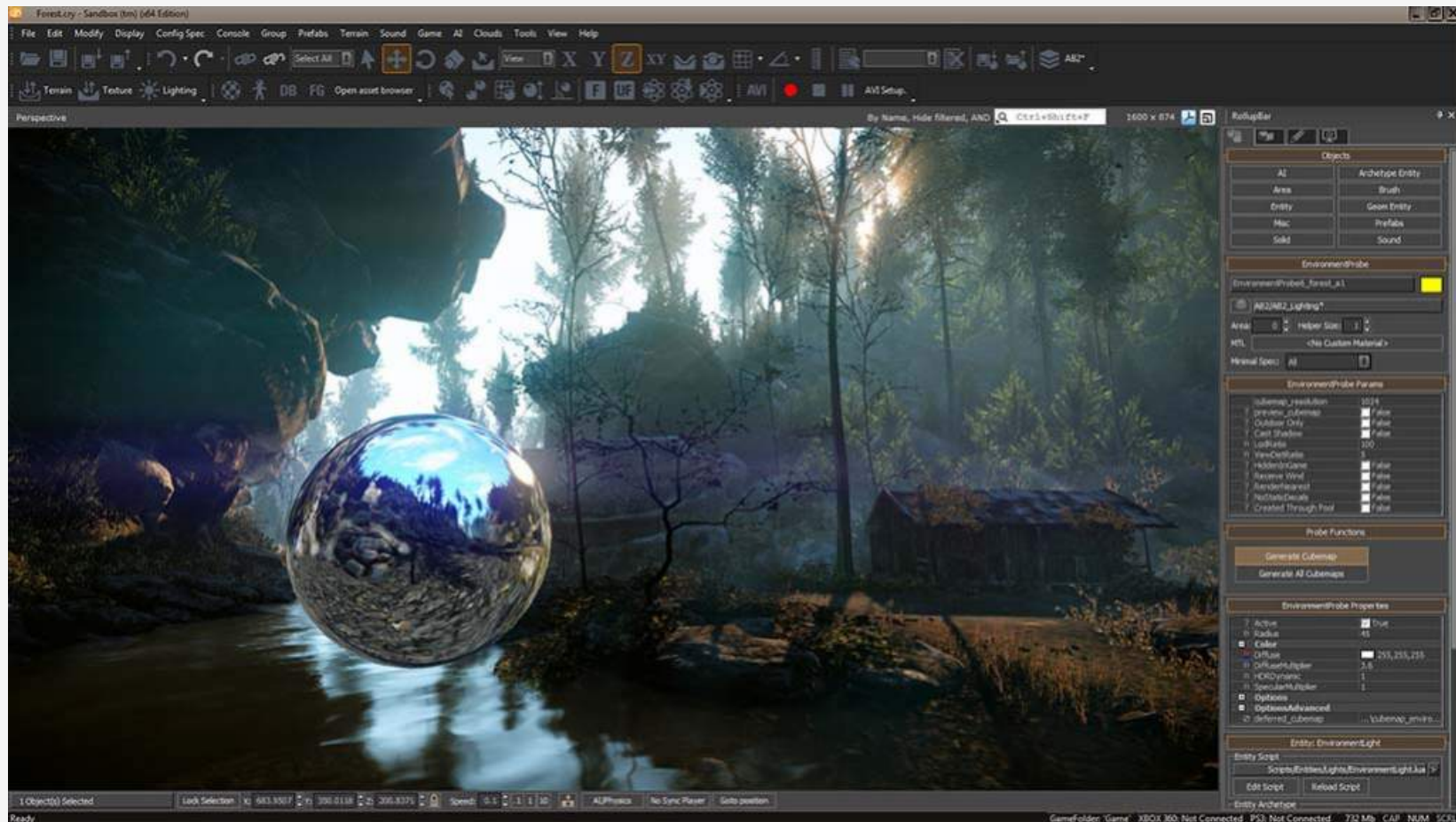
بازی های ساخته شده:
Ori - Super Mario Run - Inside
۲ Angry Birds

برنامه نویسی: C#

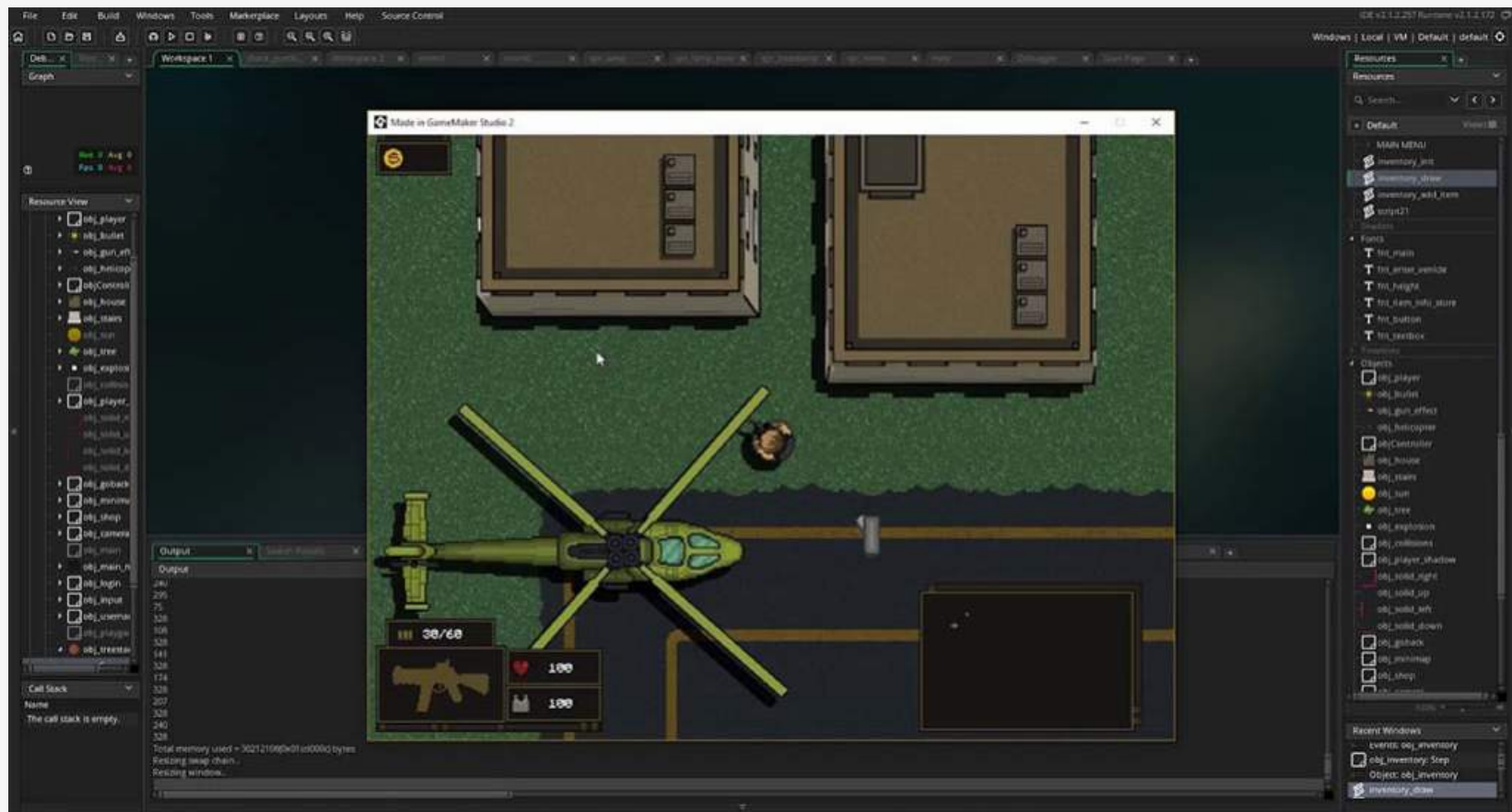
جامعه کاربری و مستندات خیلی خوب
Asset Store +

معرفی و مقایسه چند موتور بازی سازی : CryEngine

سازنده : Crytek
۳D
کاملاً OpenSource
نامناسب برای تازه کاران
رایگان (عضویت ماهانه)
خروجی: Win-Linux-Mobile-Console
بازی های ساخته شده:
Far Cry, Crysis
برنامه نویسی: C++
جامعه کاربری و مستندات خیلی خوب
Asset Store +



معرفی و مقایسه چند موتور بازی سازی : Game Maker



سازنده : Yoyo Games
۲d

برنامه نویسی ساده

رایگان برای یادگیری

خروجی: Win-Linux-Mobile-Console

بازی های ساخته شده:
Spelunky، Hotline Miami

برنامه نویسی: visual و GML

جامعه کاربری و مستندات خوب

معرفی و مقایسه چند موتور بازی سازی : Cocos Creator

سازنده : YAJI Software

۲D & ۳D

سبک و مناسب دستگاه های ضعیف

رایگان مادام العمر و متن باز

خروجی: Win-Mac-Mobile-Console
web-html

بازی های ساخته شده:
BADLAND, Castle Clash

برنامه نویسی: javascript - typescript

جامعه کاربری و مستندات خیلی خوب



COCOS CREATOR

معرفی و مقایسه چند موتور بازی سازی : Construct

سازنده : Scirra

۲D

ساده - بر پایه html-۵
بسیار مناسب تازه کارها

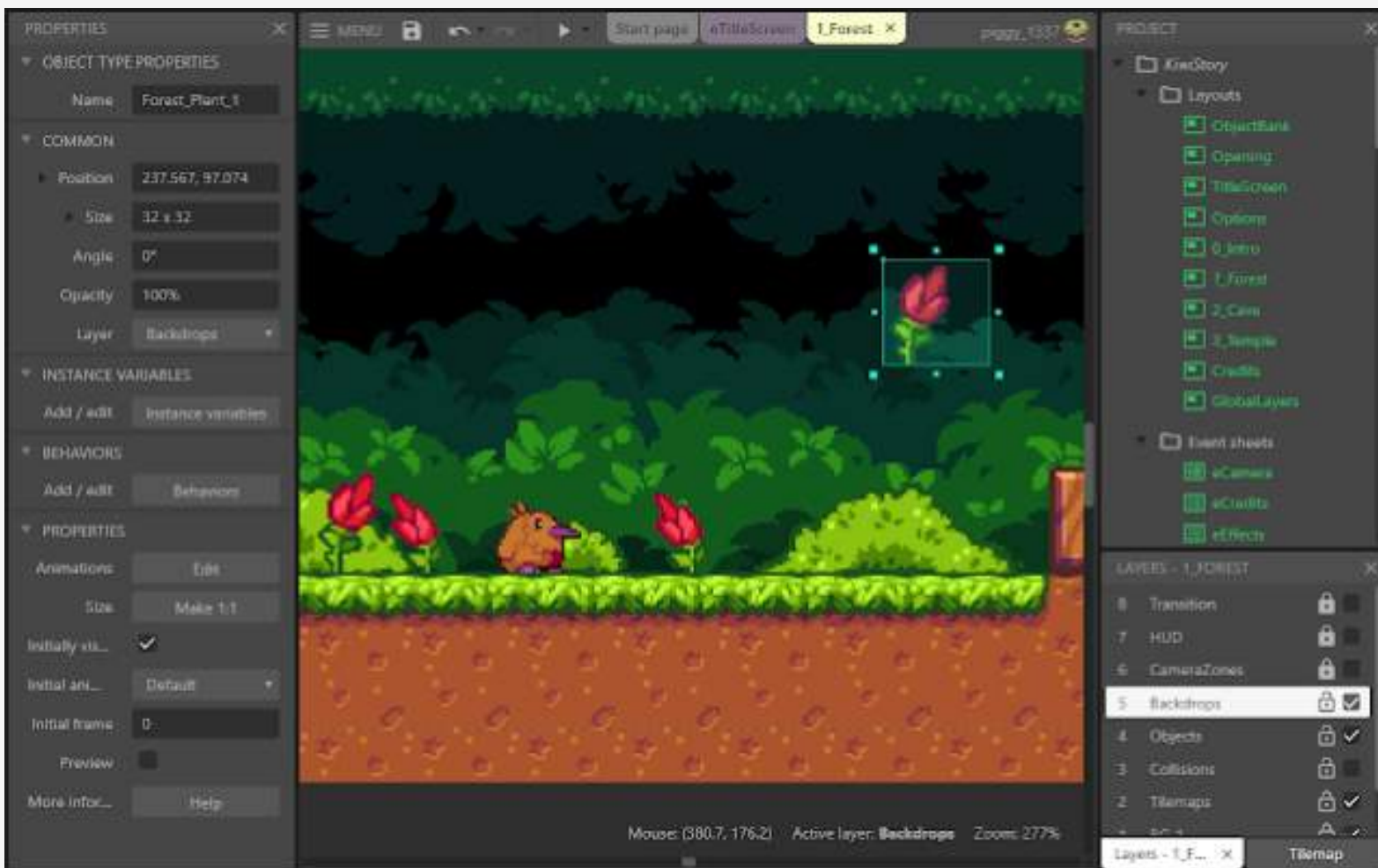
رایگان برای یادگیری (نیاز به خرید لایسنس)

خروجی: Win-Mac-Mobile-Console
facebook instant games-web-html

بازی های ساخته شده:
Guinea Pig Parkour

برنامه نویسی: event&action : visual

جامعه کاربری و مستندات خوب
AssetStore +



نتیجه گیری

اختراع نکردن چرخ!
بازی ساختن
استفاده از تجربیات دیگران و نمونه کد های آماده
جستجو در منابع موجود
docs.unity
forum.unity
answers.unity
github
stackoverflow
asset store
....

پرسش و پاسخ با تشکر

amirhosein.avagames@gmail.com

Telegram: @amirhm64

Twitter: @amirhoseinm85