



آموزش ساخت بازی اول شخص در Unity
قسمت سوم (سطح پیشرفته)

مترجم و مجری: علی زنجیران (AliyerEdon)

آبان-آذر 1388

مقدمه

مقدمه بی مقدمه!! فقط یادتون باشه از سطح این آموزش نترسید. بعد یه ماه دوباره بخونید فکر
مکنید زیر مقدماتیه!! به هر حال. موفق باشید!!

علی زنجیران (AliyerEdon) – www.Persian-Designers.com

یک بازی پیشرفته ی اول شخص در Unity



این سطح از آموزش (پیشرفته)، با گسترش دادن عناصر قسمت های قبلی آموزش مانند: Waypoint ها، هوش مصنوعی دشمن، ragdoll ها و انیمیشن، ساخت بازی اول شخص را تکمیل می کند.

زمان برای تکمیل : 2-3 ساعت

نویسنده : گراهام مک آلیستر

محتوا:

1. هدف های آموزش
2. پیش نیاز ها
3. آماده کردن محیط
4. Waypoint ها
5. هوش مصنوعی روبات
6. صدمه دیدن روبات
7. Ragdoll ها
8. صدا
9. GUI بازی

1. هدف های آموزش

این قسمت از آموزش تکمیل کننده ی بازی اول شخص شماس است. بعد از تمام کردن این قسمت از آموزش شما می توانید بازی اول شخصی بسازید که دارای ویژگی هایی چون: فیزیک، هوش مصنوعی دشمن، صدمه دیدن از نوع Ragdoll و انیمیشن شخصیت، باشد.

2. پیش نیازها

شما باید با مفاهیم دو قسمت قبل آموزش آشنا باشید. این قسمت بر پایه ی قسمت های قبل ساخته شده است.

در این آموزش مثل قسمت های قبل کدهای اسکریپت را توضیح نخواهیم داد و شما باید خودتان در مورد جزئیات آنها تحقیق کنید.

توجه: هر متنی که با حرف "-" شروع شده باشد بیانگر نیاز انجام عملی توسط شماس است.

3. آماده کردن محیط

ما با پروژه ی قسمت قبل آموزش کار خواهیم کرد.
- پروژه ی قبلی خود را باز کنید. (قسمت متوسطه).

4. Waypoint ها

این قسمت مقدمه ایست بر Waypoint ها در بازی شما که مسیری را که روبات می تواند در آن حرکت یا گردش کند را مشخص می کند. بیایید Waypoint ها را به بازی اضافه کنیم:

- شی جدید ایجا کنید و نام آن را Waypoint بگذارید. مطمئن شوید که این شی تقریباً یک متر بالاتر از سطح زمین قرار دارد.

- اسکریپت AutoWayPoint را به این شی اضافه کنید. اگر دقت کنید خواهید دید که شی waypoint به صورت نماد W در قسمت Scene View نمایان می شود.

- دو کپی از شی Waypoint بگیرید و به صورت مثلثی در محیط قرار دهید. (فاصله ی شان مقداری زیاد باشد).

- چک کنید که آیا waypoint ها یکدیگر را می بینند یا خیر. برای این کار هریک را انتخاب کرده و در Inspector View در قسمت AutoWayPoint از منوی پایین افتادنی (Context Menu) گزینه ی Update Waypoint را انتخاب کنید. حال خواهید دید که اگر بین دو waypoint شی ای نباشد یا به اصطلاح همدیگر را نبینند، خط سبز رنگ رسم می شود و در غیر اینصورت یک خط قرمز رنگ رسم می شود.

حال ما مسیری را که دشمن می تواند در آن گردش کند را مشخص می کنیم. حال بیاید دشمن را به صحنه اضافه کنیم.

5. هوش مصنوعی روبات (دشمن)

این قسمت یک روبات دشمن به صحنه اضافه خواه کرد.

Robot Artwork/robot را از Project View انتخاب کرده و به داخل Scene View درگ کنید. مطمئن شوید که کل بدنه ی روبات بالای زمین باشد.

بیاید مقداری رفتار به روبات بدهیم:

- اسکریپت WeopensScript/AI را به روبات اضافه کنید. در Inspector View در قسمت AI، متغیر Target را پیدا کرده و FPS Controller را بر روی آن درگ کنید (هدف روبات).

- اسکریپت AIAnimation ره به روبات اضافه کنید. این اسکریپت انیمیشن روبات را کنترل می کند. (زمانی که می دود، شلیک می کند و ...). این اسکریپت با اسکریپت AI ترکیب می شود تا تصمیم بگیرند چه زمانی چه انیمیشنی پخش شود. مثلا زمانی که می دود یا شلیک می کند. همچنین انیمیشن را به صورت CrossFade اجرا می کند تا نرم جلوه کند.

حال ما باید شی Character Controller روبات را مقداری بزرگتر از روبات کنیم تا در زمین فرو نرود. دقت کنید که Character Controller به صورت یک کپسول در اطراف روبات نشان داده می شود.

حال ما باید این کپسول را بزرگتر از خود روبات کنیم تا با زمین تداخل نداشته باشد.

- روبات را انتخاب کنید و در Inspector View در قسمت کامپوننت Character Controller مقادیر radius و height را افزایش دهید.

- بازی را اجرا کنید و این مقادیر را دقیق تر تنظیم کنید.

حال ما باید به روبات این امکان را بدهیم که بتواند با اسلحه ی خود شلیک کند. روبات به خوبی حرکت و انیمیت می شود. زمانی که روبات با اسلحه ی خود شلیک می کند، ما انیمیشن مربوط به شلیک و اسلحه را اجرا می کنیم و سپس راکتی از مکان لوله ی اسلحه ی روبات پرتاب می کنیم. مکان شلیک گلوله با انیمیشن روبات حرکت نمی کند و فقط به سادگی زمانی که انیمیشن شلیک اجرا شد، تقریبا از جلوی اسلحه پرتاب می شود.

- شی جدیدی ایجاد کنید و آن را فرزند شی robot قرار دهید (از Hierarchy View استفاده کنید).

- نام شی جدید را gun_spawn بگذارید.

- حال ما باید آن را در جای مناسب قرار دهیم. از قسمت Inspector View در کامپوننت transform از منوی پایین افتادنی گزینه ی Reset را انتخاب کرده و سپس مقداری شی را در جهت محور Z به جلو ببرید.

- اسکریپت RocketLauncher را به شی gun_spawn اضافه کنید. Prefab ه rocket را به داخل متغیر Projectile درگ کنید.

- یک Prefab جدید از روبات بسازید و نام آن را robot بگذارید. این باعث می شود که دشمنان بیشتری را به سادگی ایجاد کنیم.

- بازی را اجرا کنید و مطمئن شوید که روبات به سمت شما شلیک می کند.

شما نیز می توانید به روبات شلیک کنید ولی او صدمه نمی بیند. برای صدمه دیدن قسمت بعد را دنبال کنید.

6. صدمه دیدن روبات

- اسکریپت CharacterDamage را به روبات اضافه کنید.

- بازی را اجرا کنید. حال می توانید به روبات صدمه زده و آن را ناپدید کنید.

برای بهتر شدن انفجار اسکریپت explosion-advanced را به Prefab ه EXplosion اضافه کنید.

7. Ragdoll ها

Ragdoll ها ساختار bone های اسکلت را به صورت چیزی که در واقعیت است، شبیه سازی می کنند که باعث می شود روبات ما هنگام مردن به صورت طبیعی به زمین بیافتد یا پرتاب شود.

این قسمت به شما نشان می دهد که چگونه Ragdoll را تنظیم کنید.

- ابتدا صحنه ی جدیدی بسازید (File->New). ما می خواهیم Ragdoll خود را در این صحنه تنظیم کنیم. صحنه ی خود را ذخیره کنید.

- یک مکعب (Cube) بسازید و آنرا اگر لازم است تغییر اندازه دهید. ما از این مکعب به عنوان سکویی برای روبات استفاده می کنیم. روبات را به داخل صحنه درگ کنید طوری که در بالای سکو قرار گیرد.

- کامپوننت Animtion را از روبات حذف کنید (از قسمت Inspector View). این کار لازم است چون انیمیشن مردن روبات توسط فیزیک شبیه سازی می شود.

- حال از ویزارد Ragdoll استفاده کنید: Game Object->Create Other->Ragdoll. یک پنجره نمایان می شود. حال ما باید bone مناسب را در اسلات (Slot) مربوطه در این ویزارد تعیین کنیم. ما برای این کار bone مناسب را از Hierarchy View به داخل اسلات مناسب در این پنجره درگ می کنیم.

- ابتدا rootHandle (Heirarchy View) را بر روی root (در پنجره) درگ کنید.

پاها (Leg):

- Upleg_L بر روی Left Hip.
- Lowleg_L بر روی Left Knee.
- heel_L بر روی Left Foot.
- این کار را برای پای راست نیز تکرار کنید.

بدنه ی بالا (دست ها یا Arm):

- upArm_L بر روی Left Arm.
- elbow_L بر روس Left Elbow.

- upArm_R بر روی Right Arm.

- spin3 بر روی Middle Spin.

- سرانجام head بر روی Head.

- دکمه ی Create را بزنید و سپس بازی را اجرا کنید. خواهید دید که روبات سقوط می کند ولی اسلحه اش نه.

برای سقوط اسلحه:

- gun را در Hierarchy View انتخاب کنید.

- کامپوننت Rigidbody را به آن اضافه کنید.

- یک Box Collider به آن اضافه کنید. اندازه و مرکز آن را با اندازه و مرکز gun ست کنید.

حال بیاید به روبات خصوصیت وزن (mass) اضافه کنیم.

- پنجره را دوباره باز کنید (Game Object->Create Other->Ragdoll) و مقدار Mass را 4 قرار دهید.

حال ما روبات Rig شده را به یک Prefab آپلود می کنیم:

- گزینه ی Assets->Create->Prefab را انتخاب کنید.

- نام آن را robot-Ragdoll بگذارید.

- روبات را از Hierarchy View به داخل این Prtefab درگ کنید. این کار باعث می شود

که اسلحه نیز همراه روبات باشد.

حال ما زمانی که روبات کشته شد، روبات قبلی را از بین می بریم و روبات Ragdoll شده را

جایگزین می کنیم. با این کار روبات به صورت واقعی می میرد.

استفاده از Ragdoll ها:

صحنه ی اصلی را باز کنید (نیاز به ذخیره ی صحنه ی جاری نیست). روبات را از Hierarchy

View انتخاب کرده و در Inspector View در قسمت Character Damage ، Prefab

ه robot-Ragdoll را به داخل متغیر dead replacement درگ کنید.

- بازی را اجرا کنید. حال اگر روبات را هدف قرار دهید، بعد از کشته شدن به زمین میافتد. ممکن است شما بخواهید وزن (mass) روبات را اصلاح کنید تا نتیجه ی کار بهتر شود. این کار را می توانید از ویزارد Ragdoll همانطور که قبلا نشان داده شد انجام دهید. مخصوصا زمانی که روبات تحت انفجار است و به هوا قرار است پرتاب شود. در این هنگام وزن نقش بسیار مهمی را ایفا می کند.

8. صدا (Sound):

در این قسمت افکت های صوتی را به بازی اضافه خواهیم کرد.

اسلحه ی تیربار (MachineGun):

- فایل صوتی machineGunSingleShotLoop را بر روی شی machineGun درگ کنید.

- خاصیت PlayOnAwake را خاموش کنید.

موشک انداز (Rocket Launcher):

ما می خواهیم منبع صدا را به راکت اضافه کنیم. بنابراین صدا نیز هنگام دور شدن راکت دور می شود.

- Prefab ه rocket را انتخاب کنید. فایل صوتی RocketLauncherFire را به جایی دلخواه در Inspector View درگ کنید تا به Prefab اضافه شود.

- مقدار rolloff را 0,5 قرار دهید تا صدا زودتر قطع شود.

- همچنین می توانید با تغییر مقدار explotion power راکت را کم صدا یا پر صدا کنید. مقدار 50 خوب است.

9. GUI (رابط کاربری):

رابط کاربری بازی یا به اصطلاح GUI مسئول اطلاع دادن وضعیت فعلی بازی به بازیکن است. عناصر اصلی رابط کاربری شامل این عناصر است: تعداد تیرهای باقی مانده، سلامتی، هدف مرحله و ...

در این بازی ما سه عنصر اصلی داریم:

Overlay:

Overlay قسمت های مختلف رابط کاربری را تنظیم می کند مانند سلامتی یا تیر.

- فایل GUOverlay را در Project View انتخاب کنید.

- گزینه ی Game Object->Create Other->GUI Texture را انتخاب کنید. این کار

باعث می شود که تکسچر انتخابی در مرکز Game View قرار گیرد.

حال ما نیاز داریم تا Overlay را در جای مناسبی از صفحه قرار دهیم.

- transform آن را 0 0 0 قرار دهید.

- مقدار Pixel Inset را 0 0 256 128 قرار دهید. (Xmin Ymin Xmax Ymax).

سلامتی (Health):

- تکسچر healthbar GUI->healthbar را از Project View انتخاب کنید.

- گزینه ی Game Object->Create Other->GUI Texture را انتخاب کنید.

- مقدار transform را 0 0 0 کنید.

- مقدار Pixel Inset را 37 83 148 112 کنید.

- تکسچر healthBar در حین بازی تغییر اندازه خواهد داد تا محیط کنار آیکون سلامتی را

پوشش دهد.

اسلحه ی تیرباز:

- گزینه ی Game Object->Create Other->Text را انتخاب کنید.

- Transform آن را 0 0 0 قرار دهید.

- مقدار Pixel Inset را 188 109 کنید.

شما می توانید متن این Text را عوض کنید ولی به هر حال در حین بازی تغییر می کند.

راکت ها:

- فایل GUIRocket را از Project View انتخاب کنید.
- گزینه ی GUI Texture->Create Other->Game Object را انتخاب کنید.
- اسکریپت FPSPlayer را به FPS Controller اضافه کنید و GUI را در قسمت اسکریپت FPSPlayer قرار دهید.

در انتها:

- ما تقریباً آموزش را تمام کردیم ولی بیایم مقدار صدای بیشتری را اضافه کنیم:
- صداها صداهای صدمه دیدن بازیکن:
- به اسکریپت FPSPlayer اضافه شده به FPS Controller صداها زیر را اضافه کنید:
 - moanOfPainSmall -
 - moanOfPainBig -
 - PlayerDie -

صداها قدم زدن:

- ما می توانیم هر تعداد صدای قدم زدن به FPS Controller اضافه کنیم (برای دوری از تکراری شدن). اینها منتخب صداها تصادفی انتخاب شده از لیست هستند:
- FPSPlayer را انتخاب کنید. مثلث کنار Walk Sound را باز کنید. سائز آن را 5 کنید.
 - فایل های صوتی footStep1-5 را به ترتیب برای 5 عنصر ایجاد شده اضافه کنید.
 - یک Audio Source به FPS Player اضافه کنید. فقط همین و کار دیگری نیاز نیست.

نورهای برای اشیای متحرک

صحنه ی ما لایت مپ (Lightmap) شده است یعنی قبلاً در نرم افزار مدلسازی نورپردازی شده است و سایه ی اشیاء محاسبه شده و بر روی شیء انداخته شده است و به اصطلاح داینامیک

نیست. پس اشیاء متحرک و شخصیت های بازی از این نور نمی توانند استفاده کنند چون حرکت می کنند. پس ما باید نوری را ایجاد کنیم که فقط به این اشیاء تاثیر بگذارند و شبیه لایت مپ باشند. برای این هدف ما از خاصیت culling mask ه لایت خود استفاده می کنیم.

- یک Directional Light ایجاد کنید.

ما از culling mask استفاده می کنیم که چیزی را که Directional Light بر آن تاثیر می گذارد را مشخص می کند.

- ابتدا به قسمت Tags->Project Setting->Edit بروید. 8 User Layer را انتخاب کرده و مقدار lightmapped را وارد کنید.

- حال Directional Light را انتخاب کرده و گزینه ی lightmapped را خاموش کنید.

- شی mainLevelMesh را انتخاب کنید. خاصیت Layer آن را به lightmapped تغییر دهید. به سوال پرسیده شده پاسخ yes را بدهید.

هرچی بود تموم شد...

آموزش اول شخص ما هم تموم شد. خیلی از راه حل ها و تکنیک های آورده شده در سبک های دیگر بازی ها نیز استفاده می شود و مختص سبک اول شخص نیست. ما امیداریم که شما با پایه های بازی سازی آشنا شده باشید و از این به بعد بازی های زیبا و فوق العاده با Unity بسازید.

سپاسگزاری:

سپاس فراوان از جوچیم آنته (کد) و اتان و سبورق (گرافیک) برای کمکی که برای تالیف این کتاب کردند و همچنین تشکر فراوان از همه ی کسانی که این مطلب را ترجمه می کنند!!!!

با یاد و ذکر خداوند متعال و مهربان و شافی قلوب و دلها



