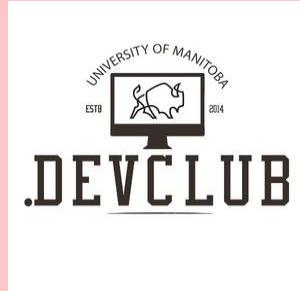


Dress to Impress Workshop Day 2 - Poses

SPONSORED BY



Day 2 - Poses

01 *Introduction*

04 *Unity Animator*

02 *Setting up Unity*

05 *Pose Behaviour*

03 *Opening Package*

06 *Posing UI*

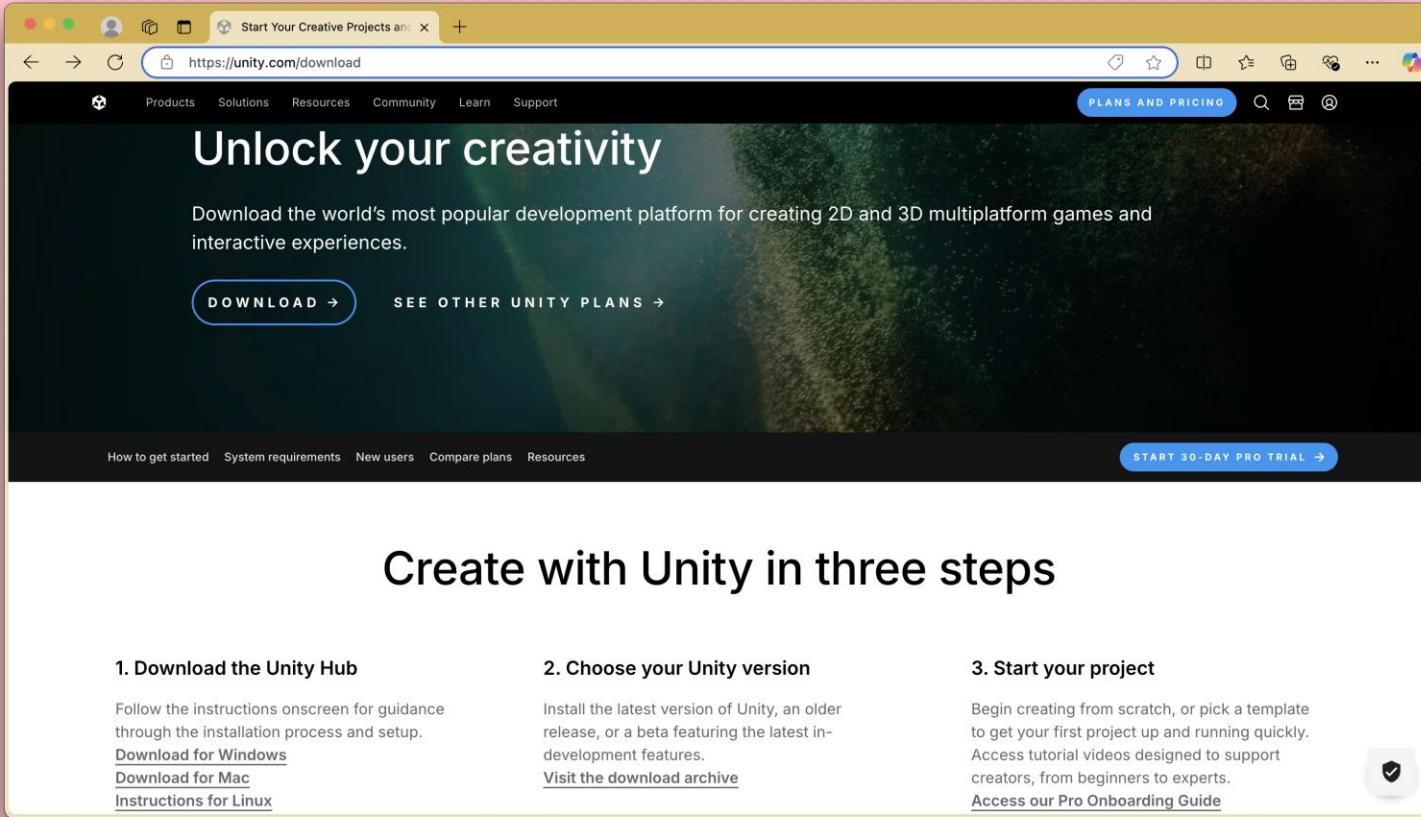
About Me

- My name is Francesca Carino
- 2nd Computer Science Major
- Made Unity game projects in high-school (2D and 3D platformers, shooters, RPG)
- Participated in Game-Up in elementary which sparked my interest in coding and game design
- Some of my favourite games are Stardew Valley, Hades, the Ori Duology and Pokemon!

Setting Up Unity

- [Downloading Unity Hub](#)
- [Installing Editor](#)
- [Creating New Blank 3D Project](#)

Go to <https://unity.com/download> to download Unity for your device



The screenshot shows the Unity website's download page. At the top, there's a navigation bar with links for Products, Solutions, Resources, Community, Learn, Support, and PLANS AND PRICING. Below the navigation is a large banner with the text "Unlock your creativity" and a subtext: "Download the world's most popular development platform for creating 2D and 3D multiplatform games and interactive experiences." It features two main calls-to-action: "DOWNLOAD →" and "SEE OTHER UNITY PLANS →". The background of the banner is a dark image of a forest. At the bottom of the banner, there are links for "How to get started", "System requirements", "New users", "Compare plans", "Resources", and "START 30-DAY PRO TRIAL →". Below the banner, the page title "Create with Unity in three steps" is displayed, followed by three numbered sections: 1. Download the Unity Hub, 2. Choose your Unity version, and 3. Start your project. Each section includes descriptive text and links to further resources.

Create with Unity in three steps

1. Download the Unity Hub

Follow the instructions onscreen for guidance through the installation process and setup.

[Download for Windows](#)
[Download for Mac](#)
[Instructions for Linux](#)

2. Choose your Unity version

Install the latest version of Unity, an older release, or a beta featuring the latest in-development features.

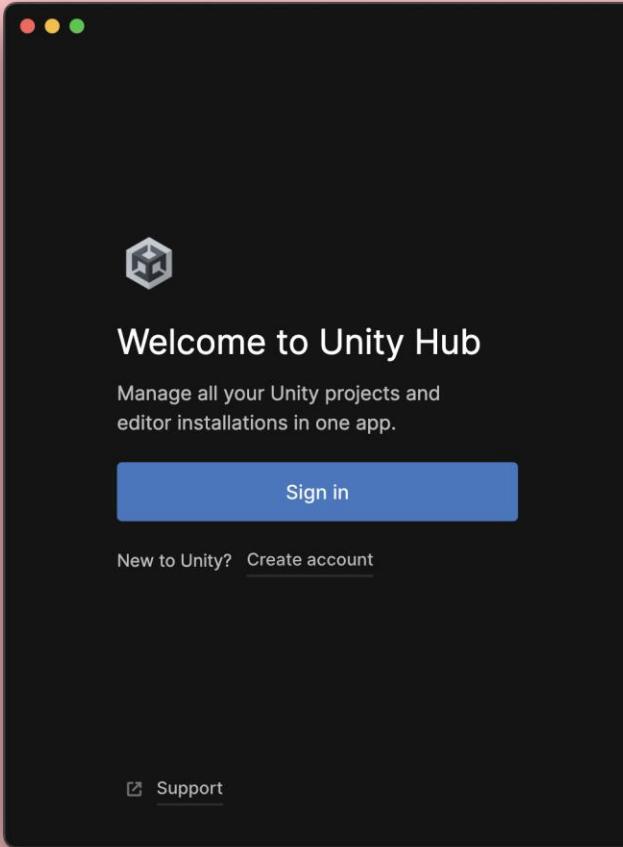
[Visit the download archive](#)

3. Start your project

Begin creating from scratch, or pick a template to get your first project up and running quickly. Access tutorial videos designed to support creators, from beginners to experts.

[Access our Pro Onboarding Guide](#)

Open Unity Hub once it is installed



The image shows the Unity Hub application window. It features a dark background with a central white area. At the top left is the Unity logo (a hexagon with a play button icon). Below it is the text "Welcome to Unity Hub". Underneath that, a message reads "Manage all your Unity projects and editor installations in one app." A large blue "Sign in" button is centered below the message. At the bottom, there is a link "New to Unity? [Create account](#)". At the very bottom, there is a "Support" link with a gear icon.



The right side of the image is a screenshot from a game titled "Time Ghost". It shows a character from behind, wearing a green jacket and a backpack, standing in a field of tall grass under a cloudy sky. In the distance, there are hills or mountains. The game's title and developer information are overlaid at the bottom right of the screenshot.

Create an account or sign-in if you have an account already

Unity ID

Create a Unity ID

If you already have a Unity ID, please [sign in](#).

Email

Password

Username

Full Name

I have read and agree to the [Unity Terms of Service](#) (required).

I acknowledge the [Unity Privacy Policy](#) [Republic of Korea Residents agree to the [Unity Collection and Use of Personal Information](#)] (required).

I agree to have [Marketing Activities](#) directed to me by and receive marketing and promotional information from Unity, including via email and social media (optional).

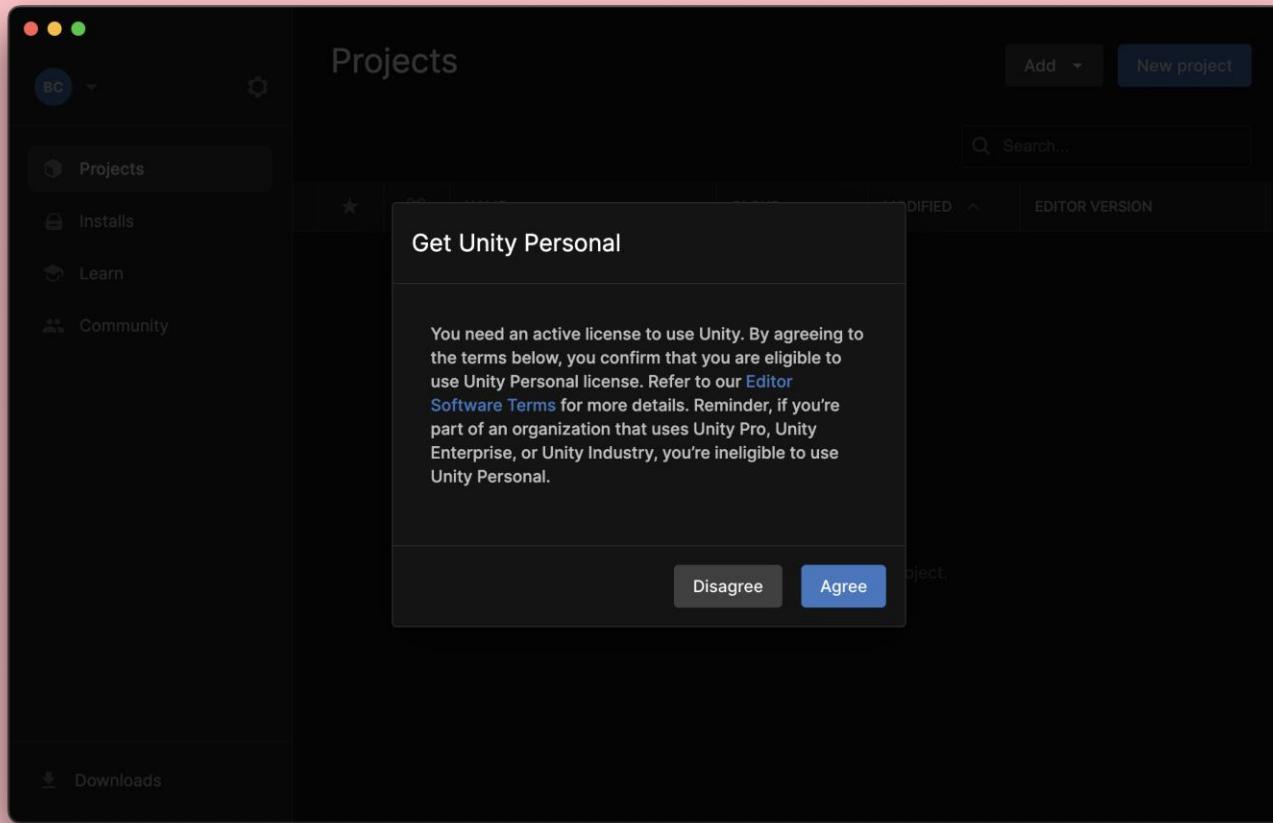
I'm not a robot 
reCAPTCHA
Privacy - Terms

[Create a Unity ID](#) [Already have a Unity ID?](#)

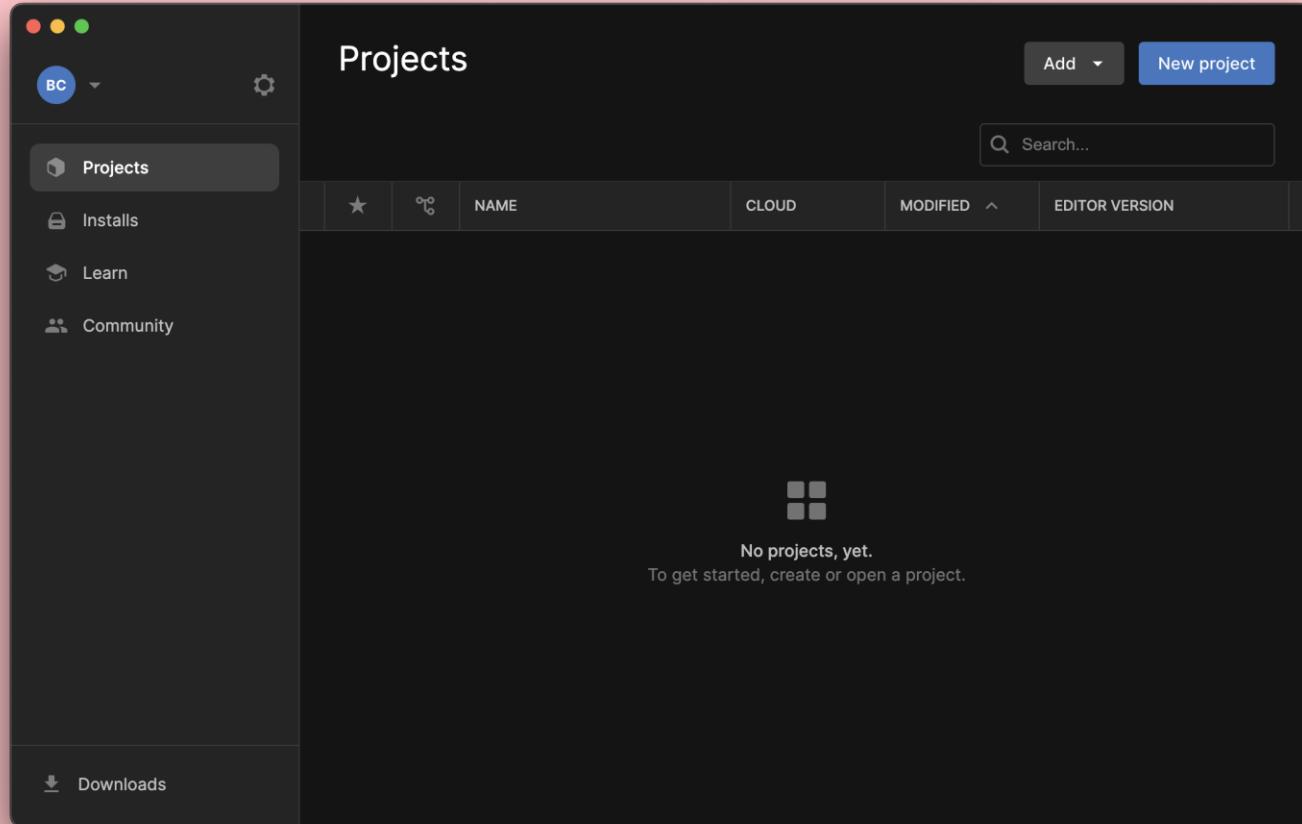
OR



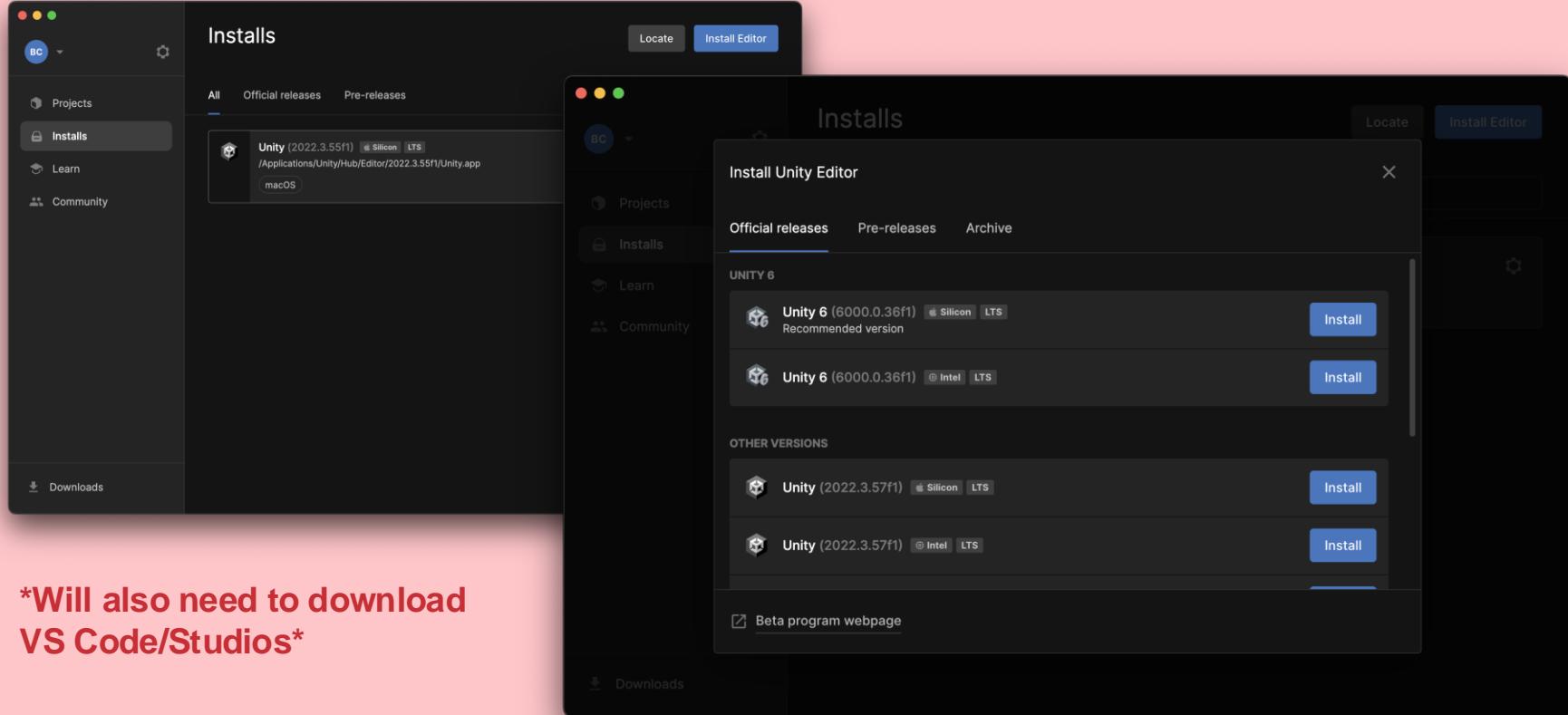
Click agree, Unity is free for personal use!



This is where projects will be, but first we have to install an editor.

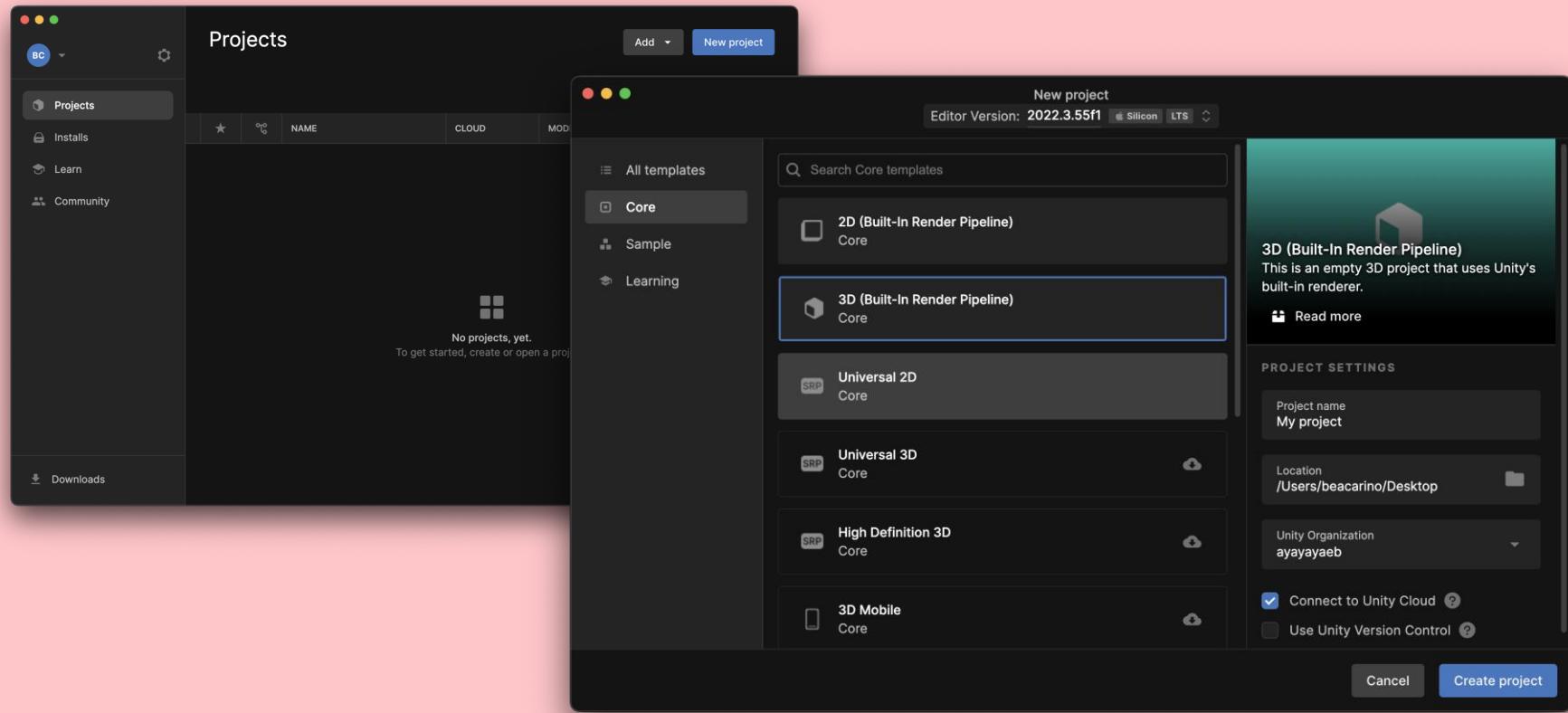


Go to "Installs" tab and click "Install Editor" and select the appropriate *Unity* version *2022.3.57f1* or *Unity 6* for your device

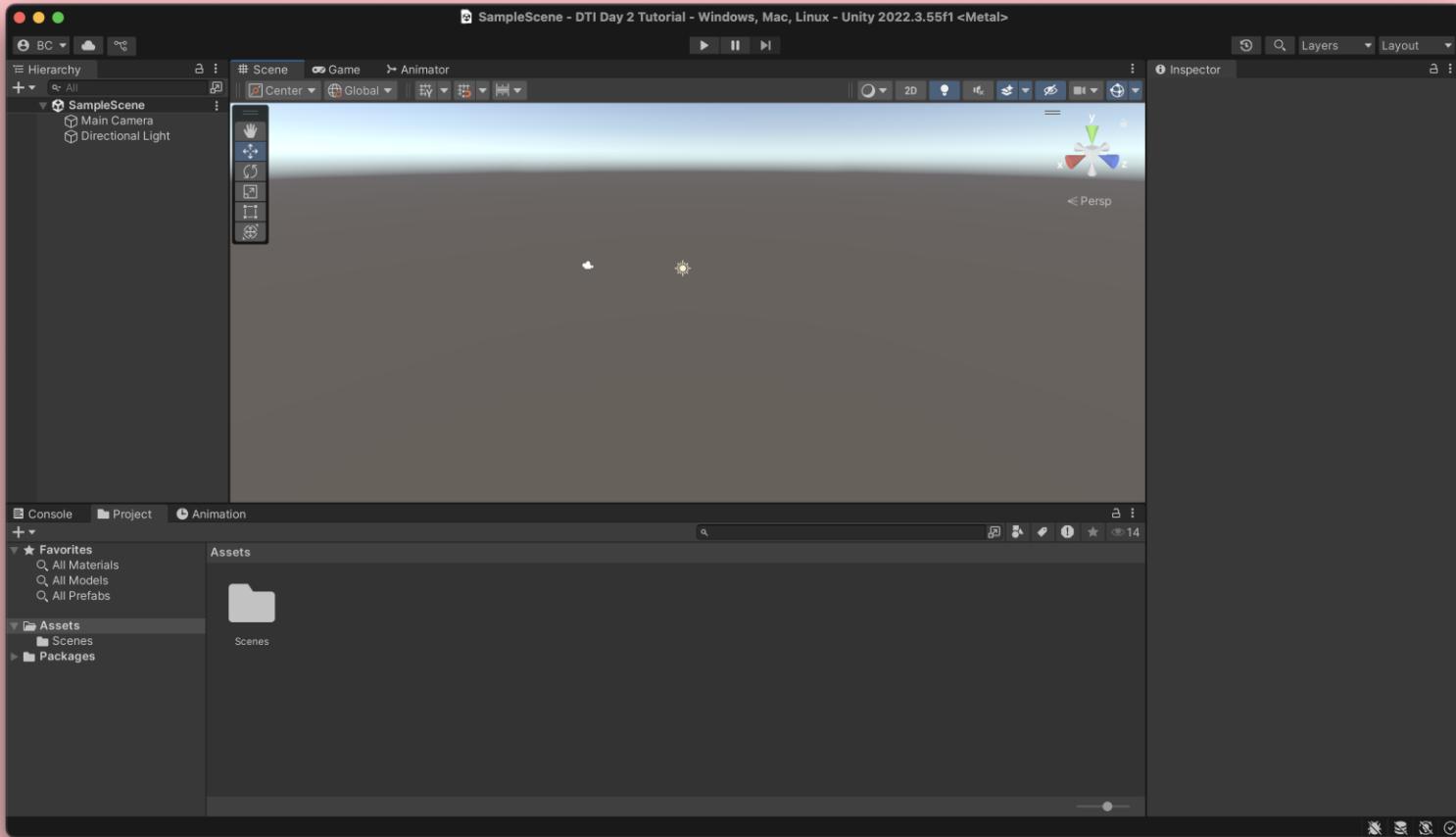


***Will also need to download
VS Code/Studios***

Go back to the "Projects" page, click "New Project", select "3D Core", name your project and hit "Create Project"!



Unity should open your project through the editor and look like this

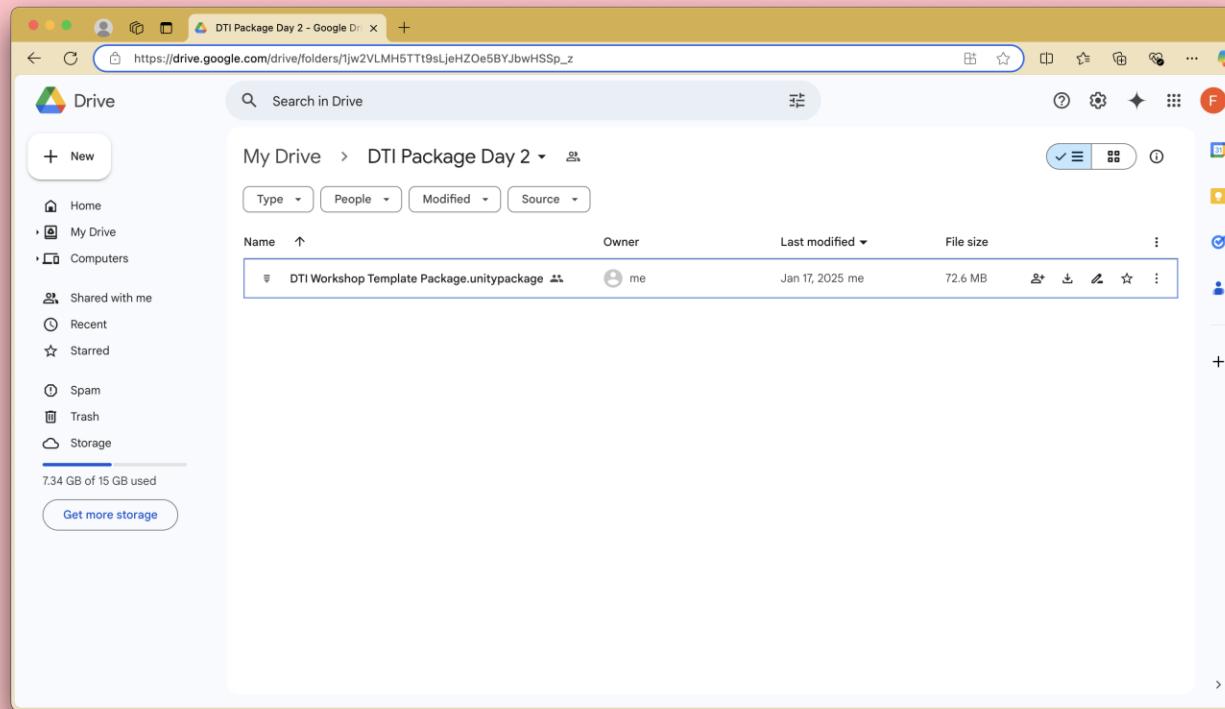


Opening Package

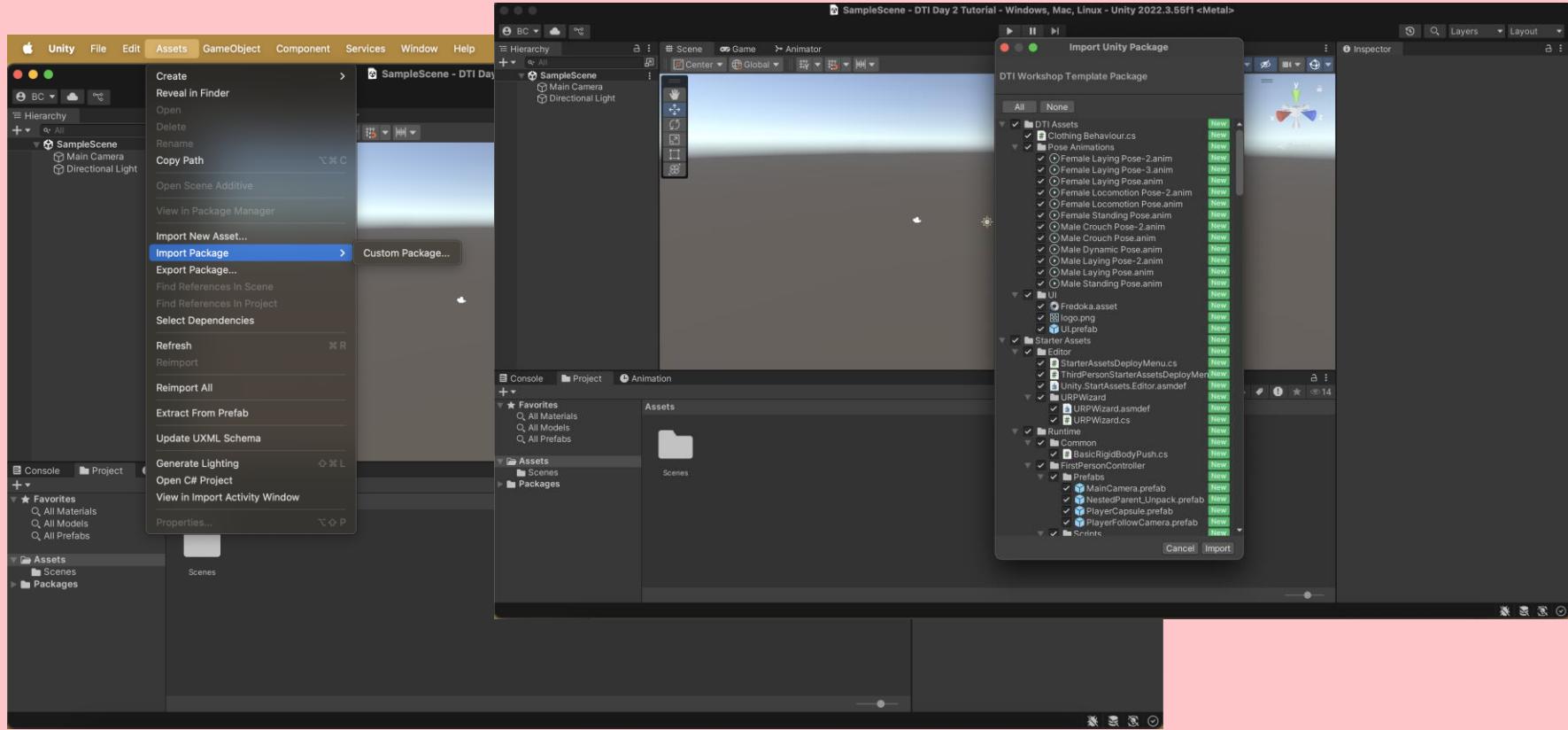
- **Downloading DTI Template package**
- **Importing packages in project**

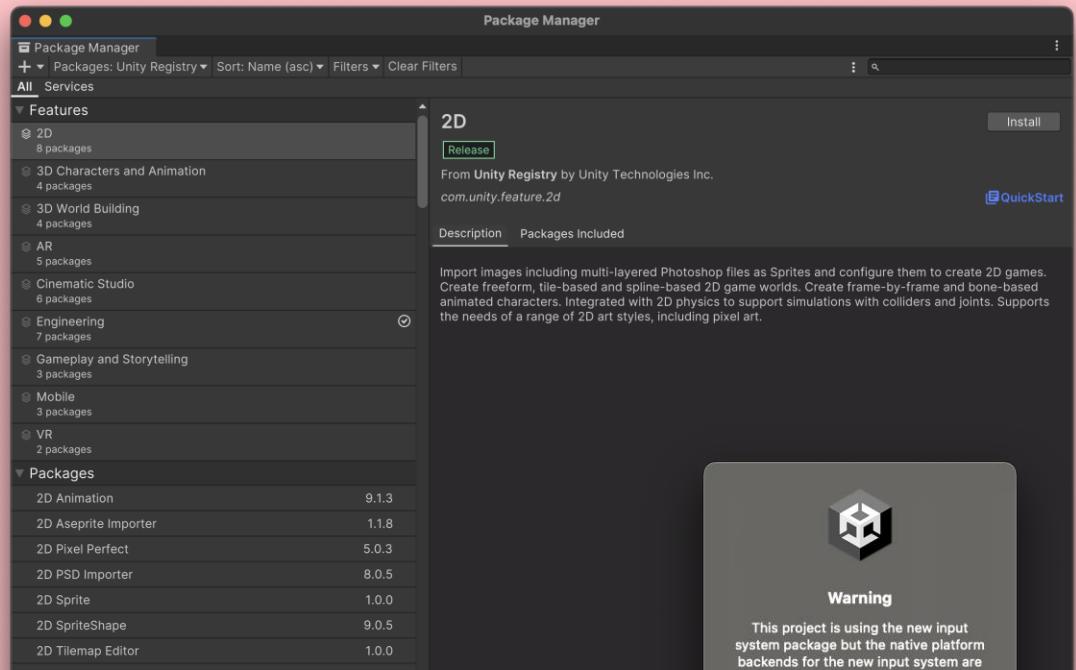
Go to the following link and click "day2" to access the Google Drive folder and download the file inside.

<https://www-test.cs.umanitoba.ca/~carinof/>



Once downloaded open your Unity project again and import the downloaded package

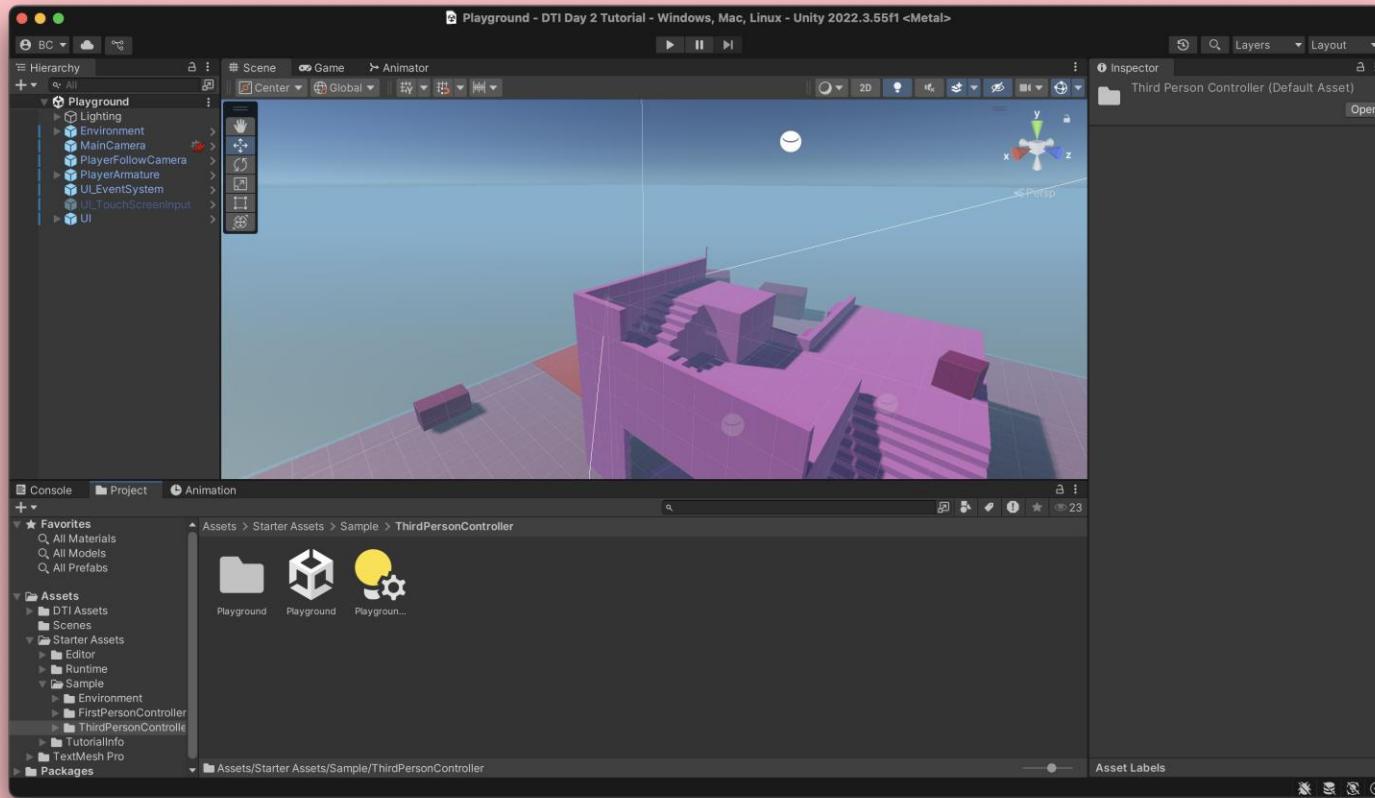




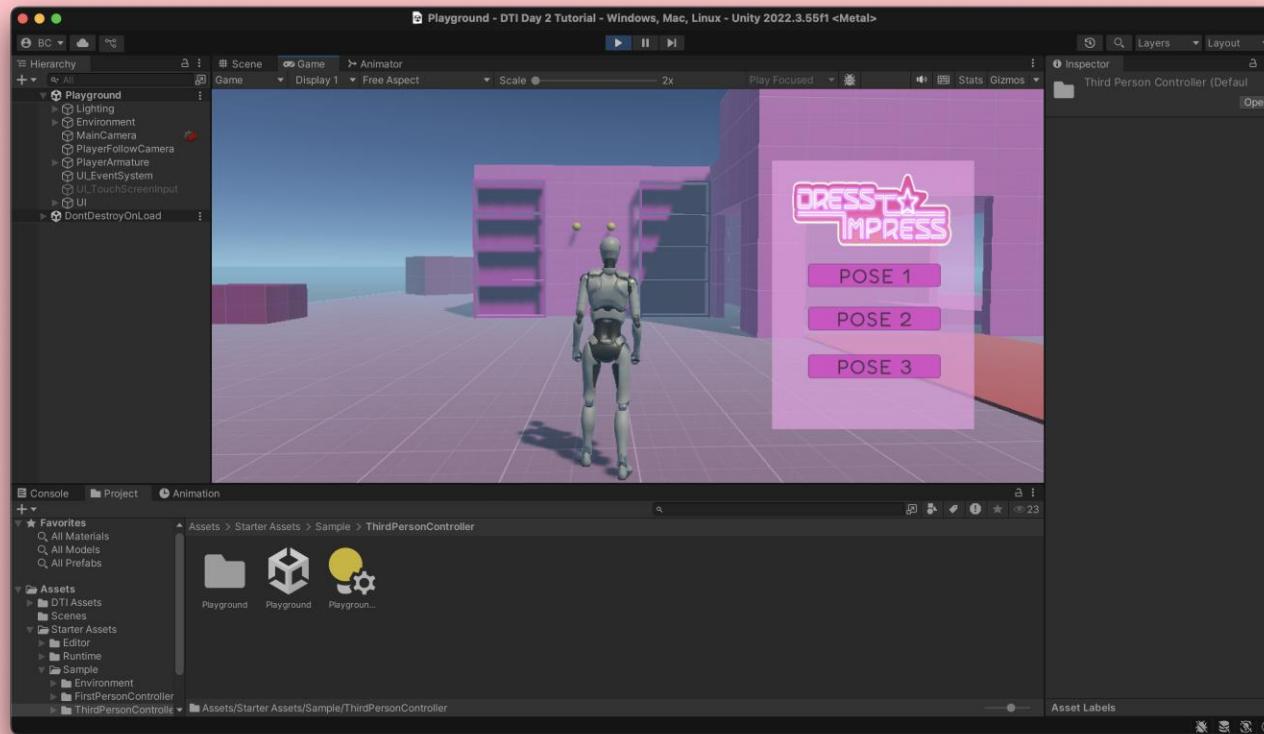
There may be errors regarding missing packages. To resolve open "Package Manager" and filter to "Unity Registry" then search for "Input System" and "Cinemachine" and install both.

Click yes after installing these!

Package is completed imported! Go to Assets > Starter Assets > Sample > ThirdPersonController and open up the "Playground" scene



Click the play button on the top of screen and play around the environment, (none of the pose button should work – for now!)

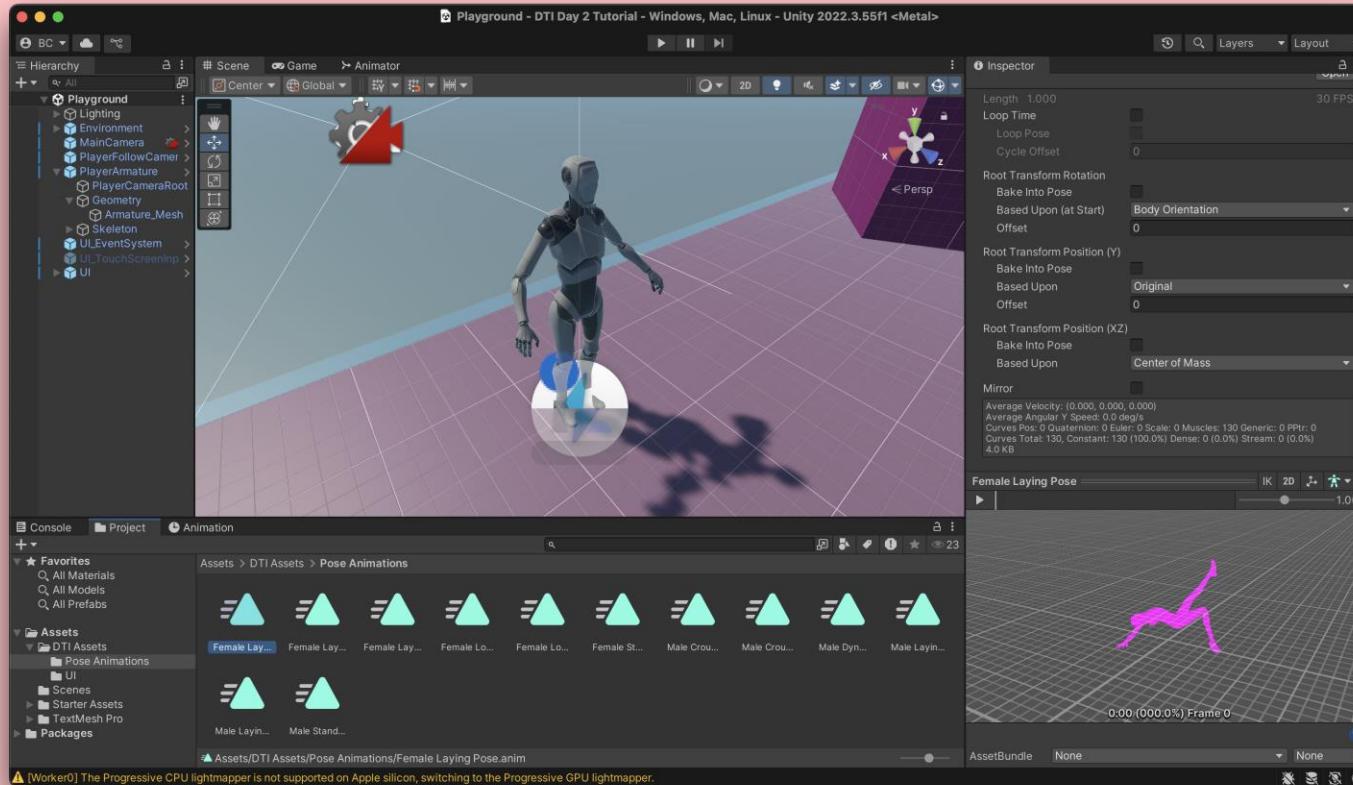


**AWSD or Arrows – Move
Space – Jump
Shift – Sprint
Hold Left-click – Camera**

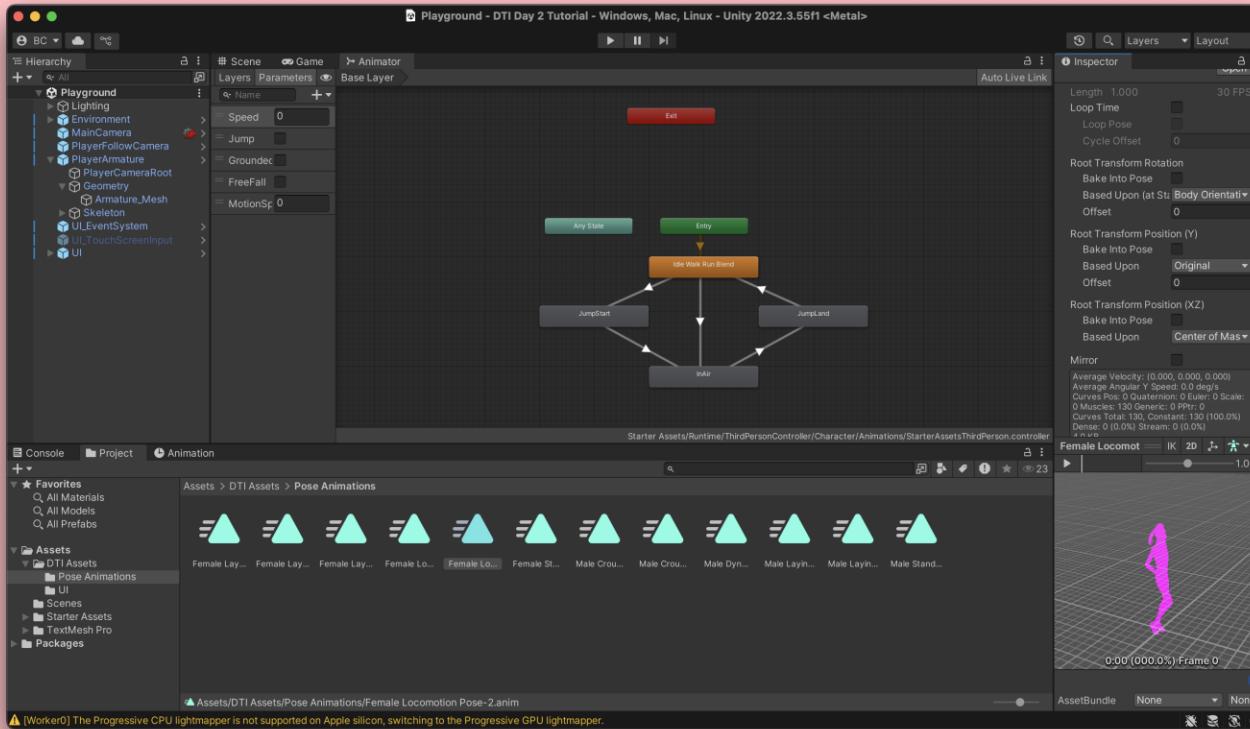
Unity Animator

- **Select poses**
- **Set-up poses in animator**
 - **Transitions**
 - **Triggers**

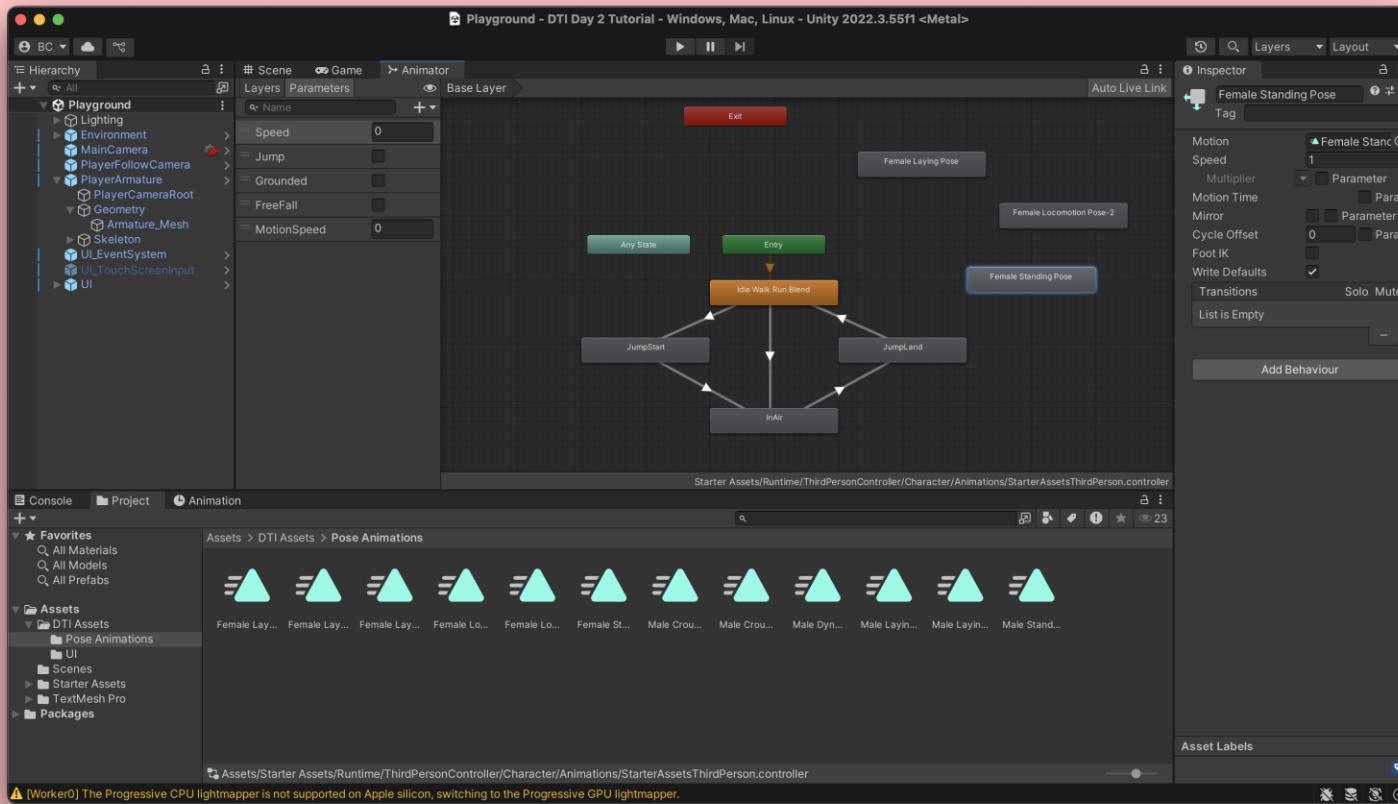
Go to Assets > DTI Assets > Pose Animations, preview the poses and select 3 poses for your model!



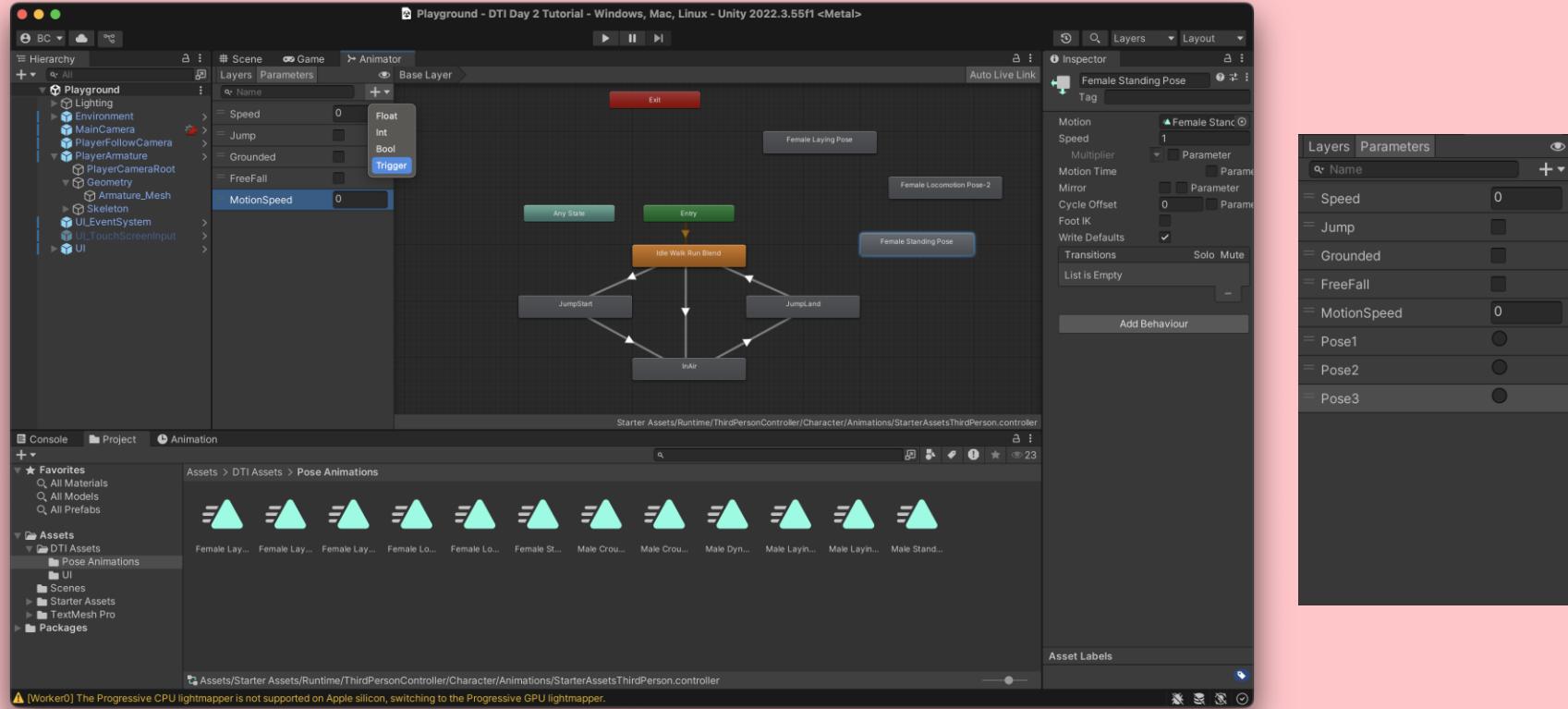
Open up animator at the top of the screen. If its not there go to Assets > Starter Assets > Runtime > ThirdPersonController > Character > Animations and open "StarterAssetsThirdPerson"



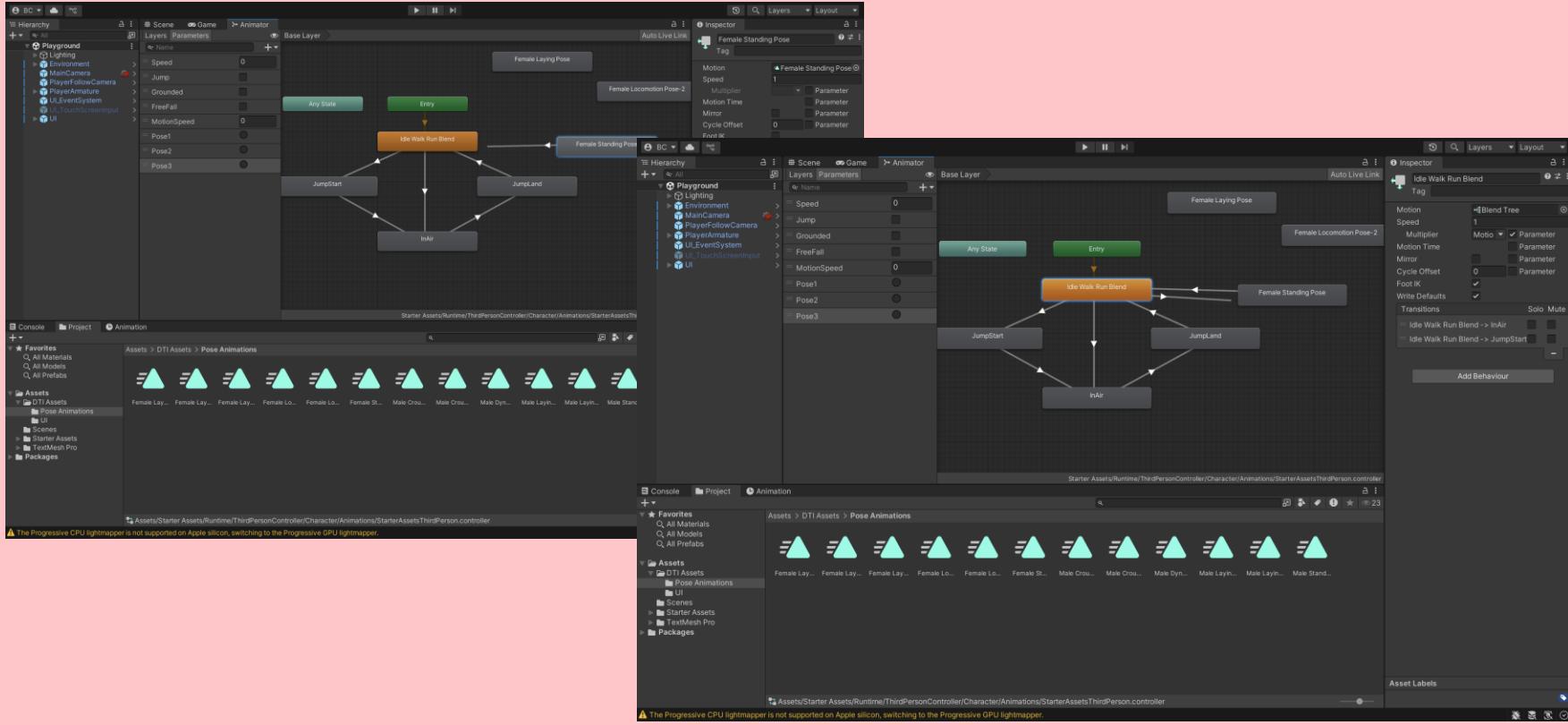
In the animator drag your selected poses onto it



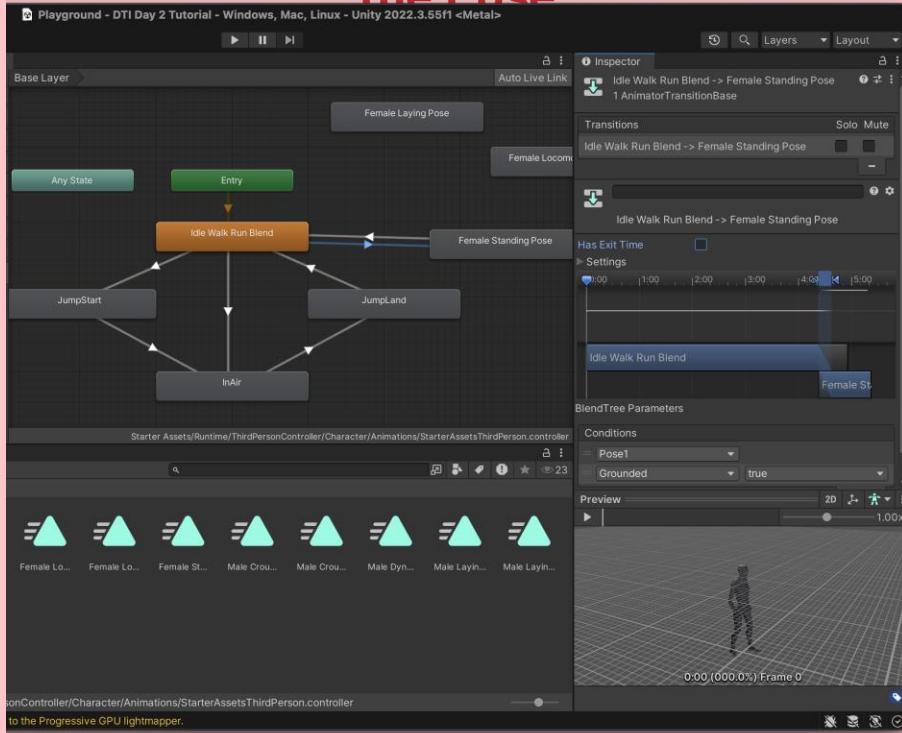
Add 3 trigger parameters and name it "Pose1", "Pose2", "Pose3"



Left-click on one of your poses and click "Make Transition" and connect it to the "Idle Walk Run Blend" and vice versa

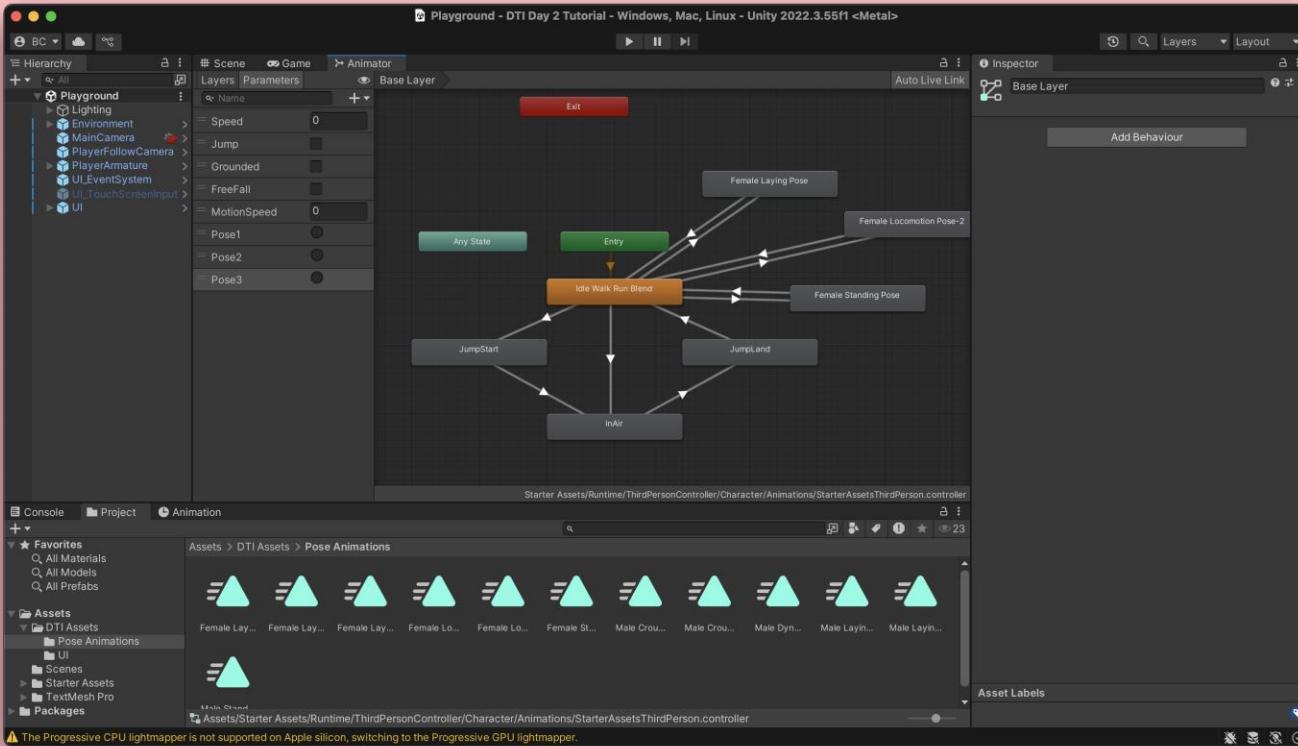


Select transition from Idle to the Pose



- Uncheck "Has Exit Time"
- Add conditions
 - Pose trigger
 - Grounded boolean to true

Do the same for the other poses ensuring you use your different pose triggers for each



Pose Behaviour

- Code model posing

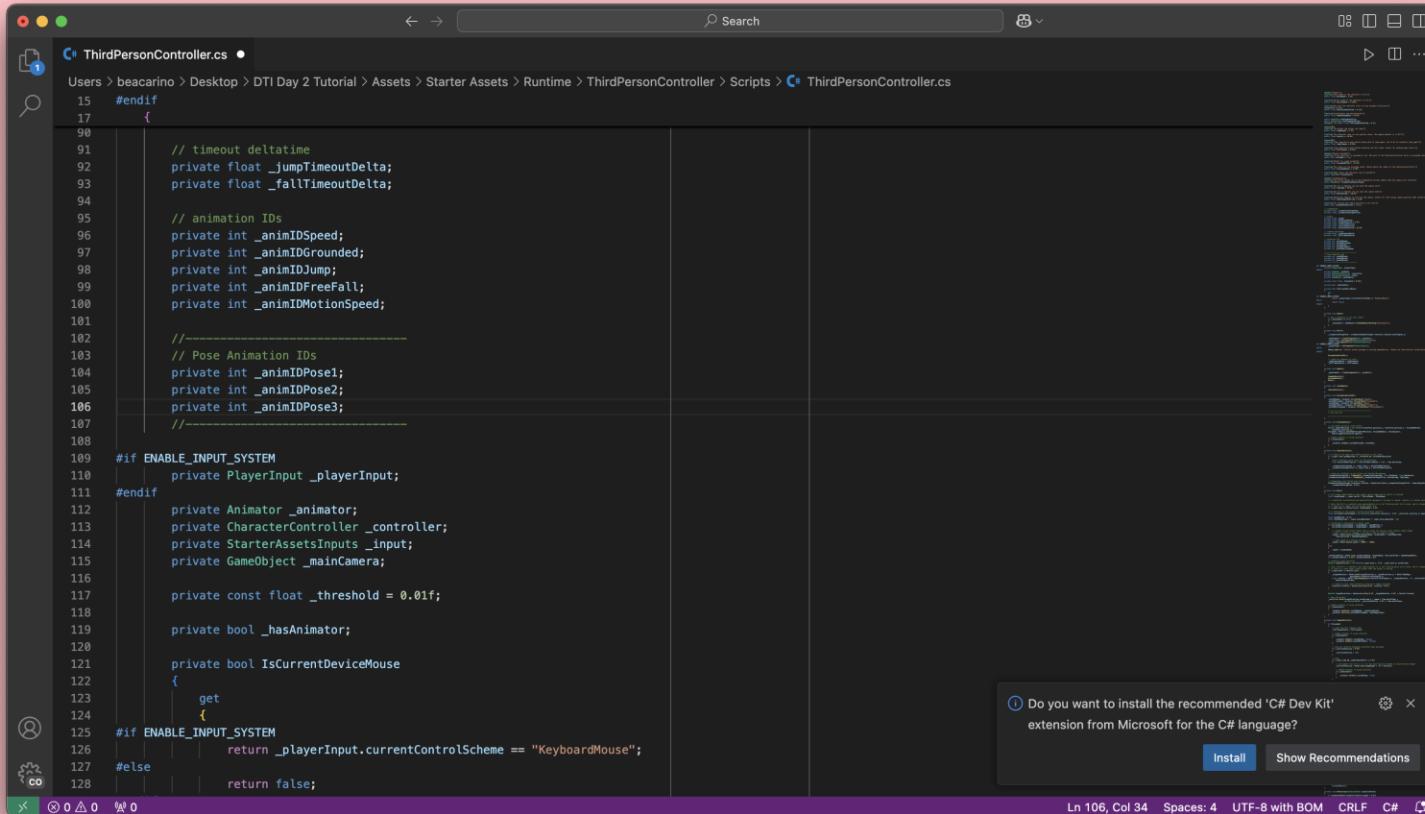
Open Assets > StarterAssets > Runtime > ThirdPersonController > Scripts "ThirdPersonController" C# script

```
ThirdPersonController.cs
Users > beacarino > Desktop > DTI Day 2 Tutorial > Assets > Starter Assets > Runtime > ThirdPersonController > Scripts > ThirdPersonController.cs

1 using JetBrains.Annotations;
2 using UnityEngine;
3 #if ENABLE_INPUT_SYSTEM
4 using UnityEngine.InputSystem;
5 #endif
6
7 /* Note: animations are called via the controller for both the character and capsule using animator null checks
8 */
9
10 namespace StarterAssets
11 {
12     [RequireComponent(typeof(CharacterController))]
13 #if ENABLE_INPUT_SYSTEM
14     [RequireComponent(typeof(PlayerInput))]
15 #endif
16     public class ThirdPersonController : MonoBehaviour
17     {
18         [Header("Player")]
19         [Tooltip("Move speed of the character in m/s")]
20         public float MoveSpeed = 2.0f;
21
22         [Tooltip("Sprint speed of the character in m/s")]
23         public float SprintSpeed = 5.335f;
24
25         [Tooltip("How fast the character turns to face movement direction")]
26         [Range(0.0f, 0.3f)]
27         public float RotationSmoothTime = 0.12f;
28
29         [Tooltip("Acceleration and deceleration")]
30         public float SpeedChangeRate = 10.0f;
31
32         public AudioClip LandingAudioClip;
33         public AudioClip[] FootstepAudioClips;
34         [Range(0, 1)] public float FootstepAudioVolume = 0.5f;
35
36         [Space(10)]
37         [Tooltip("The height the player can jump")]
38         public float JumpHeight = 1.2f;
39
40         [Tooltip("The character uses its own gravity value. The engine default is -9.81f")]
41         public float Gravity = -15.0f;
}

Ln 1, Col 1  Spaces: 4  UTF-8 with BOM  CRLF  C#  ⌂
```

Fill in the blocked out sections with the following code:



```
C# ThirdPersonController.cs
Users > beacarino > Desktop > DTI Day 2 Tutorial > Assets > Starter Assets > Runtime > ThirdPersonController > Scripts > C# ThirdPersonController.cs

15  #endif
16  {
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109 #if ENABLE_INPUT_SYSTEM
110     private PlayerInput _playerInput;
111 #endif
112     private Animator _animator;
113     private CharacterController _controller;
114     private StarterAssetsInputs _input;
115     private GameObject _mainCamera;
116
117     private const float _threshold = 0.01f;
118
119     private bool _hasAnimator;
120
121     private bool IsCurrentDeviceMouse
122     {
123         get
124     }
125 #if ENABLE_INPUT_SYSTEM
126     private bool IsCurrentDeviceMouse
127     {
128         get
129     }
130
131     private void Update()
132     {
133         if (IsCurrentDeviceMouse)
134         {
135             if (_hasAnimator)
136             {
137                 _animator.SetFloat("VerticalWalkSpeed", -_input.Vertical);
138                 _animator.SetFloat("HorizontalWalkSpeed", _input.Horizontal);
139             }
140
141             if (_input.Jump)
142             {
143                 if (_input.Jump)
144                 {
145                     _animator.Play(_animIDJump);
146                 }
147
148                 if (_input.Fall)
149                 {
150                     _animator.Play(_animIDFreeFall);
151                 }
152
153                 if (_input.Motion)
154                 {
155                     _animator.Play(_animIDMotionSpeed);
156                 }
157
158                 if (_input.Crouch)
159                 {
160                     _animator.Play(_animIDCrouch);
161                 }
162
163                 if (_input.Sprint)
164                 {
165                     _animator.Play(_animIDSprint);
166                 }
167
168                 if (_input.Duck)
169                 {
170                     _animator.Play(_animIDDuck);
171                 }
172
173                 if (_input.Run)
174                 {
175                     _animator.Play(_animIDRun);
176                 }
177
178                 if (_input.Walk)
179                 {
180                     _animator.Play(_animIDWalk);
181                 }
182
183                 if (_input.TurnLeft)
184                 {
185                     _animator.Play(_animIDTurnLeft);
186                 }
187
188                 if (_input.TurnRight)
189                 {
190                     _animator.Play(_animIDTurnRight);
191                 }
192
193                 if (_input.Punch)
194                 {
195                     _animator.Play(_animIDPunch);
196                 }
197
198                 if (_input.Kick)
199                 {
200                     _animator.Play(_animIDKick);
201                 }
202
203                 if (_input.Slide)
204                 {
205                     _animator.Play(_animIDSlide);
206                 }
207
208                 if (_input.Sprint)
209                 {
210                     _animator.Play(_animIDSprint);
211                 }
212
213                 if (_input.Duck)
214                 {
215                     _animator.Play(_animIDDuck);
216                 }
217
218                 if (_input.Run)
219                 {
220                     _animator.Play(_animIDRun);
221                 }
222
223                 if (_input.Walk)
224                 {
225                     _animator.Play(_animIDWalk);
226                 }
227
228                 if (_input.TurnLeft)
229                 {
230                     _animator.Play(_animIDTurnLeft);
231                 }
232
233                 if (_input.TurnRight)
234                 {
235                     _animator.Play(_animIDTurnRight);
236                 }
237
238                 if (_input.Punch)
239                 {
240                     _animator.Play(_animIDPunch);
241                 }
242
243                 if (_input.Kick)
244                 {
245                     _animator.Play(_animIDKick);
246                 }
247
248                 if (_input.Slide)
249                 {
250                     _animator.Play(_animIDSlide);
251                 }
252
253                 if (_input.Jump)
254                 {
255                     _animator.Play(_animIDJump);
256                 }
257
258                 if (_input.Fall)
259                 {
260                     _animator.Play(_animIDFreeFall);
261                 }
262
263                 if (_input.Motion)
264                 {
265                     _animator.Play(_animIDMotionSpeed);
266                 }
267
268                 if (_input.Crouch)
269                 {
270                     _animator.Play(_animIDCrouch);
271                 }
272
273                 if (_input.Sprint)
274                 {
275                     _animator.Play(_animIDSprint);
276                 }
277
278                 if (_input.Duck)
279                 {
280                     _animator.Play(_animIDDuck);
281                 }
282
283                 if (_input.Run)
284                 {
285                     _animator.Play(_animIDRun);
286                 }
287
288                 if (_input.Walk)
289                 {
290                     _animator.Play(_animIDWalk);
291                 }
292
293                 if (_input.TurnLeft)
294                 {
295                     _animator.Play(_animIDTurnLeft);
296                 }
297
298                 if (_input.TurnRight)
299                 {
299                     _animator.Play(_animIDTurnRight);
300                 }
301
302                 if (_input.Punch)
303                 {
304                     _animator.Play(_animIDPunch);
305                 }
306
307                 if (_input.Kick)
308                 {
309                     _animator.Play(_animIDKick);
310                 }
311
312                 if (_input.Slide)
313                 {
314                     _animator.Play(_animIDSlide);
315                 }
316
317                 if (_input.Jump)
318                 {
319                     _animator.Play(_animIDJump);
320                 }
321
322                 if (_input.Fall)
323                 {
324                     _animator.Play(_animIDFreeFall);
325                 }
326
327                 if (_input.Motion)
328                 {
329                     _animator.Play(_animIDMotionSpeed);
330                 }
331
332                 if (_input.Crouch)
333                 {
334                     _animator.Play(_animIDCrouch);
335                 }
336
337                 if (_input.Sprint)
338                 {
339                     _animator.Play(_animIDSprint);
340                 }
341
342                 if (_input.Duck)
343                 {
344                     _animator.Play(_animIDDuck);
345                 }
346
347                 if (_input.Run)
348                 {
349                     _animator.Play(_animIDRun);
350                 }
351
352                 if (_input.Walk)
353                 {
354                     _animator.Play(_animIDWalk);
355                 }
356
357                 if (_input.TurnLeft)
358                 {
359                     _animator.Play(_animIDTurnLeft);
360                 }
361
362                 if (_input.TurnRight)
363                 {
364                     _animator.Play(_animIDTurnRight);
365                 }
366
367                 if (_input.Punch)
368                 {
369                     _animator.Play(_animIDPunch);
370                 }
371
372                 if (_input.Kick)
373                 {
374                     _animator.Play(_animIDKick);
375                 }
376
377                 if (_input.Slide)
378                 {
379                     _animator.Play(_animIDSlide);
380                 }
381
382                 if (_input.Jump)
383                 {
384                     _animator.Play(_animIDJump);
385                 }
386
387                 if (_input.Fall)
388                 {
389                     _animator.Play(_animIDFreeFall);
390                 }
391
392                 if (_input.Motion)
393                 {
394                     _animator.Play(_animIDMotionSpeed);
395                 }
396
397                 if (_input.Crouch)
398                 {
399                     _animator.Play(_animIDCrouch);
400                 }
401
402                 if (_input.Sprint)
403                 {
404                     _animator.Play(_animIDSprint);
405                 }
406
407                 if (_input.Duck)
408                 {
409                     _animator.Play(_animIDDuck);
410                 }
411
412                 if (_input.Run)
413                 {
414                     _animator.Play(_animIDRun);
415                 }
416
417                 if (_input.Walk)
418                 {
419                     _animator.Play(_animIDWalk);
420                 }
421
422                 if (_input.TurnLeft)
423                 {
424                     _animator.Play(_animIDTurnLeft);
425                 }
426
427                 if (_input.TurnRight)
428                 {
429                     _animator.Play(_animIDTurnRight);
430                 }
431
432                 if (_input.Punch)
433                 {
434                     _animator.Play(_animIDPunch);
435                 }
436
437                 if (_input.Kick)
438                 {
439                     _animator.Play(_animIDKick);
440                 }
441
442                 if (_input.Slide)
443                 {
444                     _animator.Play(_animIDSlide);
445                 }
446
447                 if (_input.Jump)
448                 {
449                     _animator.Play(_animIDJump);
450                 }
451
452                 if (_input.Fall)
453                 {
454                     _animator.Play(_animIDFreeFall);
455                 }
456
457                 if (_input.Motion)
458                 {
459                     _animator.Play(_animIDMotionSpeed);
460                 }
461
462                 if (_input.Crouch)
463                 {
464                     _animator.Play(_animIDCrouch);
465                 }
466
467                 if (_input.Sprint)
468                 {
469                     _animator.Play(_animIDSprint);
470                 }
471
472                 if (_input.Duck)
473                 {
474                     _animator.Play(_animIDDuck);
475                 }
476
477                 if (_input.Run)
478                 {
479                     _animator.Play(_animIDRun);
480                 }
481
482                 if (_input.Walk)
483                 {
484                     _animator.Play(_animIDWalk);
485                 }
486
487                 if (_input.TurnLeft)
488                 {
489                     _animator.Play(_animIDTurnLeft);
490                 }
491
492                 if (_input.TurnRight)
493                 {
494                     _animator.Play(_animIDTurnRight);
495                 }
496
497                 if (_input.Punch)
498                 {
499                     _animator.Play(_animIDPunch);
500                 }
501
502                 if (_input.Kick)
503                 {
504                     _animator.Play(_animIDKick);
505                 }
506
507                 if (_input.Slide)
508                 {
509                     _animator.Play(_animIDSlide);
510                 }
511
512                 if (_input.Jump)
513                 {
514                     _animator.Play(_animIDJump);
515                 }
516
517                 if (_input.Fall)
518                 {
519                     _animator.Play(_animIDFreeFall);
520                 }
521
522                 if (_input.Motion)
523                 {
524                     _animator.Play(_animIDMotionSpeed);
525                 }
526
527                 if (_input.Crouch)
528                 {
529                     _animator.Play(_animIDCrouch);
530                 }
531
532                 if (_input.Sprint)
533                 {
534                     _animator.Play(_animIDSprint);
535                 }
536
537                 if (_input.Duck)
538                 {
539                     _animator.Play(_animIDDuck);
540                 }
541
542                 if (_input.Run)
543                 {
544                     _animator.Play(_animIDRun);
545                 }
546
547                 if (_input.Walk)
548                 {
549                     _animator.Play(_animIDWalk);
550                 }
551
552                 if (_input.TurnLeft)
553                 {
554                     _animator.Play(_animIDTurnLeft);
555                 }
556
557                 if (_input.TurnRight)
558                 {
559                     _animator.Play(_animIDTurnRight);
560                 }
561
562                 if (_input.Punch)
563                 {
564                     _animator.Play(_animIDPunch);
565                 }
566
567                 if (_input.Kick)
568                 {
569                     _animator.Play(_animIDKick);
570                 }
571
572                 if (_input.Slide)
573                 {
574                     _animator.Play(_animIDSlide);
575                 }
576
577                 if (_input.Jump)
578                 {
579                     _animator.Play(_animIDJump);
580                 }
581
582                 if (_input.Fall)
583                 {
584                     _animator.Play(_animIDFreeFall);
585                 }
586
587                 if (_input.Motion)
588                 {
589                     _animator.Play(_animIDMotionSpeed);
590                 }
591
592                 if (_input.Crouch)
593                 {
594                     _animator.Play(_animIDCrouch);
595                 }
596
597                 if (_input.Sprint)
598                 {
599                     _animator.Play(_animIDSprint);
600                 }
601
602                 if (_input.Duck)
603                 {
604                     _animator.Play(_animIDDuck);
605                 }
606
607                 if (_input.Run)
608                 {
609                     _animator.Play(_animIDRun);
610                 }
611
612                 if (_input.Walk)
613                 {
614                     _animator.Play(_animIDWalk);
615                 }
616
617                 if (_input.TurnLeft)
618                 {
619                     _animator.Play(_animIDTurnLeft);
620                 }
621
622                 if (_input.TurnRight)
623                 {
624                     _animator.Play(_animIDTurnRight);
625                 }
626
627                 if (_input.Punch)
628                 {
629                     _animator.Play(_animIDPunch);
630                 }
631
632                 if (_input.Kick)
633                 {
634                     _animator.Play(_animIDKick);
635                 }
636
637                 if (_input.Slide)
638                 {
639                     _animator.Play(_animIDSlide);
640                 }
641
642                 if (_input.Jump)
643                 {
644                     _animator.Play(_animIDJump);
645                 }
646
647                 if (_input.Fall)
648                 {
649                     _animator.Play(_animIDFreeFall);
650                 }
651
652                 if (_input.Motion)
653                 {
654                     _animator.Play(_animIDMotionSpeed);
655                 }
656
657                 if (_input.Crouch)
658                 {
659                     _animator.Play(_animIDCrouch);
660                 }
661
662                 if (_input.Sprint)
663                 {
664                     _animator.Play(_animIDSprint);
665                 }
666
667                 if (_input.Duck)
668                 {
669                     _animator.Play(_animIDDuck);
670                 }
671
672                 if (_input.Run)
673                 {
674                     _animator.Play(_animIDRun);
675                 }
676
677                 if (_input.Walk)
678                 {
679                     _animator.Play(_animIDWalk);
680                 }
681
682                 if (_input.TurnLeft)
683                 {
684                     _animator.Play(_animIDTurnLeft);
685                 }
686
687                 if (_input.TurnRight)
688                 {
689                     _animator.Play(_animIDTurnRight);
690                 }
691
692                 if (_input.Punch)
693                 {
694                     _animator.Play(_animIDPunch);
695                 }
696
697                 if (_input.Kick)
698                 {
699                     _animator.Play(_animIDKick);
700                 }
701
702                 if (_input.Slide)
703                 {
704                     _animator.Play(_animIDSlide);
705                 }
706
707                 if (_input.Jump)
708                 {
709                     _animator.Play(_animIDJump);
710                 }
711
712                 if (_input.Fall)
713                 {
714                     _animator.Play(_animIDFreeFall);
715                 }
716
717                 if (_input.Motion)
718                 {
719                     _animator.Play(_animIDMotionSpeed);
720                 }
721
722                 if (_input.Crouch)
723                 {
724                     _animator.Play(_animIDCrouch);
725                 }
726
727                 if (_input.Sprint)
728                 {
729                     _animator.Play(_animIDSprint);
730                 }
731
732                 if (_input.Duck)
733                 {
734                     _animator.Play(_animIDDuck);
735                 }
736
737                 if (_input.Run)
738                 {
739                     _animator.Play(_animIDRun);
740                 }
741
742                 if (_input.Walk)
743                 {
744                     _animator.Play(_animIDWalk);
745                 }
746
747                 if (_input.TurnLeft)
748                 {
749                     _animator.Play(_animIDTurnLeft);
750                 }
751
752                 if (_input.TurnRight)
753                 {
754                     _animator.Play(_animIDTurnRight);
755                 }
756
757                 if (_input.Punch)
758                 {
759                     _animator.Play(_animIDPunch);
760                 }
761
762                 if (_input.Kick)
763                 {
764                     _animator.Play(_animIDKick);
765                 }
766
767                 if (_input.Slide)
768                 {
769                     _animator.Play(_animIDSlide);
770                 }
771
772                 if (_input.Jump)
773                 {
774                     _animator.Play(_animIDJump);
775                 }
776
777                 if (_input.Fall)
778                 {
779                     _animator.Play(_animIDFreeFall);
780                 }
781
782                 if (_input.Motion)
783                 {
784                     _animator.Play(_animIDMotionSpeed);
785                 }
786
787                 if (_input.Crouch)
788                 {
789                     _animator.Play(_animIDCrouch);
790                 }
791
792                 if (_input.Sprint)
793                 {
794                     _animator.Play(_animIDSprint);
795                 }
796
797                 if (_input.Duck)
798                 {
799                     _animator.Play(_animIDDuck);
800                 }
801
802                 if (_input.Run)
803                 {
804                     _animator.Play(_animIDRun);
805                 }
806
807                 if (_input.Walk)
808                 {
809                     _animator.Play(_animIDWalk);
810                 }
811
812                 if (_input.TurnLeft)
813                 {
814                     _animator.Play(_animIDTurnLeft);
815                 }
816
817                 if (_input.TurnRight)
818                 {
819                     _animator.Play(_animIDTurnRight);
820                 }
821
822                 if (_input.Punch)
823                 {
824                     _animator.Play(_animIDPunch);
825                 }
826
827                 if (_input.Kick)
828                 {
829                     _animator.Play(_animIDKick);
830                 }
831
832                 if (_input.Slide)
833                 {
834                     _animator.Play(_animIDSlide);
835                 }
836
837                 if (_input.Jump)
838                 {
839                     _animator.Play(_animIDJump);
840                 }
841
842                 if (_input.Fall)
843                 {
844                     _animator.Play(_animIDFreeFall);
845                 }
846
847                 if (_input.Motion)
848                 {
849                     _animator.Play(_animIDMotionSpeed);
850                 }
851
852                 if (_input.Crouch)
853                 {
854                     _animator.Play(_animIDCrouch);
855                 }
856
857                 if (_input.Sprint)
858                 {
859                     _animator.Play(_animIDSprint);
860                 }
861
862                 if (_input.Duck)
863                 {
864                     _animator.Play(_animIDDuck);
865                 }
866
867                 if (_input.Run)
868                 {
869                     _animator.Play(_animIDRun);
870                 }
871
872                 if (_input.Walk)
873                 {
874                     _animator.Play(_animIDWalk);
875                 }
876
877                 if (_input.TurnLeft)
878                 {
879                     _animator.Play(_animIDTurnLeft);
880                 }
881
882                 if (_input.TurnRight)
883                 {
884                     _animator.Play(_animIDTurnRight);
885                 }
886
887                 if (_input.Punch)
888                 {
889                     _animator.Play(_animIDPunch);
890                 }
891
892                 if (_input.Kick)
893                 {
894                     _animator.Play(_animIDKick);
895                 }
896
897                 if (_input.Slide)
898                 {
899                     _animator.Play(_animIDSlide);
900                 }
901
902                 if (_input.Jump)
903                 {
904                     _animator.Play(_animIDJump);
905                 }
906
907                 if (_input.Fall)
908                 {
909                     _animator.Play(_animIDFreeFall);
910                 }
911
912                 if (_input.Motion)
913                 {
914                     _animator.Play(_animIDMotionSpeed);
915                 }
916
917                 if (_input.Crouch)
918                 {
919                     _animator.Play(_animIDCrouch);
920                 }
921
922                 if (_input.Sprint)
923                 {
924                     _animator.Play(_animIDSprint);
925                 }
926
927                 if (_input.Duck)
928                 {
929                     _animator.Play(_animIDDuck);
930                 }
931
932                 if (_input.Run)
933                 {
934                     _animator.Play(_animIDRun);
935                 }
936
937                 if (_input.Walk)
938                 {
939                     _animator.Play(_animIDWalk);
940                 }
941
942                 if (_input.TurnLeft)
943                 {
944                     _animator.Play(_animIDTurnLeft);
945                 }
946
947                 if (_input.TurnRight)
948                 {
949                     _animator.Play(_animIDTurnRight);
950                 }
951
952                 if (_input.Punch)
953                 {
954                     _animator.Play(_animIDPunch);
955                 }
956
957                 if (_input.Kick)
958                 {
959                     _animator.Play(_animIDKick);
960                 }
961
962                 if (_input.Slide)
963                 {
964                     _animator.Play(_animIDSlide);
965                 }
966
967                 if (_input.Jump)
968                 {
969                     _animator.Play(_animIDJump);
970                 }
971
972                 if (_input.Fall)
973                 {
974                     _animator.Play(_animIDFreeFall);
975                 }
976
977                 if (_input.Motion)
978                 {
979                     _animator.Play(_animIDMotionSpeed);
980                 }
981
982                 if (_input.Crouch)
983                 {
984                     _animator.Play(_animIDCrouch);
985                 }
986
987                 if (_input.Sprint)
988                 {
989                     _animator.Play(_animIDSprint);
990                 }
991
992                 if (_input.Duck)
993                 {
994                     _animator.Play(_animIDDuck);
995                 }
996
997                 if (_input.Run)
998                 {
999                     _animator.Play(_animIDRun);
1000                }
1001            }
1002        }
1003    }
1004
1005    private void Update()
1006    {
1007        if (IsCurrentDeviceMouse)
1008        {
1009            if (_hasAnimator)
1010            {
1011                _animator.SetFloat("VerticalWalkSpeed", -_input.Vertical);
1012                _animator.SetFloat("HorizontalWalkSpeed", _input.Horizontal);
1013            }
1014
1015            if (_input.Jump)
1016            {
1017                if (_input.Jump)
1018                {
1019                    _animator.Play(_animIDJump);
1020                }
1021
1022                if (_input.Fall)
1023                {
1024                    _animator.Play(_animIDFreeFall);
1025                }
1026
1027                if (_input.Motion)
1028                {
1029                    _animator.Play(_animIDMotionSpeed);
1030                }
1031
1032                if (_input.Crouch)
1033                {
1034                    _animator.Play(_animIDCrouch);
1035                }
1036
1037                if (_input.Sprint)
1038                {
1039                    _animator.Play(_animIDSprint);
1040                }
1041
1042                if (_input.Duck)
1043                {
1044                    _animator.Play(_animIDDuck);
1045                }
1046
1047                if (_input.Run)
1048                {
1049                    _animator.Play(_animIDRun);
1050                }
1051
1052                if (_input.Walk)
1053                {
1054                    _animator.Play(_animIDWalk);
1055                }
1056
1057                if (_input.TurnLeft)
1058                {
1059                    _animator.Play(_animIDTurnLeft);
1060                }
1061
1062                if (_input.TurnRight)
1063                {
1064                    _animator.Play(_animIDTurnRight);
1065                }
1066
1067                if (_input.Punch)
1068                {
1069                    _animator.Play(_animIDPunch);
1070                }
1071
1072                if (_input.Kick)
1073                {
1074                    _animator.Play(_animIDKick);
1075                }
1076
1077                if (_input.Slide)
1078                {
1079                    _animator.Play(_animIDSlide);
1080                }
1081
1082                if (_input.Jump)
1083                {
1084                    _animator.Play(_animIDJump);
1085                }
1086
1087                if (_input.Fall)
1088                {
1089                    _animator.Play(_animIDFreeFall);
1090                }
1091
1092                if (_input.Motion)
1093                {
1094                    _animator.Play(_animIDMotionSpeed);
1095                }
1096
1097                if (_input.Crouch)
1098                {
1099                    _animator.Play(_animIDCrouch);
1100                }
1101
1102                if (_input.Sprint)
1103                {
1104                    _animator.Play(_animIDSprint);
1105                }
1106
1107                if (_input.Duck)
1108                {
1109                    _animator.Play(_animIDDuck);
1110                }
1111
1112                if (_input.Run)
1113                {
1114                    _animator.Play(_animIDRun);
1115                }
1116
1117                if (_input.Walk)
1118                {
1119                    _animator.Play(_animIDWalk);
1120                }
1121
1122                if (_input.TurnLeft)
1123                {
1124                    _animator.Play(_animIDTurnLeft);
1125                }
1126
1127                if (_input.TurnRight)
1128                {
1129                    _animator.Play(_animIDTurnRight);
1130                }
1131
1132                if (_input.Punch)
1133                {
1134                    _animator.Play(_animIDPunch);
1135                }
1136
1137                if (_input.Kick)
1138                {
1139                    _animator.Play(_animIDKick);
1140                }
1141
1142                if (_input.Slide)
1143                {
1144                    _animator.Play(_animIDSlide);
1145                }
1146
1147                if (_input.Jump)
1148                {
1149                    _animator.Play(_animIDJump);
1150                }
1151
1152                if (_input.Fall)
1153                {
1154                    _animator.Play(_animIDFreeFall);
1155                }
1156
1157                if (_input.Motion)
1158                {
1159                    _animator.Play(_animIDMotionSpeed);
1160                }
1161
1162                if (_input.Crouch)
1163                {
1164                    _animator.Play(_animIDCrouch);
1165                }
1166
1167                if (_input.Sprint)
1168                {
1169                    _animator.Play(_animIDSprint);
1170                }
1171
1172                if (_input.Duck)
1173                {
1174                    _animator.Play(_animIDDuck);
1175                }
1176
1177                if (_input.Run)
1178                {
1179                    _animator.Play(_animIDRun);
1180                }
1181
1182                if (_input.Walk)
1183                {
1184                    _animator.Play(_animIDWalk);
1185                }
1186
1187                if (_input.TurnLeft)
1188                {
1189                    _animator.Play(_animIDTurnLeft);
1190                }
1191
1192                if (_input.TurnRight)
1193                {
1194                    _animator.Play(_animIDTurnRight);
1195                }
1196
1197                if (_input.Punch)
1198                {
1199                    _animator.Play(_animIDPunch);
1200                }
1201
1202                if (_input.Kick)
1203                {
1204                    _animator.Play(_animIDKick);
1205                }
1206
1207                if (_input.Slide)
1208                {
1209                    _animator.Play(_animIDSlide);
1210                }
1211
1212                if (_input.Jump)
1213                {
1214                    _animator.Play(_animIDJump);
1215                }
1216
1217                if (_input.Fall)
1218                {
1219                    _animator.Play(_animIDFreeFall);
1220                }
1221
1222                if (_input.Motion)
1223                {
1224                    _animator.Play(_animIDMotionSpeed);
1225                }
1226
1227                if (_input.Crouch)
1228                {
1229                    _animator.Play(_animIDCrouch);
1230                }
1231
1232                if (_input.Sprint)
1233                {
1234                    _animator.Play(_animIDSprint);
1235                }
1236
1237                if (_input.Duck)
1238                {
1239                    _animator.Play(_animIDDuck);
1240                }
1241
1242                if (_input.Run)
1243                {
1244                    _animator.Play(_animIDRun);
1245                }
1246
1247                if (_input.Walk)
1248                {
1249                    _animator.Play(_animIDWalk);
1250                }
1251
1252                if (_input.TurnLeft)
1253                {
1254                    _animator.Play(_animIDTurnLeft);
1255                }
1256
1257                if (_input.TurnRight)
1258                {
1259                    _animator.Play(_animIDTurnRight);
1260                }
1261
1262                if (_input.Punch)
1263                {
1264                    _animator.Play(_animIDPunch);
1265                }
1266
1267                if (_input.Kick)
1268                {
1269                    _animator.Play(_animIDKick);
1270                }
1271
1272                if (_input.Slide)
1273                {
1274                    _animator.Play(_animIDSlide);
1275                }
1276
1277                if (_input.Jump)
1278                {
1279                    _animator.Play(_animIDJump);
1280                }
1281
1282                if (_input.Fall)
1283                {
1284                    _animator.Play(_animIDFreeFall);
1285                }
1286
1287                if (_input.Motion)
1288                {
1289                    _animator.Play(_animIDMotionSpeed);
1290                }
1291
1292                if (_input.Crouch)
1293                {
1294                    _animator.Play(_animIDCrouch);
1295                }
1296
1297                if (_input.Sprint)
1298                {
1299                    _animator.Play(_animIDSprint);
1300                }
1301
1302                if (_input.Duck)
1303                {
1304                    _animator.Play(_animIDDuck);
1305                }
1306
1307                if (_input.Run)
1308                {
1309                    _animator.Play(_animIDRun);
1310                }
1311
1312                if (_input.Walk)
1313                {
1314                    _animator.Play(_animIDWalk);
1315                }
1316
1317                if (_input.TurnLeft)
1318                {
1319                    _animator.Play(_animIDTurnLeft);
1320                }
1321
1322                if (_input.TurnRight)
1323                {
1324                    _animator.Play(_animIDTurnRight);
1325                }
1326
1327                if (_input.Punch)
1328                {
1329                    _animator.Play(_animIDPunch);
1330                }
1331
1332                if (_input.Kick)
1333                {
1334                    _animator.Play(_animIDKick);
1335                }
1336
1337                if (_input.Slide)
1338                {
1339                    _animator.Play(_animIDSlide);
1340                }
1341
1342                if (_input.Jump)
1343                {
1344                    _animator.Play(_animIDJump);
1345                }
1346
1347                if (_input.Fall)
1348                {
1349                    _animator.Play(_animIDFreeFall);
1350                }
1351
1352                if (_input.Motion)
1353                {
1354                    _animator.Play(_animIDMotionSpeed);
1355                }
1356
1357                if (_input.Crouch)
1358                {
1359                    _animator.Play(_animIDCrouch);
1360                }
1361
1362                if (_input.Sprint)
1363                {
1364                    _animator.Play(_animIDSprint);
1365                }
1366
1367                if (_input.Duck)
1368                {
1369                    _animator.Play(_animIDDuck);
1370                }
1371
1372                if (_input.Run)
1373                {
1374                    _animator.Play(_animIDRun);
1375                }
1376
1377                if (_input.Walk)
1378                {
1379                    _animator.Play(_animIDWalk);
1380                }
1381
1382                if (_input.TurnLeft)
1383                {
1384                    _animator.Play(_animIDTurnLeft);
1385                }
1386
1387                if (_input.TurnRight)
1388                {
1389                    _animator.Play(_animIDTurnRight);
1390                }
1391
1392                if (_input.Punch)
1393                {
1394                    _animator.Play(_animIDPunch);
1395                }
1396
1397                if (_input.Kick)
1398                {
1399                    _animator.Play(_animIDKick);
1400                }
1401
1402                if (_input.Slide)
1403                {
1404                    _animator.Play(_animIDSlide);
1405                }
1406
1407                if (_input.Jump)
1408                {
1409                    _animator.Play(_animIDJump);
1410                }
1411
1412                if (_input.Fall)
1413                {
1414                    _animator.Play(_animIDFreeFall);
1415                }
1416
1417                if (_input.Motion)
1418                {
1419                    _animator.Play(_animIDMotionSpeed);
1420                }
1421
1422                if (_input.Crouch)
1423                {
1424                    _animator.Play(_animIDCrouch);
1425                }
1426
1427                if (_input.Sprint)
1428                {
1429                    _animator.Play(_animIDSprint);
1430                }
1431
1432                if (_input.Duck)
1433                {
1434                    _animator.Play(_animIDDuck);
1435                }
1436
1437                if (_input.Run)
1438                {
1439                    _animator.Play(_animIDRun);
1440                }
1441
1442                if (_input.Walk)
1443                {
1444                    _animator.Play(_animIDWalk);
1445                }
1446
1447                if (_input.TurnLeft)
1448                {
1449                    _animator.Play(_animIDTurnLeft);
1450                }
1451
1452                if (_input.TurnRight)
1453                {
1454                    _animator.Play(_animIDTurnRight);
1455                }
1456
1457                if (_input.Punch)
1458                {
1459                    _animator.Play(_animIDPunch);
1460                }
1461
1462                if (_input.Kick)
1463                {
1464                    _animator.Play(_animIDKick);
1465                }
1466
1467                if (_input.Slide)
1468                {
1469                    _animator.Play(_animIDSlide);
1470                }
1471
1472                if (_input.Jump)
1473                {
1474                    _animator.Play(_animIDJump);
1475                }
1476
1477                if (_input.Fall)
1478                {
1479                    _animator.Play(_animIDFreeFall);
1480                }
1481
1482                if (_input.Motion)
1483                {
1484                    _animator.Play(_animIDMotionSpeed);
1485                }
1486
1487                if (_input.Crouch)
1488                {
1489                    _animator.Play(_animIDCrouch);
1490                }
1491
1492                if (_input.Sprint)
1493                {
1494                    _animator.Play(_animIDSprint);
1495                }
1496
1497                if (_input.Duck)
1498                {
1499                    _animator.Play(_animIDDuck);
1500                }
1501
1502                if (_input.Run)
1503                {
1504                    _animator.Play(_animIDRun);
1505                }
1506
1507                if (_input.Walk)
1508                {
1509                    _animator.Play(_animIDWalk);
1510                }
1511
1512                if (_input.TurnLeft)
1513                {
1514                    _animator.Play(_animIDTurnLeft);
1515                }
1516
1517                if (_input.TurnRight)
1518                {
1519                    _animator.Play(_animIDTurnRight);
1520                }
1521
1522                if (_input.Punch)
1523                {
1524                    _animator.Play(_animIDPunch);
1525                }
1526
1527                if (_input.Kick)
1528                {
1529                    _animator.Play(_animIDKick);
1530                }
1531
1532                if (_input.Slide)
1533                {
1534                    _animator.Play(_animIDSlide);
1535                }
1536
1537                if (_input.Jump)
1538                {
1539                    _animator.Play(_animIDJump);
1540                }
1541
1542                if (_input.Fall)
1543                {
1544                    _animator.Play(_animIDFreeFall);
1545                }
1546
1547                if (_input.Motion)
1548                {
1549                    _animator.Play(_animIDMotionSpeed);
1550                }
1551
1552                if (_input.Crouch)
1553                {
1554                    _animator.Play(_animIDCrouch);
1555                }
1556
1557                if (_input.Sprint)
1558                {
1559                    _animator.Play(_animIDSprint);
1560                }
1561
1562                if (_input.Duck)
1563                {
1564                    _animator.Play(_animIDDuck);
1565                }
1566
1567                if (_input.Run)
1568                {
1569                    _animator.Play(_animIDRun);
1570                }
1571
1572                if (_input.Walk)
1573                {
1574                    _animator.Play(_animIDWalk);
1575                }
1576
1577                if (_input.TurnLeft)
1578                {
1579                    _animator.Play(_animIDTurnLeft);
1580                }
1581
1582                if (_input.TurnRight)
1583                {
1584                    _animator.Play(_animIDTurnRight);
1585                }
1586
1587                if (_input.Punch)
1588                {
1589                    _animator.Play(_animIDPunch);
1590                }
1591
1592                if (_input.Kick)
1593                {
1594                    _animator.Play(_animIDKick);
1595                }
1596
1597                if (_input.Slide)
1598                {
1599                    _animator.Play(_animIDSlide);
1600                }
1601
1602                if (_input.Jump)
1603                {
1
```

Fill in the blocked out section with the following code:

```
154 #endif
171     private void LateUpdate()
172     {
173         CameraRotation();
174     }
176
177     private void AssignAnimationIDs()
178     {
179         _animIDSpeed = Animator.StringToHash("Speed");
180         _animIDGrounded = Animator.StringToHash("Grounded");
181         _animIDJump = Animator.StringToHash("Jump");
182         _animIDFreeFall = Animator.StringToHash("FreeFall");
183         _animIDMotionSpeed = Animator.StringToHash("MotionSpeed");
184
185         //-----
186         // Set Pose IDs
187         _animIDPose1 = Animator.StringToHash("Pose1");
188         _animIDPose2 = Animator.StringToHash("Pose2");
189         _animIDPose3 = Animator.StringToHash("Pose3");
190     }
191
192
193     private void GroundedCheck()
194     {
195         // set sphere position, with offset
196         Vector3 spherePosition = new Vector3(transform.position.x, transform.position.y - GroundedOffset,
197             transform.position.z);
198         Grounded = Physics.CheckSphere(spherePosition, GroundedRadius, GroundLayers,
199             QueryTriggerInteraction.Ignore);
200
201         // update animator if using character
202         if (_hasAnimator)
203         {
204             _animator.SetBool(_animIDGrounded, Grounded);
205         }
206     }
207
208     private void CameraRotation()
209     {
210         // if there is an input and camera position is not fixed
```

Do you want to install the recommended 'C# Dev Kit' extension from Microsoft for the C# language?

[Install](#) [Show Recommendations](#)

Ln 189, Col 59 Spaces: 4 UTF-8 with BOM CRLF C#

Fill in the blocked out section with the following code:

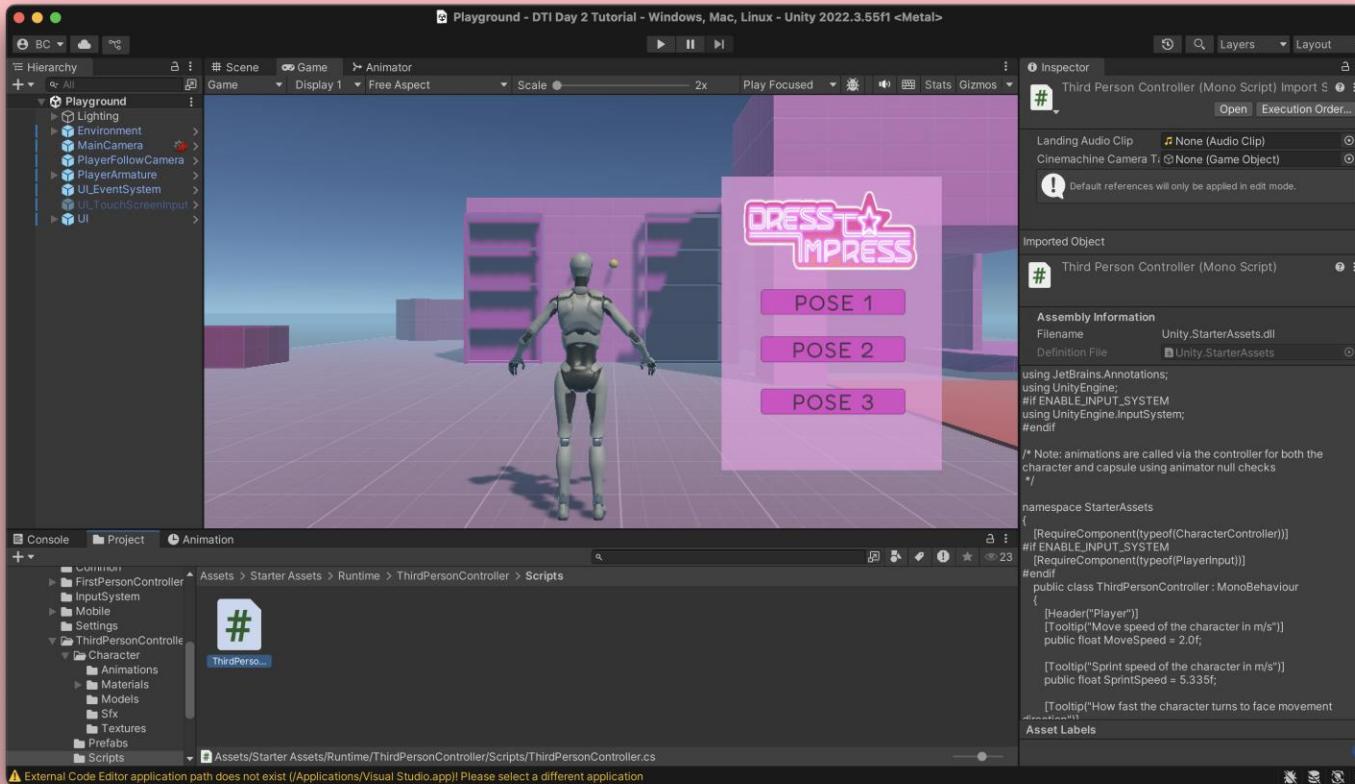
```
154 #endif
388     {
389 }
397 }
398
399 private void OnLand(AnimationEvent animationEvent)
400 {
401     if (animationEvent.animatorClipInfo.weight > 0.5f)
402     {
403         AudioSource.PlayClipAtPoint(LandingAudioClip, transform.TransformPoint(_controller.center), FootstepAudioVolume);
404     }
405 }
406
407 //-----
408 // Pose Button Functions
409 public void Pose1(){
410     if(_hasAnimator){
411         _animator.SetTrigger(_animIDPose1);
412     }
413 }
414
415 public void Pose2(){
416     if(_hasAnimator){
417         _animator.SetTrigger(_animIDPose2);
418     }
419 }
420
421 public void Pose3(){
422     if(_hasAnimator){
423         _animator.SetTrigger(_animIDPose3);
424     }
425 }
426
427 //-----
428 }
429 }
```

Do you want to install the recommended 'C# Dev Kit' extension from Microsoft for the C# language?

[Install](#) [Show Recommendations](#)

Ln 416, Col 31 Spaces: 4 UTF-8 with BOM CRLF C#

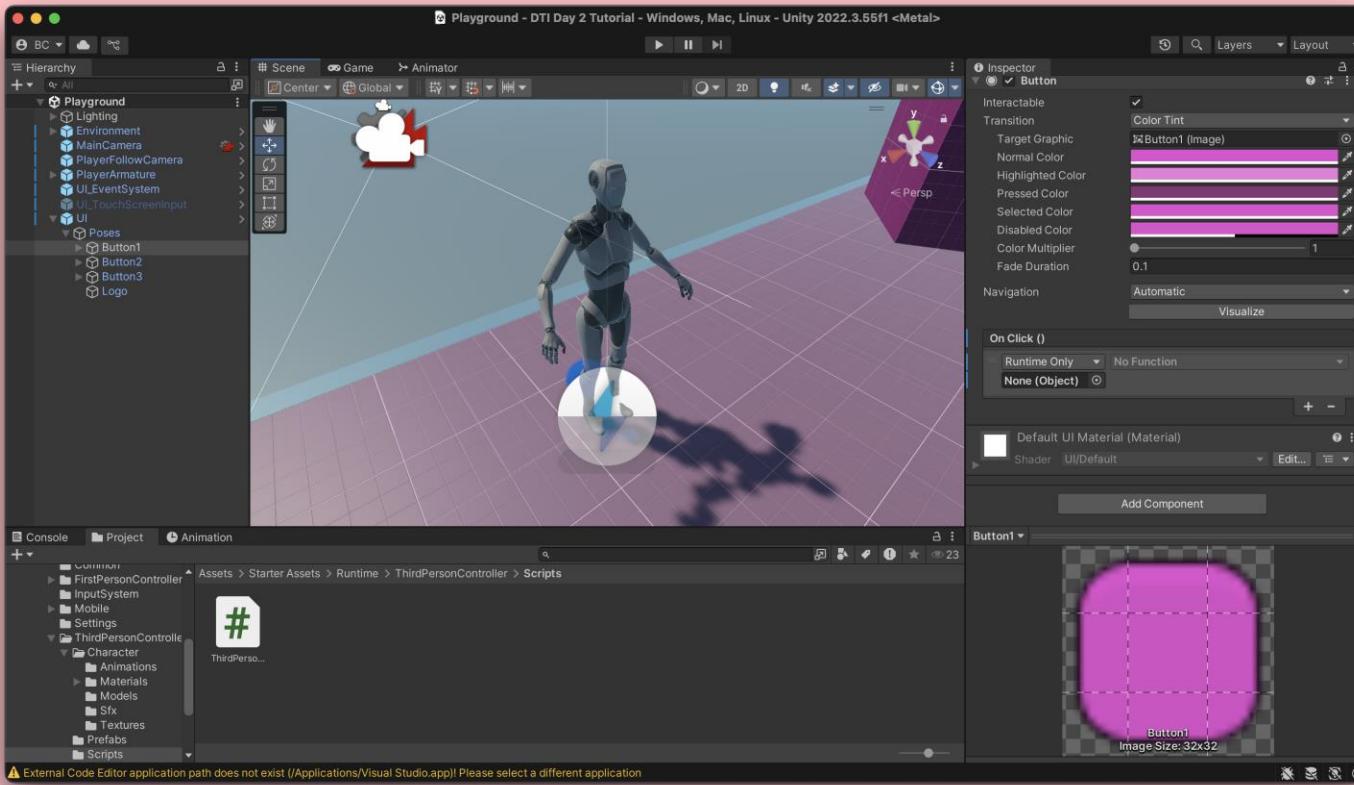
Save code and go back to the editor



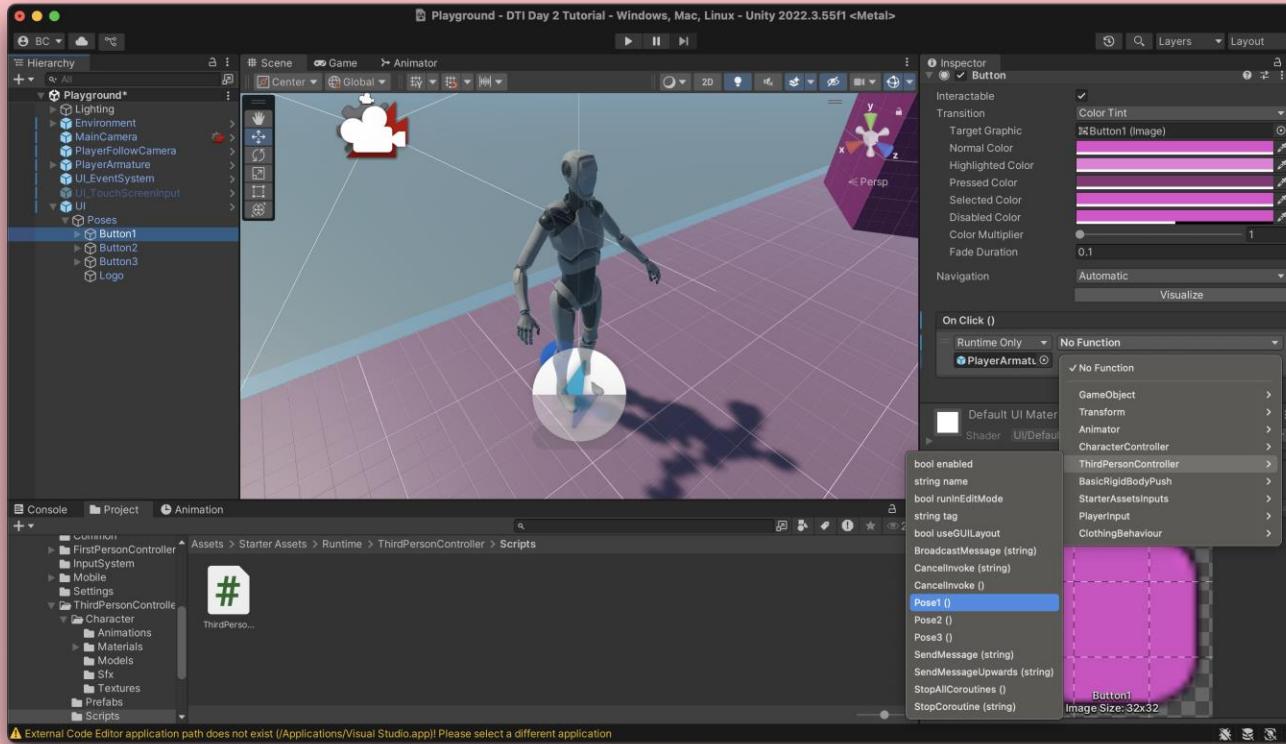
Posing UI

- Connect UI buttons to the created functions from code

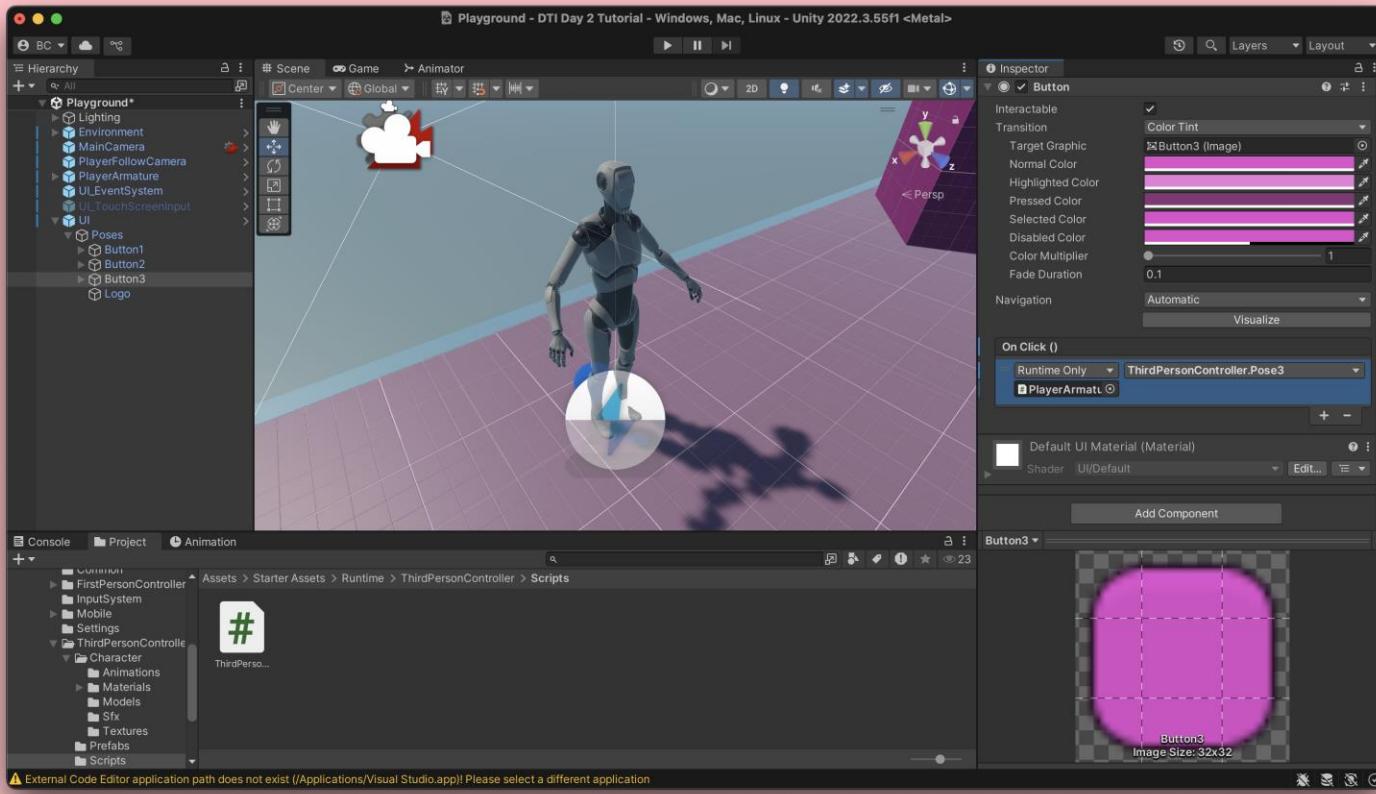
On the left click the drop down for UI > Poses and click a button



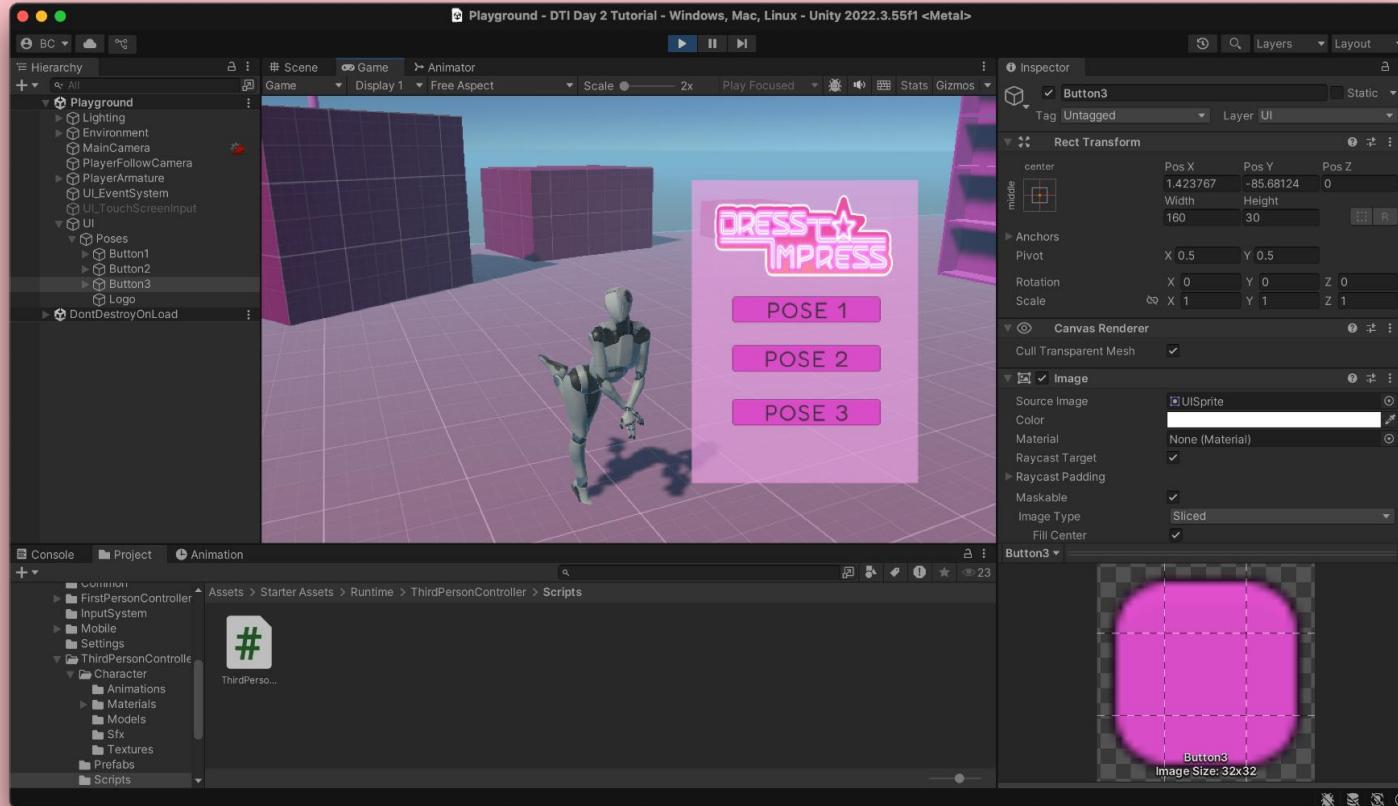
On the right is the button menu, scroll to the bottom and drag "PlayerArmature" from the right to the "On Click" section and select one of the pose functions created.



Do the same for the rest of the buttons ensuring you select the different pose functions for each.



Save your Unity Project, hit play and test all your poses.



Now Your Model Can Pose!

