

2024

2η ΑΣΚΗΣΗ - UNITY

ΓΡΥΛΛΙΑΣ ΓΕΡΑΣΙΜΟΣ 1084651

1. Προσθήκη Κατάβασης από Πλατφόρμα

Για την υλοποίηση του ερωτήματος δημιούργησα ένα νέο script Downhill.cs και χρησιμοποίησα την μέθοδο OnCollisionStay2D για να ανιχνεύω αν ο παίχτης ακουμπάει πάνω σε κάποια πλατφόρμα (οποιοδήποτε object έχει collider). Έπειτα χρησιμοποίησα την κλήση Physics2D.IgnoreCollision (σε true) για να αγνοήσω την σύγκρουση και ο παίχτης να μπορεί να κατεβεί. Καθώς η κλήση δεν είναι εξ ορισμού προσωρινή και ο παίχτης δεν μπορεί να εισέλθει ξανά στην πλατφόρμα, μέσω μιας Coroutine δίνω μια καθυστέρηση 0.7s μέχρι να καλέσω ξανά την Ignore Collision με λογικό false.

Παρατήρηση: Όταν ο παίχτης βρίσκεται πάνω στην πλατφόρμα, δεν αρκούσε να καταργήσουμε την σύγκρουση για να κατέβει, οπότε τοποθέτησα μια κατακόρυφη δύναμη προς τα κάτω, ως ώθηση.

2. Επέκταση του Επιπέδου

Για την υλοποίηση αυτού του ερωτήματος, πρόσθεσα αρκετές νέες πλατφόρμες, που μεταξύ τους έχουν κενά ώστε ο παίχτης να μπορεί να χάνει.

Για την υλοποίηση του τραμπολίνου (Trampoline.cs), χρησιμοποίησα τη συνάρτηση OnCollisionEnter2D που ενεργοποιείται όταν ο παίχτης το ακουμπήσει για 1^η φορά. Όταν γίνει, το έχω ρυθμίσει ώστε να του δίνει μια ώθηση προς τα πάνω για να φαίνεται σαν να πηδάει συνεχώς.

Την ίδια συνάρτηση την χρησιμοποίησα και στον τελικό στόχο, όταν ο παίχτης αγγίζει το σημαϊάκι νίκης (Victory.cs), καθώς και στα καρφιά (SpikesDamage.cs).

Για τις κινούμενες πλατφόρμες, τροποποίησα το MovingPlatform.cs και συγκεκριμένα το finishPos διάνυσμα, ώστε η πλατφόρμα να μην έχει κατεύθυνση κίνησης προς το (0, 0, 0) αλλά προς το διάνυσμα startPos αυξημένο κατά 2. Έτσι, η κάθε πλατφόρμα που κληρονομεί το script θα έχει διαφορετικό χώρο κίνησης.

Σημείωση: οι πλατφόρμες που κινούνται κατακόρυφα κληρονομούν το LadderPlatform.cs που είναι παρόμοιο με το MovingPlatform.cs αλλά στο finishPos αλλάζει ο y άξονας (αυξάνεται κατά 5).

3. Προσθήκη Αντιπάλων με Διαφορετικά AI

Στο πρώτο σκέλος του ερωτήματος, ο Enemy1 (Enemy1Attack.cs) έχει ένα απλό AI ώστε να κινείται δεξιά και αριστερά πάνω στα όρια μιας πλατφόρμας (το πως κινείται σε αυτά τα όρια το εξηγώ στα σχόλια του κώδικα). Ο Player υποχρεούται να τον περάσει κάνοντας Jump, καθώς αν τον ακουμπήσει, ο έχθρος βγάζει το σπαθί και τον σκοτώνει (για την υλοποίηση του death χρησιμοποίησα ένα Coroutine που ο παίχτης στην ουσία χάνει το Collider που έχει και πέφτει).

Για το δεύτερο σκέλος, ο Shooter, κινούμενος στα όρια μιας πλατφόρμας, έχει σύστημα εντοπισμού του Player μέσω ενός Raycast (φαίνεται στο Scene View) που σκανάρει τον χώρο και όταν «δει» τον παίχτη, ενεργοποιείται το Animation που πυροβολεί και τον σκοτώνει. Το Raycast είναι ένα διάνυσμα που αλλάζει συνεχώς κατεύθυνση (κοιτάει προς την κατεύθυνση που κοιτάει και ο παίχτης και πάνω κάτω μεταξύ 0.2 και -0.2 στον Y)

4. Σύστημα καταμέτρησης αποτυχιών

Για την μόνιμη αποθήκευση των αποτυχιών στο παιχνίδι, δημιούργησα μια Public συνάρτηση Loss(), ορισμένη μέσα στο DeathCount.cs που να μπορώ να την καλώ από όλα τα scripts. Κάθε φορά που ο παίχτης χάνει (από εχθρούς, spikes κλπ.) γίνεται Restart το παιχνίδι με την εντολή

`SceneManager.LoadScene(SceneManager.GetActiveScene().name);`

Από την βιβλιοθήκη UnityEngine.SceneManagement, και στην συνέχεια, με τη συνάρτηση Loss() χειρίζομαι τα PlayerPrefs και συγκεκριμένα τη μεταβλητή "PlayerLosses"

5. Προσθήκη UI

Για την υλοποίηση του UI χρησιμοποίησα ένα canvas για να φαίνεται μόνιμα στο game το death count και ένα για το pause menu που ενεργοποιείται με το πάτημα του P. Δημιούργησα ακόμα ένα game object “UImanager” που περιέχει το UI.cs. Το script αυτό περιέχει συναρτήσεις που ενεργοποιούνται κατάλληλα με το πάτημα των UI κουμπιών Restart Game, Easy, Medium, Hard, Continue. Όταν ο χρήστης πατάει Restart γίνεται επανεκκίνηση και reset το death count, ενώ όταν επιλέγει κάποιο βαθμό δυσκολίας και ύστερα Continue, το σύστημα τυπώνει (για 1.5s) κατάλληλο μήνυμα στην οθόνη (Changes Saved) και η δυσκολία αλλάζει (η ταχύτητα των εχθρών). Επίσης, όταν ο παίχτης φτάσει στο σημειάκι της νίκης, εμφανίζεται κατάλληλο μήνυμα στην οθόνη μέσω του UI.

Link για το Project:

<https://drive.google.com/file/d/1MqdbHOigibVpeuiOkEEnuhitu3293hVS/view?usp=sharing>