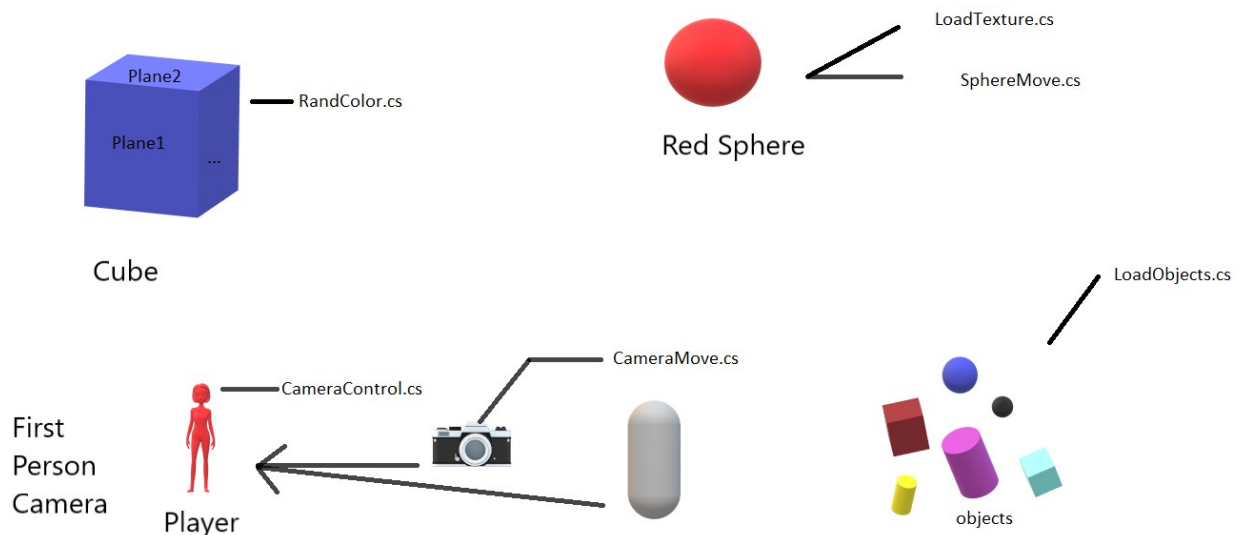


2η προγραμματιστική Άσκηση-Unity

Μέλη ομάδας :

Πηνελόπη Ελευθεριάδη 3221

Δανάη -Αθανασία Τσαούση 3349



(i)ΘΕΣΗ,ΜΕΓΕΘΟΣ ΚΑΙ ΧΡΩΜΑ

Ο **κύβος** που δημιουργούμε στη σκηνή εκτείνεται από το (0,0,0) μέχρι το (100,100,100) και αυτό επιτυγχάνεται με Position : (50,50,50) και Scale : (100,100,100) .Προστίθεται σε αυτόν το component `RandColor.cs` ώστε να έχει τυχαίο χρώμα μέσω της `Random.ColorHSV()` .

ΔΙΑΦΑΝΕΙΑ

Με την συνάρτηση `ChangeMode()` ενεργοποιείται σε runtime το blend mode : Transparent περνώντας τις τιμές που βρίσκονται στο `StandardShaderGUI` για το συγκεκριμένο mode και με την συνάρτηση `ChangeAlpha(Material mat,float alphaVal)` αλλάζουμε σε runtime την τιμή της alpha.

ΔΙΑΡΡΥΘΜΙΣΗ ΚΥΒΟΥ ΓΙΑ ΕΠΟΜΕΝΑ ΕΡΩΤΗΜΑΤΑ

Έχουμε τοποθετήσει ακριβώς πάνω στις πλευρές του 6 planes ώστε να μπορούμε να επιτύχουμε την αναπήδηση των αντικειμένων .Για το κάθε plane,εκτός από την θέση και το μέγεθος του ,το έχουμε ορίσει ως collider και του έχουμε προσθέσει physical material ώστε να πραγματοποιείται η αναπήδηση πάνω του. Επίσης , χρησιμοποιεί όπως και ο κύβος την `RandomColors` ώστε να έχει ίδιο χρώμα και να είναι διαφανές .Έτσι δεν ξεχωρίζει από τον κύβο και μπορούμε να παρατηρήσουμε την κίνηση μέσα στον κύβο.

BACKGROUND

Ορίσαμε το resolution χειροκίνητα μέσω του tab Game σε 1024x768.Το background ακόμη ορίζεται άσπρο μέσω του inspector στη main camera (Solid Color)

(ii)ΣΩΜΑΤΙΔΙΑ

Με το script `LoadObjects` δημιουργούμε τυχαία κύβους ,σφαίρες και κυλίνδρους κάθε φορά που πατάμε το space.Στην update εντοπίζεται το πάτημα του space,στην `LoadOBJ` διαλέγεται τυχαία ποιο αντικείμενο θα εμφανιστεί και αντίστοιχα στις `makeSphere`,`makeCylinder`,`makeCube` και δημιουργούνται τα αντικείμενα. Μέσα σε κάθε συνάρτηση(`makeSphere`,

makeCylinder,makeCube) δημιουργούνται σε τυχαίο μέγεθος και χρώμα αντικείμενα, με κατεύθυνση τυχαία στο [0.1,0.9] και την ονομάζουμε direction και με την χρήση collider ,Rigidbody και τελικά την AddForce υλοποιείται η κίνηση τους. Η αρχική τους ταχύτητα είναι 100.0f. Στην OnCollisionEnter φροντίζουμε να αναπηδάνε τα αντικείμενα με γωνιά ανάκλασης ίδια με την γωνία πρόσκρουσης.

(iii)ΣΦΑΙΡΑ

Δημιουργούμε την σφαίρα μέσα στον κύβο με το κέντρο της στο (50,50,50) και ακτίνα 15(δηλαδή scale 30). Μέσω του script SphereMove πραγματοποιείται η κίνηση της με τα βελάκια στους άξονες x,y και με το =,-(αντί των +,-) στον z ,με χρήση του GetKeyDown για να δεχτεί πιο πλήκτρο πατήσαμε και με το transform για να γίνει η κίνηση. Επίσης, με το script LoadTexture ενεργοποιείται και απενεργοποιείται το texture που μας δόθηκε πατώντας το πλήκτρο t. Προσθέτουμε ένα SphereCollider στην σφαίρα ώστε να αναπηδούν τα μικρά αντικείμενα πάνω της.

(iv)ΚΑΜΕΡΑ

Δημιουργούμε ένα GameObject που το ονομάζουμε Player. Προσθέτουμε σε αυτό το την κάμερα και μια κάψουλα . Αφαιρούμε το collider από την κάψουλα και την τοποθετούμε ακριβώς στην ίδια θέση με την κάμερα. Έπειτα για την κίνηση της κάμερας μέσω ποντικιού προσθέτουμε στη κάμερα το component CameraMove επιλέγοντας ως player στον Inspector τον Player που ορίσαμε αρχικά και για την κίνηση του player μέσω των πλήκτρων w,a,s,d προσθέτουμε στον player το component CameraControl μέσα στο οποίο καθορίσαμε και το walking speed της fps κάμερας.

(v)TEXTURE

Η φόρτωση της υφής γίνεται στην LoadTexture. Πρακτικά υλοποιήσαμε το κουμπί t ως toggle key (on/off). Στην Start() διαβάζουμε το filepath του αρχείου ,έπειτα δημιουργούμε έναν πίνακα filedata με τα bytes της εικόνας ,φτιάχνουμε ένα νέο Texture2D(2,2) tex στο οποίο μετά φορτώνουμε τον byte array με την εντολή tex.LoadImage(fileData). Τέλος κρατάμε το παλιό texture και το ονομάζουμε oldtex ώστε στην update οπότε το t πατιέται 2^η φορά να το επαναφέρουμε στην σφαίρα.

BONUS:

(iii) Πραγματοποιείται η αναπήδηση κατά την σύγκρουση μεταξύ των μικρών αντικειμένων με χρήση του Physical Material “bounceMat” κατά την δημιουργία κάθε αντικειμένου (στις makeSphere, makeCylinder, makeCube).

(v) Για να μπορούν κατά την σύγκρουση τα αντικείμενα με τον παίκτη που κουβαλά την κάμερα και να αναπηδούν ,έχουμε βάλει στο Capsule ένα Capsule Collider.

Links:

[https://docs.unity3d.com/ScriptReference/PhysicMaterial.html?
_ga=2.1855668.443189383.1609704681-1147040739.1607336705](https://docs.unity3d.com/ScriptReference/PhysicMaterial.html?_ga=2.1855668.443189383.1609704681-1147040739.1607336705)

[https://docs.unity3d.com/ScriptReference/Collider.html?
_ga=2.197884970.443189383.1609704681-1147040739.1607336705](https://docs.unity3d.com/ScriptReference/Collider.html?_ga=2.197884970.443189383.1609704681-1147040739.1607336705)

<https://www.youtube.com/watch?v=6C4KfuW2q8Y>
<https://www.youtube.com/watch?v=dLYTwDQmjdo>

<https://docs.unity3d.com/Manual/class-Rigidbody.html>