

ENGLISH VERSION

Adding Animal Counter and Level Timer

Hello everyone! Welcome back to the game development tutorial using UNITY!

In the last video, we have completed our jungle scene where some animals are animated and some are stationary to make it look interesting. Also, we have coded for animal selection which is a requirement for our gameplay.

In this video, we will add the counter which counts the number of animals once selected. Also, we will add a timer in Level 1 to ensure that the level is completed within a stipulated time. If the level didn't get completed in a stipulated time, then we are going to learn how to put in a pop up message that says, "Game over! Please try again."

Let's start by opening UNITY HUB! Select the saved project named 'Food Web'. First, we are going to merge the Level1Controller with GameController to make it easier for functioning. To do that, go to GameController, click on Add Component and type Level 1 Gameplay and press enter. Level1Controller is added successfully and delete the Level1Controller from the hierarchy window.

Now, open the canvas and select the "Fox" tab. You will see the object script is missing as we have moved the script previously attached to the GameController. So, let's update it. We need to update the object for all animals separately. To do that, select 'Fox' and drag GameController to the Object section of Event Trigger component. From the drop down, select Level1Gameplay and click on 'SelectAnimal' option. Similarly, repeat the same steps for all the other animals like deer, lion, leopard, Snake, Eagle, Rhino, Rat, Sparrow and Grasshopper.

Once finished, press Control + S to save the scene directly. Now that we have updated the scripts, let us run and see if it is working! Let's zoom in a bit by dragging the slider. Select the animals and I can see everything is working fine.

Let's go back to the edit mode again. Scale it to minimum by using the slider so as to see the complete screen.

Before proceeding further with the timer, let's organize the canvas a little more. For that, right-click on canvas and click on 'Create Empty' object. Rename it as 'Forest'. Select all the objects in the list from Forest background to Grasshopper and drag them to the forest object. Now, it looks more arranged.

Let's create another UI element called 'Text'. For that, right-click on canvas and select 'UI' and from the list select 'Text' object. Select color option from Hierarchy window and change it to white. Double click on the Text to highlight it. Increase the size of the box, change the font size to 64 and align it centrally. Also, change the height to 72 and change the text to 'Animal Remaining:' Resize the box, make it left aligned and drag it to the top left corner of the screen. Add the number of animals i.e. 14 in the text. Rename it to 'Animals Remaining Text'.

Next, we need to update the text every time an animal is selected. To do that, let's jump to the project window and select Level1Gameplay and double click to open it in Visual Studio.

To update the text every time, first we will access the text label that we have created there. So, to access it, we're going to write the code:

```
using UnityEngine.UI;
```

And inside the public class, we will define

```
{
```

```
public Text animalsRemainingText;
```

But what is the maximum number of animals that we have?

Let's go to defines and write

```
Public static int totalAnimals = 14;
```

Now, go to the Level1Gameplay tab. Next we want to update the number of animals in the text as soon as the animal is selected. For that, we'll define another variable called private integer animals count.

```
Private int animalsCount;
```

To set the value of animals count, we are going to define the start function in the public class

Private void Start()

And in the start function we are going to set the animal count by writing

```
{  
AnimalsCount = Defines.totalAnimals;  
}
```

This function is to be applied to the animals remaining text. So, we can do that by writing

```
animalsRemainingText.text = "Animals Remaining: " +  
animalsCount.ToString();
```

But this is something that we are going to continuously keep on updating. So, what we are going to do, we can create a variable to update animal by writing

Private void updateAnimals()

And copy the animals remaining text function to paste it in the update animals function and reduce the count by writing

```
animalsCount--;
```

What animals count minus minus does is that every time an animal is selected, it reduces the count so as to display it on the screen.

Next, when do we call this animal update function? So, we are going to call it as soon as an animal is selected i.e. in the select animal function. For that, write a code

UpdateAnimals();

So, what we have done here is that first we have declared 2 variables i.e. calling animals remaining text here which is the text box that we have created in the scene and other is animals count which is keeping track of how many total animals are currently present in the game. In this start of the game, we have set animals count to total animals which is the default value of 14 and then we have set the value of animals count to animals remaining text. Then we have created one function called update animals which reduces the number of animals and

displays it on the text every time an animal is selected. Then the update animal function updates the text every time the animal is selected.

Let's go back to the Unity editor and update this function into the scene.

To update it, select the GameController object and drag Animals Remaining Text to the script field in the Inspector window. Now, click on the play button to test it.

You can see that as soon as the animal is selected, the counter keeps decreasing.

After selecting all animals, still the count is showing 3 remaining. So, let's check on the number of animals we have in our game. I can see that we have 11 animals in total so let's update the code accordingly. Switch back to the Defines sheet and change the total animals count from 14 to 11.

Next, I want to show a pop-up screen that says 'Great job! You have completed this level!' as soon as I have finished selecting all the animals. To do that, right-click on Canvas, select UI and click on Panel option. Rename it to 'Game Complete'. Again right-click on Game complete, select UI and click on Image option. Resize & position it as shown on the screen. Change the colour to your choice. Rename it 'Popup Image'.

Right click on the image, select UI and click on Text option. Resize & reposition it appropriately. Change the text to Congratulations!. Tick the option Align by Geometry & Best fit. Change the font size to 128, font size to bold, center the text. Resize to 94 and position it to set appropriately on the screen. Rename it to 'Congratulations Text'.

Again right-click on Congratulations text and select duplicate. Drag it below on the screen. Rename it 'Level Complete Text'. Change the text to 'You have selected all the animals' and change the size to 64. Again duplicate the text and drag it below. Change the text to 'Next Level Starts in...5'

Now, go to the asset in project folder and create a new folder called 'Prefab'. In the Hierarchy window, rename the 'Game Complete' object to 'Level 1 Game Complete'. Drag it to the Prefab folder to make it as a prefab. Now delete it from the hierarchy because we want to instantiate that at the runtime. We do not want it to be present in the scene and only appears when the level is complete.

Next, let's go to the Level1Gameplay script. As soon as the Level 1 is complete, let's define a variable as

Public GameObject levelCompleteScreen;

And we are going to instantiate it in the UpdateAnimal function. We write it as

If(animalCount == 0)

then

{

**GameObject _levelComplete = Instantiate(levelCompleteScreen,
parentCanvas);**

}

Let's define the parent canvas as

Public Transform parentCanvas;

Now, this code will help the message pop up as soon as the level is complete.

Go back to the editor and let unity refresh the script.

Select the GameController object. Once selected, you will see that 2 more public variables that we defined starts appearing. Parent Canvas is the field which will help in showing the new screen. So, drag Canvas and place it in Parent canvas. Then drag Level 1 Complete prefab and place it in the Level Complete Screen. Save your scene and click play to see the result of it. So, as you can see the game is working perfectly so far.

Now, we'll start working on the 'Next level starts in' sentence. Go back to the edit mode and create a timer to play this level which indicates how much time is left to complete the game.

To do that, right-click on Animal Remaining Text and select duplicate. To move it to the right side, get on to the rect transform window, click the fourth box from the left, which should be right at the intersection of right and top. Now rename this as "Time Remaining Text". Also, change the text to 'Time Remaining: 30' and change the alignment to left side. Reposition time remaining & animals remaining text to make it more visible. Disable the Raycast for Animal Remaining

Text so as to not interfere with the animal selection. Similarly, disable it for Time remaining text. Don't forget to save the scene.

Let's code for Time Remaining Text. To do that, go to the Level1Gameplay script in visual studio.

We'll define the variable as

Public Text timeRemainingText;

Next go to defines script and create a new variable as

Public static int totalLevel1Time = 30;

Switch back to Level1Gameplay script tab. Now, we want the timer to reduce time by 1 second until it reaches 0. To do that, we will create an enumerator by writing

IEnumerator Level1Timer();

{

Yield return null;

}

And declare a temporary variable & code for timer count in reverse order as

Int _tempCount = Defines.totalLevel1Time;

While (_tempCount > 0)

{

_tempCount--; *(Read it as tempCount minus minus)*

timeRemainingText.text = "Time Remaining: " + _tempCount.ToString();

yield return new WaitForSeconds(1);

}

Debug.Log ("The timer is complete");

So, what does this code do is that it starts the timer in reverse order and start reducing the count by 1 second.

Now, we want the timer to start as soon as the Level 1 begins. For that, let's create a new function after start function by writing

Public void Init()

{

StartCoroutine(Level1Timer());

This will only start if the function is called.

Go to the start function and define it as

Init();

Go to the defines time as reduce the time from 30 to 10 for testing purpose. Switch to editor and click on play. As I can see the time is not reducing, there's an error found.

Go to the console section next to project section and see the issue. The error tells us that though we have coded the time remaining module but we didn't link it to our game scene. So, let do so.

To do that, select GameController. Drag Time Remaining Text object to the blank field of Level 1 Gameplay. Save your scene and click on play to check again.

Let's leave the screen running and from the Windows section in the top bar, select general and click on console. You will see that the message appears on the screen as soon as 10 seconds are over. So, our timer function is complete for the game.

Go back to the Edit mode. Right-click on Canvas and select 'Create Empty'. Rename it to 'Level1 Texts'. Select the 2 texts and drop it in the Level1 Texts object. This will make the text appear always on the screen.

The next screen that we want is the one which indicates that the time is over when someone is not done finding all the animals. For that, right click on the canvas, select UI and click on 'Panel'. Rename it as "Level1 Time Over".

Right click on it, select UI and click on 'Image' object. Resize & reposition it accordingly in the center. Rename it as 'Popup image'. Change the colour as per your choice. Right click on popup image, select UI and click on Text object. Resize

it appropriately & reposition at the top. Change the font size to 64 and change the text to 'Game Over' in the white box.

Right-click on the text and select duplicate. Drag it to the center and change the text to 'Better Luck Next Time!'. Select the Game over text object and change the font style to bold.

We'll add button to this popup image. To add button, right-click on Popup image, select UI and click on Button object. Resize and reposition it appropriately. Expand the button, select Text option and tick on Best fit option. Change the text to 'Replay'. Rename the button object to 'Replay Button'.

Let's go to the project window and drag Level1 Time Over and make it as a prefab. Delete it from the hierarchy window.

Now, go to the Level1Gameplay script and below time remaining text function, define a public function as

Public GameObject timeOverScreen;

What we want now is as soon as the time is up for Level 1, all the touch functions gets disabled and the game over message should pop up on the screen.

For that, go to the timer function, delete the debug line code and write the new code in place of it as

GameObject _tempScreen = Instantiate(timeOverScreen, parentCanvas);

Go back to the editor and select GameController. Drag Level1 Time Over object from prefab and drop it in the Time Over Screen field. Click on Play and see what happens.

As you can see, the game is working fine. Let's code for the replay button now to make it functional.

For that, go to the assets, go inside scripts, right-click and create new C-sharp script. Rename it as ReplayGame. We're creating this script separately so that we can use the same script in the future for our other games. Double-click on it to open the new Replay game script.

Let's code this now.

We'll delete the start & update function and define a function for replay. To do that, write

```
{
```

```
Public void PlayAgain()
```

And in the beginning define the function as

```
Using UnityEngine.SceneManagement;
```

And then come back to play again function and write

```
{
```

```
SceneManager.LoadScene("Forest Scene");
```

```
}
```

Let the script gets refreshed. Now, go to assets again. Select Prefab and double click on Level1 Time Over object. It will open the Prefab editor. Select Level1 Time Over object. Click on Add component in the inspector window and type 'Replay Game' and press enter.

Select Replay button object where we have on-click event added. Click on + to show the click function. Drag Level1 Time Over object from hierarchy window and drop it in the object field of on click function. From the drop down, select Replay game option and click on Play Again function.

Click on the small arrow to come back again to scene. Click on play and see what happens. As you can see, once I click on replay, the game starts from the beginning with timer getting reset to the maximum.

Go to Level1Gameplay script.

One thing we noticed here is that one more condition will come into focus now. If the player is able to select all the animals in time and congratulations screen appears, then the time timer should get stopped too as we do not want it to appear on the screen in this condition.

There are 2 ways to do that. One way is that we let the timer completely run and say whenever the game gets over, the screen shouldn't get pop up on the

screen. The other way is to define the variable and calling the function to stop the pop-up.

So, let's define a variable after levelCompleteScreen as

Private bool isLevelComplete;

And in the start function, define as

isLevelComplete = false;

In the Level1Timer function, we will modify the while condition as

While (_tempCount > 0 && !isLevelComplete)

It means that until the timer count is greater than 0 or the level is not complete, the timer will keep running otherwise it will break the loop and the code that will run is

If(!isLevelComplete)

```
{  
}
```

And move the instantiate object code inside the if loop and it will show the time over screen.

So, we can say that with animal remaining count function, we are checking the count for animals selected and as soon as the animal remaining count become 0 then we say

isLevelComplete = true;

Just to recap, we have created a variable called isLevelComplete which is set to false in the beginning and as soon as the animal remaining count reaches to zero then the condition isLevelComplete become true which indicates that the level is complete and in the timer, we have put the condition in while that, if the level is complete, skip the code mentioned in the loop and if the level is completed, don't show the time over screen.

Also, go to defines tab and change the total level time from 10 to 30 seconds.

Go back to the editor and click on play. Let's select all the animals within the limited time and see what message pops up.

So, as you can see the congratulatory message pops up on selecting all the animals and time remaining is also stopped. It means that the code written is working properly.

So far, we have completed all the parts of Level 1 game play which is selection of animals in stipulated time and if not we can replay the game.

From the next video onwards, we will jump into the level 2 of this game where we are going to display our selected animals and create a food web out of them through connections.

So, keep practicing & exploring unity.

See you in the next video! Bye bye.

HINDI VERSION

Adding Animal Counter and Level Timer

Hello everyone! UNITY का use करके game development tutorial में आपका स्वागत है!

पिछले video में, हमने अपने जंगल के scene को पूरा किया है जहां कुछ animal animated हैं और कुछ इसे interesting बनाने के लिए static हैं। इसके अलावा, हमने animals section के लिए code किया है जो हमारे game play के लिए एक requirement है।

इस video में, हम एक बार चुने गए animals के numbers count करने वाले काउंटर को add करेंगे। इसके अलावा, हम यह sure करने के लिए कि level 1, एक निश्चित समय में पूरा हुआ है कि नहीं हम level 1 में एक timer add करेंगे देंगे। हम यह भी सीखेंगे कि अगर level 1 एक निश्चित समय में time में पूरा नहीं हुआ, तो कैसे एक pop up message डाला जाए जो कहता है, "Game over ! Please try again."

आइये UNITY HUB को open करके शुरू करते हैं! 'Food Web' नाम से save की गई project को चुनें। सबसे पहले, हम Level1Controller को Game Controller के साथ merge करने जा रहे हैं ताकि यह function करना आसान हो सके। ऐसा करने के लिए, Game Controller पर जाएं, add component पर click करें और level 1 gameplay type करें और enter press करें। Level1 Controller successfully जोड़ा गया है और hierarchy window से Level1Controller को delete कर दें।

अब, canvas open करें और "Fox" tab चुनें। आप देखेंगे कि object script गायब है क्योंकि हमने Game Controller से पहले से attached script को हटा दिया है। तो, हम इसे update करते हैं। हमें सभी जानवरों के लिए अलग से object update करना होगा। ऐसा करने के लिए, 'Fox' चुनें और Game Controller को Event Trigger component के object section में खींचें। drop down से, Level1Gameplay चुनें और 'Select Animal' option पर click करें। इसी तरह, बाकी सभी जानवरों जैसे deer, lion, leopard, Snake, Eagle, Rhino, Rat, Sparrow and Grasshopper के लिए भी यही step repeat करें।

एक बार finish होने के बाद, directly scene को save करने के लिए Control + S press करें। अब जब हमने scripts को update कर लिया है, तो चलें और देखें कि क्या यह काम कर रहा है! आइए slider को थोड़ा खींचकर zoom करें। जानवरों को चुनें और मैं देख सकती हूँ कि सब कुछ ठीक काम कर रहा है।

चलिए फिर से edit mode में लौटते हैं। पूरी screen देखने के लिए slider का use करके इसे minimize करें।

timer के साथ आगे बढ़ने से पहले, आइए canvas को थोड़ा और organize करें। उसके लिए, canvas पर right-click करें और 'Create Empty' object पर click करें। इसका नाम forest रखें।

list में सभी objects Forest background से Grasshopper तक चुनें और उन्हें forest object पर खींचें। अब, यह अधिक arranged दिखता है।

चलो 'Text' नाम का एक और UI element बनाते हैं। उसके लिए, canvas पर right-click करें और 'UI' चुनें और list से 'Text' object चुनें। hierarchy window से colour option चुनें और इसे white में बदलें। इसे highlight करने के लिए text पर double click करें। box का size बढ़ाएँ, font size को 64 करें और इसे centre में align करें। इसके अलावा, height को 72 करें और text को 'Animal Remaining:' में बदलें। box को size दें, इसे align करें और इसे screen के top left corner में खींचें। text में animals की numbers यानी 14 add करें। इसका नाम बदलकर 'Remaining Text' रख दिया।

इसके बाद, हमें हर बार किसी animal के चुने जाने पर text को update करना होगा। ऐसा करने के लिए, चलो project window पर जाएं और Level1Gameplay चुनें और visual studio में इसे open करने के लिए double click करें।

Text को हर बार update करने के लिए, पहले हम उस text label को access करेंगे जो हमने वहाँ बनाया है। इसलिए, इसे access करने के लिए, हम code लिखने जा रहे हैं:

using Unity Engine.UI;

और public class के अंदर हम define करेंगे

{

public Text animals Remaining Text;

लेकिन हमारे पास जानवरों की maximum number क्या है?

चलिए defines पर चलते हैं और लिखें

Public static int total Animals = 14;

अब, Level1Gameplay tab पर जाएं। आगे हम text में animals के number को update करना चाहते हैं। उसके लिए, हम एक अन्य variable को define करेंगे जिसे private integer animals count कहा जाता है।

Private int animals Count;

Animal count के value को set करने के लिए, हम public class में start function को define करने जा रहे हैं

Private void Start ()

और start function में हम लिखकर animal count set करेंगे।

```
{  
AnimalCount = Defines.totalAnimals;  
}
```

इस function को animals remaining text पर apply किया जाना है। इसे, हम ऐसा लिखकर कर सकते हैं

```
animalsRemainingText.text = "Animals Remaining: " +
```

```
animalsCount.ToString ();
```

लेकिन यह कुछ ऐसा है जिसे हम continuously update करते रहेंगे। इसलिए, हम क्या करने जा रहे हैं, हम लिखकर animal को update करने के लिए एक variable बना सकते हैं

Private void updateAnimals()

और animals remaining text function को copy करके update animals function में paste करें और लिखकर count reduce करें

```
animalsCount--;
```

Animals count minus यह करता है कि हर बार जब कोई animal चुना जाता है, तो वह count को reduce कर देगा ताकि उसे screen पर display किया जा सके।

अगला, हम animal update function को कब call करते हैं? , जैसे ही किसी animal को चुना जाता है, हम उसको call करेंगे। उसके लिए, एक code लिखें

```
UpdateAnimals();
```

इसलिए, हमने यहां क्या किया है कि पहले हमने 2 variable declare किया हैं मतलब animals remaining text को यहाँ call करते हैं जो कि वह text box है जिसे हमने scene में बनाया है और दूसरा animals count है जो currently game में total animals की जानकारी रखता हैं। game की इस starting में, हमने animals count को total animals set किया है जो कि default value 14 है और फिर हमने animals count value को animals remaining text के लिए set किया है। फिर हमने update animals के नाम से एक function बनाया है जो animals number को reduce करता है और हर बार जब एक animal को चुना जाता है text पर display करता है कि । फिर update animal function text को हर बार animal के चुने जाने के बाद update करता है।

आइए Unity editor पर वापस जाएं और इस function को scene में update करें।

इसे update करने के लिए, GameController object को चुनें

और inspector windows के script field में Animals Remaining Text को खींचें। अब, test करने के लिए play button पर click करें।

आप देख सकते हैं कि जैसे ही animal को चुना जाता है, counter decrease हो जाता है।

सभी animals को चुनने के बाद, अभी भी count 3 remaining दिखाई दे रही है। तो, चलो हमारे game में हमारे पास कितने animals है check करें। मैं देख सकती हूँ कि हमारे पास total 11 animal हैं इसलिए हम according code को update करें। Defines sheet पर वापस जाएं और total animals count को 14 change करके 11 कर दें।

अगला, मैं एक pop-up screen दिखाना चाहती हूँ जो कहे 'Great job! completed this level!' जैसे ही मैंने सभी animals को select करना करना finish कर लूँ। ऐसा करने के लिए, canvas पर right-click करें, UI चुनें और panel option पर click करें। इसे 'Game Complete' का नाम दें। फिर से game पूरा पर right-click करें, UI चुनें और

इमेज option पर click करें। जैसा screen पर दिखाया है उसके according इसे size और position दें। colour को अपनी पसंद से बदलें। इसे 'Popup Image' का नाम दिया।

image पर right click करें, UI चुनें और text option पर click करें। सही से इसे और resize और reposition करें। Text को change करके Congratulations! कर दें। option Align by Geometry & Best fit align करें। font size को change करके 128 , bold, center the text करें। इसको resize करके 94 करें और और इसे screen पर सही से set करने के लिए position दें। इसका नाम 'Congratulations Text' रखें

फिर से Congratulations text पर right-click करें और duplicate चुनें। इसे screen पर नीचे खींचें। इसे 'Level Complete Text' नाम दें। text को change करके 'You have selected all the animals' कर दें और size को change करके 64 कर दें। फिर से text को duplicate करें और इसे नीचे खींचें। text को change करके 'Next Level Starts in....5'कर दें।

अब, project folder में assest पर जाएं और एक नया folder बनाएं जिसे 'Prefab' कहा जाता है। hierarchy window में , 'Game Complete' object को 'Level 1 Game Complete' नाम दें। इसे Prefab बनाने के लिए Prefab folder में खींचें। अब इसे hierarchy से delete कर दें क्योंकि हम runtime के दौरान इसे instantiate करना चाहते हैं। हम नहीं चाहते हैं कि यह scene में present हो और केवल तब दिखाई दे जब level 1 complete हो जाए।

इसके बाद, Level1Gameplay script पर जाएँ। जैसे ही level 1 complete हो जाता है, चलो एक variable को define करते हैं

Public GameObject levelCompleteScreen;

और हम UpdateAnimal function में इसे instantiate करने जा रहे हैं। हम लिखेंगे

If(animalCount == 0)

then


```
{  
GameObject _levelComplete = Instantiate(levelCompleteScreen,  
parentCanvas);  
}
```

parent canvas को define करते हैं

Public Transform parentCanvas;

अब यह code level 1 complete होते ही message को pop up करने में help करेगा।

Editor के पास वापस जाएं और unity के script को refresh करने दें।

GameController object को चुनें। एक बार चुने जाने के बाद, आप देखेंगे कि 2 और public variable जिन्हें हमने define किया था, दिखाई देने लगे। parent canvas वह field है जो नई screen दिखाने में help करेगा। इसलिए, canvas को खींचें और इसे parent canvas में रखें। फिर level 1 Complete prefab को खींचें और level complete screen में रखें। अपना scene save करें और उसका result देखने के लिए play पर click करें। तो, जैसा कि आप देख सकते हैं कि game अभी तक पूरी तरह से काम कर रहा है।

अब, हम 'Next level starts in' sentence पर काम करना शुरू करेंगे। edit mode पर वापस जाएं और इस level पर को game को खेलने के लिए एक timer बनाएं जो indicate करता है कि game को पूरा करने के लिए कितना time बचा है।

ऐसा करने के लिए, Animal Remaining Text पर right-click करें और duplicate चुनें। इसे right side ले जाने के लिए, rect transform windows पर जाएं, left से fourth box पर click करें, जो right और top के intersection पर right होना चाहिए। अब इसे "Time Remaining Text" नाम दें। इसके अलावा, text को 'Time Remaining: 30' में बदलें और alignment को left side करें। time remaining और animals remaining text को Reposition करें ताकि वह अधिक दिखाई दे।

Raycast for Animal Remaining Text को disable करें ताकि ये animal section में interfere न करे। इसी तरह, इसे time remaining text के लिए disable करें। scene को save करना न भूलें।

Time remaining text के लिए कोड करते हैं। ऐसा करने के लिए, visual studio में Level1Gameplay script पर जाएं।

हम variable को define करेंगे

Public Text timeRemainingText;

अब defines script पर जायें और एक new variable

Public static int totalLevel1Time = 30;

Level1Gameplay script tab पर वापस जाएं। अब, हम चाहते हैं कि timer 1 second से time कम करे जब तक कि यह 0 तक न पहुंच जाए। ऐसा करने के लिए, हम लिखकर एक enumerator बनाएंगे

IEnumerator Level1Timer();

{

Yield return null;

}

और temporary variable declare करें और timer count in reverse order को कोड करें

Int _tempCount =

Defines.totalLevel1Time; While

(_tempCount > 0) {

`_tempCount--;` (Read it as *tempCount minus minus*)

`timeRemainingText.text = "Time Remaining: " +`

`_tempCount.ToString(); yield return new WaitForSeconds(1); }`

`Debug.Log ("The timer is complete");`

तो, यह कोड क्या करता है कि यह timer को reverse order में शुरू करता है और count को 1 second से कम करना शुरू करता है।

अब, हम चाहते हैं कि level 1 शुरू होते ही timer शुरू हो जाए। उसके लिए, चलिए एक नया function after start function बनाते हैं, जिसके लिए ये लिखें

`Public void Init()`

`{`

`StartCoroutine(Level1Timer());`

यह केवल तभी शुरू होगा जब function को call किया जायगा।

start function पर जाएं और इसे define करें

`Init();`

testing purpose के लिए से 30 से कम करके 10 तक करने के लिए defines time पर जाएं। editor पर switch करें और play पर click करें। जैसा कि मैं देख सकता हूँ कि time reduce नहीं हो रहा है, एक error मिला है।

project section के next में console section पर जाएं और issue देखें। error बताता है कि हालांकि हमने time remaining module को code किया है, लेकिन हमने इसे अपने game scene से link नहीं किया है। तो, चलो ऐसा करते हैं।

ऐसा करने के लिए, GameController को चुनें। level 1 gameplay के blank field में Time Remaining Text object को खींचें। अपना scene save करें और फिर से check करने के लिए play पर click करें।

आइए screen run करता हुआ रहने दें और Windows section में Windows section से, general चुनें और console पर click करें। आप देखेंगे कि 10 second over होते ही message screen पर दिखाई देता है। तो, game के लिए हमारा timer function complete हो गया है।

Edit mode पर वापस जाएं। canvas पर right-click करें और 'Create Empty' चुनें। इसे 'Level1 Texts' का नाम दें। 2 texts चुनें और Level1 text object में छोड़ दें। इससे text हमेशा screen पर दिखाई देगा।

अगली screen जो हम चाहते हैं वह है जो indicate करता है कि time over हो गया है जब कोई निश्चित समय में सभी जानवरों को दूँढने का काम पूरा नहीं कर पाता तब। उसके लिए, canvas पर right click करें, UI चुनें और 'Panel' पर click करें। इसे "Level1 Time Over" नाम दें।

इस पर right click करें, UI चुनें और 'Image' object पर click करें। इसे centre में उसके अनुसार position करें और size बदलें और इसे 'Popup image' नाम दें। अपनी pasand के अनुसार colour बदलें। pop up image पर right click करें, UI चुनें और text object पर click करें। इसे size दें और top पर position करें। font size 64 में बदलें और white box में text को 'Game Over' में बदलें।

Text पर right-click करें और duplicate चुनें। इसे centre पर खींचें और text को 'Better Luck Next Time!' में बदलें। Game over text object चुनें और font style को bold में change करें।

हम इस pop up image में button add करेंगे। button add करने के लिए, pop up image पर right-click करें, UI चुनें और button object पर click करें। सही size और position दें। button को expand करें, text option चुनें और Best fit option पर tick करें। text को 'Replay' में बदलें। 'button object का नाम बदलकर 'Replay Button' कर दें।

आइए project windows पर जाएं और Level1 Time Over खींचें और इसे prefab के form में बनाएं। इसे hierarchy window से delete करें।

अब, Level1Gameplay script पर जाएं और time remaining text function के नीचे, एक public function को define करें

Public GameObject timeOverScreen;

हम अब क्या चाहते हैं जैसे ही level 1 के लिए time over हो जाए, सभी touch functions disable हो जायें और game over message पर game screen पर pop up होना चाहिए।

उसके लिए, timer function पर जाएं, debug line code delete करें और इसकी जगह new code लिखें

GameObject _tempScreen = Instantiate(timeOverScreen, parentCanvas);

Editor पर वापस जाएं और GameController को चुनें। Prefab से Level1 Time Over object को खींचें और Time Over Screen field में drop करें। Play पर click करें और देखें कि क्या होता है।

जैसा कि आप देख सकते हैं, game ठीक काम कर रहा है। इसे फिर से function करने के लिए अब replay button के लिए code करें।

उसके लिए, asset पर जाएं, scripts के अंदर जाएं, right-click करें और new C-sharp script बनाएं। इसे ReplayGame नाम दें। हम इस script को अलग से बना रहे हैं ताकि हम future में उसी script का use अपने बाकी games के लिए कर सकें। new Replay game script open करने के लिए उस पर double-click करें।

अब इसे code करते हैं।

हम start और update function को delete कर देंगे और replay के लिए function को define करेंगे। ऐसा करने के लिए, लिखें

```
{
```

Public void PlayAgain()

और beginning में function को इस तरह define करें

Using UnityEngine.SceneManagement;

और play again function पर वापस आएँ और लिखें

```
{  
SceneManager.LoadScene("Forest Scene");  
}
```

Script को refresh होने दें। अब, फिर से asset पर जाएँ। Prefab को चुनें और Level1 time over object पर double click करें। यह Prefab editor open कर देगा। Level1 time over object चुनें। inspector windows में Add component पर click करें और 'Replay Game' type करें और enter press करें।

Replay button object को चुनें जहाँ हमने on-click event add किया है। click function दिखाने के लिए + पर click करें। Level1 time object को hierarchy window से खींचें और click function के object field में drop करें। drop down से, replay game option चुनें और Play Again function पर click करें।

Scene पर वापस आने के लिए small arrow पर click करें। play पर click करें और देखें कि क्या होता है। जैसा कि आप देख सकते हैं, एक बार जब मैं replay पर click करता हूँ, तो game beginning से ही शुरू हो जाता है जिसमें timer maximum पर reset हो जाता है।

Level1Gameplay script पर जाएँ।

एक बात जो हमने यहां पर ध्यान दी है वह यह है कि एक और condition अब ध्यान में आएगी। अगर player time पर सभी animals को चुन लेता है और congratulations screen दिखाई देती है, तो time भी बंद हो जाना चाहिए क्योंकि हम नहीं चाहते हैं कि वह इस condition में screen पर दिखाई दे।

ऐसा करने के 2 तरीके हैं। एक तरीका यह है कि हम timer को पूरी तरह से चलने दें और जब भी game over हो जाता है, screen screen पर pop up नहीं करना चाहिए। दूसरा तरीका variable को define करें और pop up को रोकने के लिए function को call करें।

तो, levelCompleteScreen के form एक variable को define करते हैं

Private bool isLevelComplete;

start function में define करें

isLevelComplete = false;

Level1Timer function में, हम इस condition को modify करेंगे

While (_tempCount > 0 && !isLevelComplete)

इसका मतलब है कि जब तक timer count 0 से ज्यादा है या level 1 complete नहीं होता है, तब तक timer run करेगा otherwise यह loop को break कर देगा और जो code run करेगा

If(!isLevelComplete)

```
{  
}
```

और loop के instantiate object code को move करें और यह time over screen दिखाएगा।

तो, हम कह सकते हैं कि animal remaining count function के साथ, हम चुने हुए animals की count check कर रहे हैं और जैसे ही animal remaining count 0 हो जाता है, तो हम बोलेंगे

isLevelComplete = true;

बस recap के लिए, हमने एक variable बनाया है जिसका नाम है isLevelComplete जो beginning में गलत है और जैसे ही animal remaining count 0 पर पहुंचता है तो condition isLevelComplete true हो जायगी जो indicate करता है कि level 1 पूरा

हो गया है और timer में, while में इस condition को रखें कि if the level is complete, skip the code mentioned in the loop and if the level is completed, don't show the time over screen.

इसके अलावा, defines पर जायें और total level time को 10 से change करके 30 second कर दें।

Editor पर वापस जाएं और play पर click करें। आइए limited time के अन्दर सभी animals को चुनें और देखें कि क्या message pop up होता है।

इसलिए, जैसा कि आप देख सकते हैं कि सभी animals के चुने जाने पर congratulatory message pop up हो जाता है और सभी animals and time remaining function भी stop हो जा रहे हैं। इसका मतलब है कि लिखा गया code ठीक से काम कर रहा है।

अब तक, हमने level 1 game play के सभी parts को पूरा कर लिया है जो कि fix time में animals को चुनना है और अगर नहीं तो हम game को फिर से खेल सकते हैं।

अगले video में हम इस game के level 2 में जायेंगे जहाँ हम अपने चुने हुए animals को display करेंगे और connection के through से उनमें से एक food web तैयार करेंगे।

इसलिए, practise करते रहें और unity को explore करते रहें।

मिलते हैं next video में देखें! Bye Bye!