

Lesson7

Kada razmislimo o prvom projektu on zapravo nije bio igra.
Nego samo bazično pomicanje playera.

Naš player se pomicao i udarao u objekte koje smo mi postavili.

Nije bilo nikakvih elemenata video igre.

U igrama najčešće postoje elementi koje player ne može kontrolirati, kao neprijatelji itd, koji tjeraju playera da reagira.

Da bi player mogao reagirati obično ima neke mogućnosti, kao što su napad ili obrana.

I često imamo neke random elemente koje ne očekujemo, kako bi igra bila što zanimljivija.

Sve te stvari su elementi video igre.

I to ćemo učiti programirati u ovom projektu.

Tasks:

Hranjenje gladnih životinja

Životinje će se stvarati na nasumičnim/random pozicijama

Player će bacati hranu kako bi nahranio te životinje

Idemo krenuti.

Napravimo novi projekt i učitamo starter pack

Ime projekta – **Projekt 3**

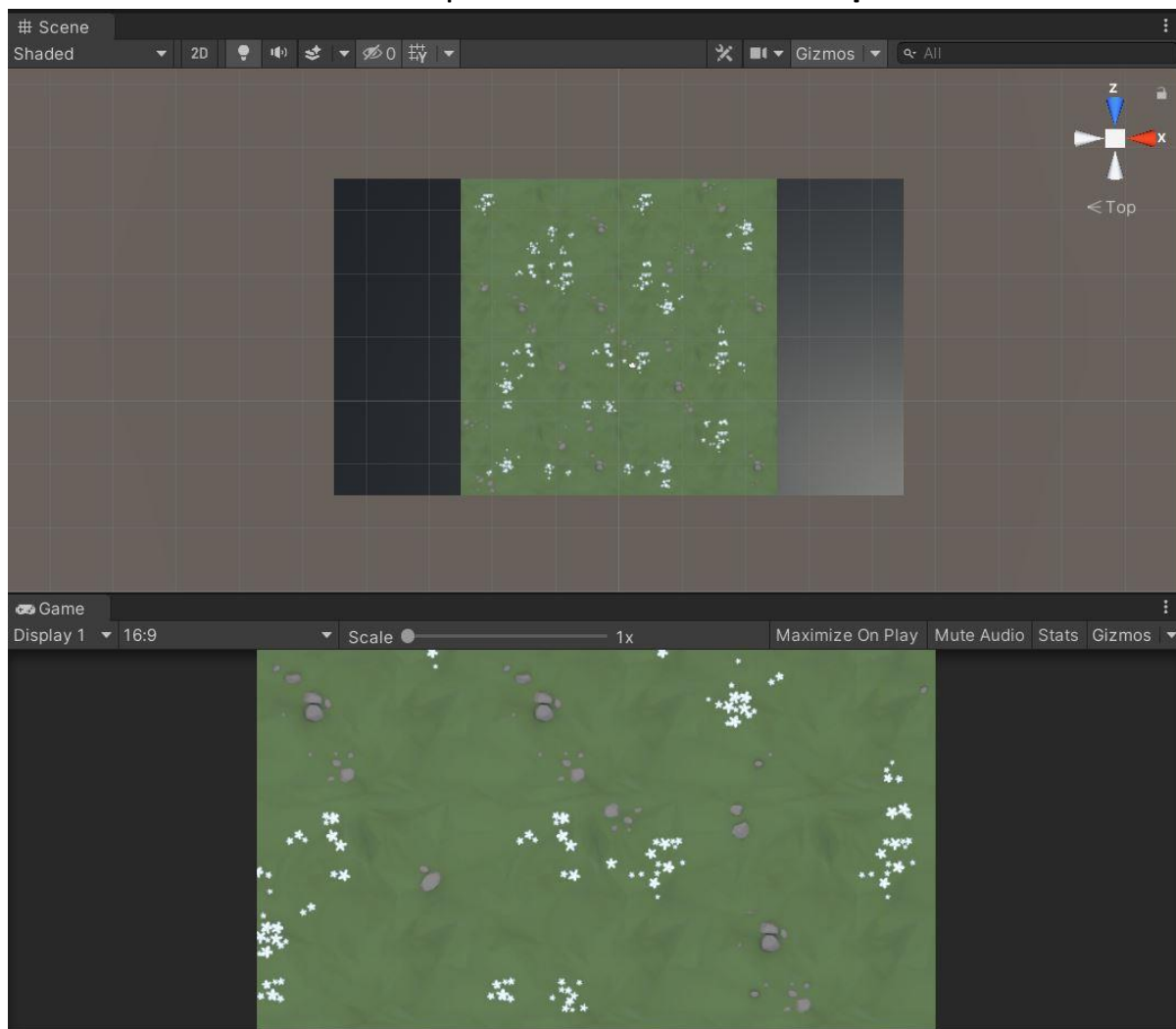
Učitaj projekt.

Učitaj scenu

U project window, assets folder, scene folder, Prototaype2.

Scena

Kliknemo na Y os u sceni da pozicioniramo scenu u **Top Down**



Coruse Library – Human, dodaj playera u Hiararhiju.

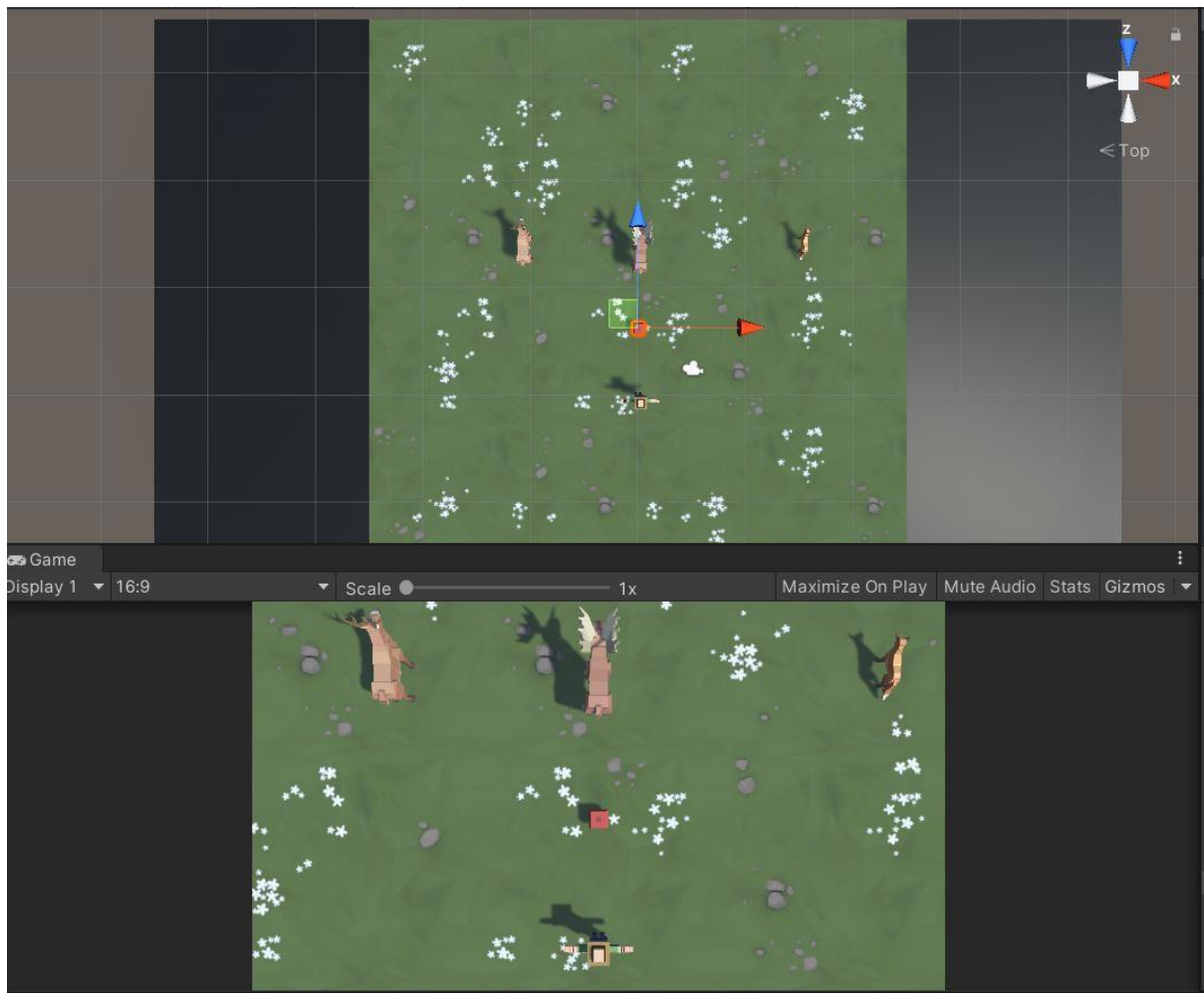
Folder Materials – dodajmo materijal na Ground kako bi promijenili izgled Gorund-a.

Folder Animals – dodajte tri životinje u Hiararhiju.

Folder Food – dodaj jedan Food objekt u Hiararhiju.

Rasporedi objekte po Sceni koristeći Move Tool

Povećaj Food objekt.



Preimenujte glavnog lika u Player.

Player

Napravite novi Folder – Scripts

U njemu novu skriptu – PlayerController i dodajte skriptu kao komponentu Player-u.

Otvorimo skriptu.

Napravimo varijablu koja će u sebi spremiti input.

U **Update** metodi varijabli dodjeljujemo vrijednost inputa.

```
public class PlayerController : MonoBehaviour
{
    public float horizontalInput;

    void Update()
    {
        horizontalInput = Input.GetAxis("Horizontal");
    }
}
```

Sljedeće što trebamo je napraviti da se Player pomiče lijevo i desno.

Zapamtite Time.deltaTime – update-a bazirano na veremenu, ne na frame-u.

```
void Update()
{
    horizontalInput = Input.GetAxis("Horizontal");
    transform.Translate(Vector3.right * Time.deltaTime * horizontalInput);
}
```

Kako bi sada ubrzali Player-a.

Moramo dodati varijablu u koju ćemo spremiti vrijednost brzine.

```

public class PlayerController : MonoBehaviour
{
    public float horizontalInput;
    public float speed;

    void Update()
    {
        horizontalInput = Input.GetAxis("Horizontal");
        transform.Translate(Vector3.right * Time.deltaTime * horizontalInput * speed);
    }
}

```

Sada nam Player može izaći iz mape.

Problem koji moramo riješiti.



Granice kretanja playera

Za ograničiti kretanje playera moramo koristiti **Logiku u programiranju**.

Moramo odlučiti kada želimo da se nešto dogodi, a kada ne.

Za to ćemo koristiti **If Statement**.

Sintaksa:

```
void Update()
{
    if(/*ako je uvijet istinit*/)
    {
        //izvršiti će se ova linija koda
    }
}
```

Sada u IF dodajemo uvijet ako je Player-ova pozicija po X osi manja od 10, zadrži Playera na tom mjestu. Y,Z os moraju ostati isti.

```
void Update()
{
    if(transform.position.x < -10)
    {
        transform.position = new Vector3(-10, transform.position.y, transform.position.z);
    }
}
```

Sada kopiramo postojeći IF statement, - 10 mjenjamo u 10, a znak manje mjenjamo u veće.

```
void Update()
{
    if(transform.position.x < -10)
    {
        transform.position = new Vector3(-10, transform.position.y, transform.position.z);
    }
    if (transform.position.x > 10)
    {
        transform.position = new Vector3(10, transform.position.y, transform.position.z);
    }
}
```

Code Clean Up

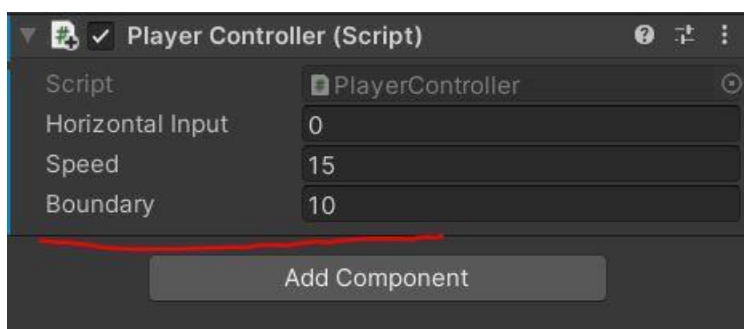
Da bi mogli manipulirati brojem 10 i mijenjati ga, napravit ćemo varijablu u koju ćemo spremiti tu vrijednost.

Vrijednosti koje su nam prije bile minus, i sad ostaju minus, samo ispred varijable ostavljamo minus znak.

```
public float boundary = 10;

void Update()
{
    if(transform.position.x < -boundary)
    {
        transform.position = new Vector3(-boundary, transform.position.y, transform.position.z);
    }
    if (transform.position.x > boundary)
    {
        transform.position = new Vector3(boundary, transform.position.y, transform.position.z);
    }
}
```

Sada u Unity-u možemo mijenjati vrijednosti.



Lesson Recap

Što smo naučili?

If-statement

Greater/Less Than Operators