

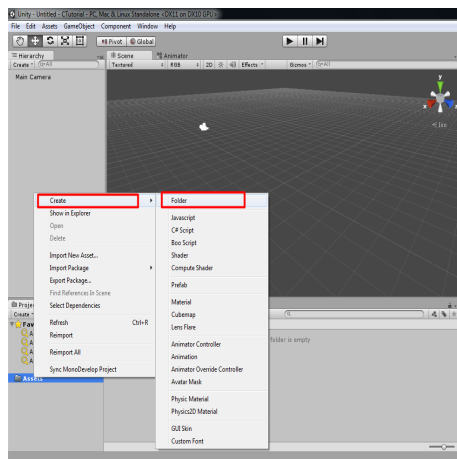
Basic 2 - Membuat C# Script di Unity

Bagikan : [Tweet](#)

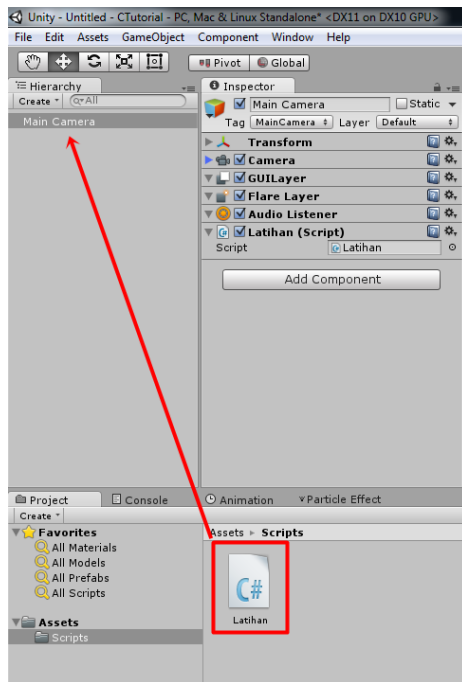
```
1 using UnityEngine;
2 using System.Collections;
3
4 public class Latihan : MonoBehaviour {
5
6     // Use this for initialization
7     void Start () {
8
9     }
10
11     // Update is called once per frame
12     void Update () {
13
14     }
15 }
16
```

Tutorial basic ke 2, dijelaskan bagaimana cara membuat C#Script di unity dan menambahkannya ke objek game.

MULAI



Buat folder baru dan beri nama "Scripts", folder ini berguna untuk menyimpan kumpulan C# Script. Di jendela project, klik kanan Assets - Create - Folder. Di dalam folder "Scripts" klik kanan - create - C# Script dan langsung beri nama "Latihan".



Kemudian tambahkan script tadi ke objek "Main Camera" dengan cara Drag n Drop.

```

1  using UnityEngine;
2  using System.Collections;
3
4  public class Latihan : MonoBehaviour {
5
6      // Use this for initialization
7      void Start () {
8
9      }
10
11     // Update is called once per frame
12     void Update () {
13
14     }
15 }
16

```

Jika sudah ditambahkan klik 2x Script "Latihan" di Folder Scripts, maka akan dibuka Editor MonoDevelop, dan akan ditampilkan listing program seperti gambar di atas.

Ubah Script Latihan menjadi seperti script di bawah :

A. Latihan.cs

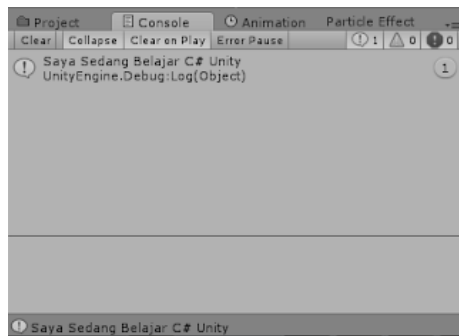
```

using                                     UnityEngine;
using System.Collections;

public class Latihan : MonoBehaviour {

    void                                     Start                                     ()                                     {
    Debug.Log("Saya Sedang Belajar C# Unity");
}
}

```



Jika sudah jalankan game dan lihat output pada jendela "Console".

Penjelasan Script Latihan.cs

using UnityEngine; dan using System.Collections;

Kedua baris program ini akan selalu ada, karena semua script atau perintah yang digunakan bersumber dari Library UnityEngine

public class Latihan

Adalah nama kelas dari script, ingat nama class harus sama dengan nama file C# Script yang ada di Folder "Scripts".

Debug.Log("Saya Sedang Belajar C# Unity")

Debug.Log digunakan untuk menghasilkan output berupa text atau angka, seperti contoh pada tutorial ini dicetak sebuah kalimat "Saya Sedang Belajar C# Unity".

Penting untuk memahami variabel C# yang akan digunakan ketika membuat game, tutorial kali ini akan dijelaskan masing-masing variabel dan cara mendeklarasikannya.

A. Modifier Akses

Oke sebelumnya kita pahami dulu apa itu modifier akses, modifier akses berguna untuk memberikan hak akses kepada user. ada banyak modifier akses, namun pada tutorial ini hanya 3 yang akan dipelajari.

1. **Default**, variabel yang diberikan MA ini hanya dapat digunakan pada class di paket yang sama. penulisan MA ini dikosongkan atau tidak ditulis.

Contoh :

```
string namaPemain = "Somat";
```

2. **Public**, variabel yang diberikan MA ini dapat diakses dan digunakan di class, paket atau method manapun. cara penulisan tambahkan kata public di depan variabel.

Contoh :

```
public int umur = 121;
```

3. **Private**, variabel yang diberikan MA ini hanya dapat diakses oleh atau di dalam kelas itu sendiri. tambahkan "private" sebelum variabel.

Contoh :

```
private string idPemain;
```

B. Variabel

Setelah memahami modifier akses, selanjutnya adalah memahami apa itu variabel dan jenis-jenis variabel. Variabel adalah satuan dasar penyimpanan dalam program. Ada 3 jenis variabel yang umum digunakan, yaitu :

1. Static Variabel

Bisa disebut juga Class Variabel adalah sebuah variabel yang dapat dipanggil di class lain tanpa harus membuat objek kelas itu sendiri.

Contoh : Ada 2 class atau program yaitu satu.cs dan dua.cs, nah dua.cs ingin mengambil variabel yang ada di satu.cs.

Satu.cs

```
public class Satu.cs{  
    public static int variabelDiambil = 9;  
  
}
```

Dua.cs

```
public class Dua.cs{  
int umur = Satu.variabelDiambil; Debug.Log(umur);  
  
}
```

2. Local Variabel

Adalah variabel yang hanya dapat dipanggil pada method dimana variabel tersebut dideklarasikan.

Contoh:

```
public class Makan{  
  
void SiapMakan(){  
string mulaiMakan = "Selamat Makan"; //lokal Variabel  
  
} }  

```

3. World Variabel

Adalah variabel yang dapat digunakan atau dipanggil oleh semua fungsi atau method.

Contoh :

```
public class Makan{  
string mulaiMakan = "Selamat Makan"; //Global Variabel void SiapMakan(){  
}  
  
}
```

4. Constant Variabel

Adalah variabel yang nilainya tetap dan tidak dapat berubah.

Contoh :

```
public class Karakter{  
private const string NAMA_KARAKTER; //Constant Variabel  
  
}
```