

## # Particle Sistem → Alex Yapımı #

create → particle system → herketter partikülerden oluşturursa istey okutucuya schmede

↳ inspecter particlelerde özellüller etkiler

Direction → Zemini, rebelerde sinyale deuen edebek.  
1.00, 5.00 sn gibit

Looping →  sistem dengesel olup okundugun belirler.

Start delay → baslangic to letkusu sona 5.00 sn gibi

" Lifetime" → herbir particlenin ömrini belirler

" Speed" → hit 2.0 gibi

" Size" → Boyut 2.0 kota boyutu.

" Retexture" → herbir particlenin Gemicde bulunan metapoller texture'ini yandaki belirler

" Heater" → baslangic regimini tenn.

Gravity modifier → yerdeki gravitasiyi belirler  
1 amirde yerdeki gravitasiyi, 180'ta dogru dogru gider.

play on awake → baslangic sistemdeki esnekliklerin

nesela katili obiectlerin esnekliklerini tek bir etkiletiler.  
Max particles → particlelerin her biri 1280 particle max  
schmede lirinde obiectteki max particle sayisidir.  
- 1, 20, 100 gibi

Emission → Rate overdrive → son yadew particlenin oluşturusunu  
1. 100 2. her sn'de 2 particle

Rate overdrive →

Bursts → onlu particle durumunu belirler

time: 0.00 min: 100 max: 100  
rate overdrive 0 durus 0.5'de particleler esnekliklerini  
rebet overdrive 0 durus 0.5'de particleler esnekliklerini

Slope → particlelerin düşenini bir seferde verebilirler.

slope → sekil, alextan core yeri konusune esneklikler

angle → aci, 0° olabilir

angle → 0.6'dir. gravity modify, 0.5

Radius → yoncopdur, 0.6'dir. gravity modify, 0.5  
1/2 3 sn'de olur (startspeed) clear dry.

Lifetime 4 olur

size: 2

Renderer → material → create → material → alex →  
texture → freepix.com → flexure texture → indir  
alreded → alex'seq

Wood  
Forest floor

particle system → renderer → material → alpha sector.  
start rotation → 0

değerlendirme → "Color of Particle" →  
Color: basitçe renk, çok komplike bir renk de.

Specular Intensity → particle'ın ışık saçılımının zaten  
sayesindeki boyutunu belirliyor.

→ konuza uygun sekilde boyutları var,  
tümü, asagıda belirtildiği gibi;  
partikülere göre de değişkenlik gösteren  
olarak.

Color of diffuse → etkileşime giren renkler, ışıkla temas eder  
saydalarını belirler.

Saydular alpha: 0, basitçe renk alpha: 206

② # create empty → Game Object → Reserve → Tlk Koduressi  
↳ Inspector → Add component → ışık kodları → News script →  
Create and Add  
Asset → scripte → create C# → oyununesine direkt olarak  
etkinleştir.

```
start()
{
    Debug.log("mehmet");
}
update()
{
    Debug.log("mehmet");
```

int  
float  
string  
bool

float sayi = 10.0f

public / private

③ Doguskenler →  
Create empty → Doguskenler → Add  
component → doguskenler  
News Script

class doguskenler

int soyu = 100;

float ortaYilSoyu = 99.0f;

string yazi = "Unity'de kodlara";

bool doguskenler = true;

int sayi → public dosya  
loans  
Debug.log(sayi); inspector  
parameters  
bir program  
yapabilmesi  
yapabilecegiler.

```
start()
{
    Debug.log(sayi);
    Debug.log(sayi + doguskenler);
    Debug.log(yazi);
    Debug.log(doguskenler);
```

## operator

2D → spectator → create empty → inspectors operator → Neudruckt

Methode für      mit sayf=5;      start();  
 $t = * / ?$       mit sayf=10;      { int source = sayf + sayf2;  
Methode }      start();      DebugLog(source);  
 82 11 !      { if (sayf1 <= 80 && sayf2 >= 10) }  
 - we use begin      { DebugLog("usayf1 sayf2");  
Atom      \* @ kachter verne von bevor estze

## Atom

=    +    -    \*=    /=  
~~int~~    ~~int~~    ~~int~~    composite    BSL operator  
~~int~~    ~~int~~    ~~int~~

mit sayf=50;      { start();      }      result → 55  
 { sayf1 += 5;      }  
 debugLog(sayf1);  
 @ kachter verne von erweiternde kettentur

## Komparation

== , !=, <, >, <=, >=

mit sayf1=50;      start();  
 mit sayf2=10;      { if (sayf1 > sayf2)  
 { debugLog("usayf1 sayf2 den log"); } }

## String Erstellen

mit sayf1=50  
 String year1 = "methoden sayf1";  
 String year2 = "drei";

start();  
 { debugLog(year1 + sayf1 + );  
 year2 }

## Kontroll

IF (kond){ }  
 Else{ }  
 3

Elseif (sayf == 4){ }  
 { }  
 If (3){ }  
 { }  
 3

anuya elmerel

## Kontroll

public int sayf;

start();

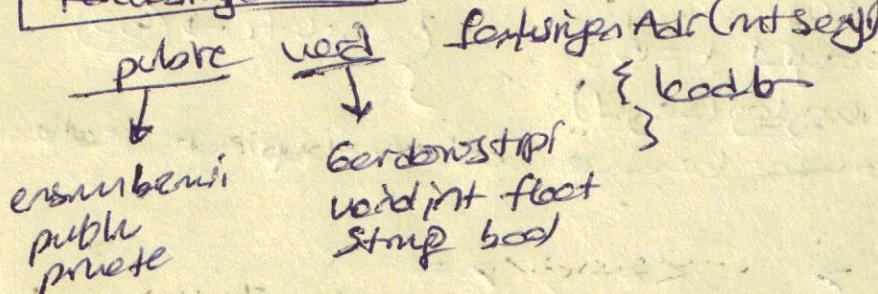
{ if (sayf == 5)  
 { debugLog("usayf 5'cess"); }  
 else { if (sayf < 5)  
 { debugLog("sayf less than 5"); }  
 else { if (sayf > 5)  
 { debugLog("sayf greater than 5"); } } } }

@ oyun yerinden sile sil kullen cogunlar yopl

switch (depot) }  
 { case depot:  
     { break;  
     default:  
         { break;  
     } break;

public String BM() }  
 {  
     switch (BM) }  
     { case "AV": DebugLog("AV");  
         break;  
     case "letz": DebugLog("letz");  
         break;  
     case "Almet": DebugLog("Almet");  
         break;  
     default:  
         DebugLog("Binnenzubereich");

### Fallstruktur



① Depot und mehrere Fallstrukturen

public void Fallstruktur()  
 {  
     DebugLog("Von gestern");

② Depot und dannen Fallstruktur

public float Fallstruktur()  
 {  
     return 4.5f;

③ Parameter oder Fallstruktur

public void Fallstruktur (int parameter)  
 {  
     if (parameter == 0)  
     {  
         DebugLog("Vor gestern");  
         DebugLog("Vor Vierzigster");  
     }  
     else if (parameter == 1)  
     {  
         DebugLog("Vor gestern");  
         DebugLog("Vor Vierzigster");

201 →

④ Vor gestern (String groß geschrieben)  
 {  
     DebugLog("Großschreibung");

⑤ Vor Vierzigster (Großschreibung)  
 {  
     DebugLog("Großschreibung");

start()

# 1 Donyler

while

while(result)  
{  
}

for

for(bastlang; "bastl")  
{  
 cars;  
}

foreach

foreach(depoten,  
 m.LBje-  
 drer)

Exit roken = 1;

Start()  
{  
 while(roken <= 10)  
 {  
 debuglog("Rokant" + roken);  
 roken++;  
 }  
}

Start()

{  
 for(m+a=0; a<=6; a++)  
 {  
 debuglog(a);  
 }  
}

foreach(string am  
Bruder)

{  
 debuglog(a);  
}

String Bruder = new String("A");

Start()  
{  
 Bruder[0] = "A";  
 Bruder[1] = "L";  
 Bruder[2] = "U";  
 Bruder[3] = "D";  
}

Debuglog(Bruder[0]);

for(m+a=0; a < Bruder.length; a++)

{  
 debuglog("Bruder[" + a + "]);  
}

③ `String getName()`  
 {  
 return "Markus Mayr";  
}  
`start()`  
{  
~~DebugLog("HelloWorld")~~;  
}

④ Konsolenerie  
 Konsolenerie  
 Konsolenerie  
 can eksriyer  


```
int con = 100;
void canReset(int doper)
{
    can -= doper;
}
int canReset()
{
    return con - 10;
}
};
```

`start()`  
{  
~~DebugLog("CanReset")~~,  
conReset(10);  
~~DebugLog("gerenReset")~~,  
~~DebugLog(canReset)~~;

### Dizler Senler

```
String[] Bilder = new String[3];
Bilder[0] = "A";
Bilder[1] = "B";
Bilder[2] = "C";
DebugLog(Bilder[0]);

```

⑤ oyun ekstra dosya verilebilir  
 dosyaların deposunu ve döner  
 ekranını quekliyor

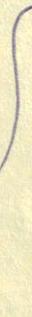
```
String[] Bilder = new String[3];
String[] { "A", "B", "C" };
DebugLog(Bilder[0]);
```

`DebugLog("Bildizide"+Bilder.length+ "tane ekran var")`

List<String> meyeler;

```
start()
{
    meyeler = new List<String>();
    meyeler.add("çilek");
}
```

```
DebugLog(meyeler[0]);
DebugLog(meyeler.Count);
```

  
meyeler.clear();  
meyeler.Add("muz");  
meyeler.RemoveAt(0);

(2)

## # Animasyon Ehleme #

plane → position → reset

cube → " → reset → Position.  $x:0$   $y:0,5$   $z:0$ cube se ~~se~~ yeri → window → Animation →↳ ~~cube se yeri~~ → Assets altına → Folder → Animations →Create file. \* Assets altına "isim verile.

" Benim Animasyonum" isim verile.

Olusturdulduğun sonucunda de cube'ın controller'i, olusturduğun sonucunda de cube'ın controller'i, olusturduğun sonucunda de cube'ın controller'i, olusturduğun sonucunda de cube'ın controller'i,

cube'ı see → Animator basılı altında ✘

(\* cube'ı hacket verenle isterseniz ;)

(\*) Add property → Transform → Position → +

Cube position daye lasteye yea bor sotır ekler.

① O'dan 1'e circa 5 sonuncu birde. position:  $x:0$   $y:4,5$   $z:0$ ② 1'deyken ekler  $0,5$  le,  $0,38$  deyken de 1'e ayterseniz,

sonuncu yubes asagıya aynasızca ekler.

transform

rotation +

(\*) Add property → Rotation  $x:0$   $y:360$   $z:0$ ① O'dan  $\frac{1}{2}$ , sn'ye geldiğinde Rotation  $x:0$   $y:360$   $z:0$ 1 seplasyona  $360^{\circ}$  devrede tur bor dısına gerekleşir

transform

(\*) Add property → Scale +① O'deyken Scale  $x:0$   $y:1$   $z:0$  Reer0,30 sn'de  $x:1$   $y:0,7$   $z:1$ 

yukarıya akturus sikişir bit kop oluyor.

0,25n'yeje Add frame →  $y:1$ 0,50 n'yeje de ekle →  $y:1$

~~Han perde konstant~~

(1)

~~plane  
cube~~

## GETAXIS KONTROL

Nous Topkontrol

(1) plane  $\rightarrow$  zoom  $\rightarrow$  scale : x:11 y:1 z:11

(1) sphere  $\rightarrow$  top  $\rightarrow$

blockade cube  $\rightarrow$  cockpit view from  $\text{ctrl}+\text{D}$  kollarlobibitir.

top run bar script olustur  $\rightarrow$  Topkontrol Scripts  $\rightarrow$  Top in Game  
Scorekobirk -  
top hikmet ettirobbukti rem br Kollarlobitir  
top hikmet ettirobbukti rem br Kollarlobitir

Top  $\rightarrow$  Add Component  $\rightarrow$  Rigidbody  $\rightarrow$  Sofit renzregel.

class Topkontrol :

{ Rigidbody rb;  
 $\text{float Hiz} = 2.5f;$

void Start()

{ rb = Get Component < Rigidbody >();

float  
sag  
sol  
yol  
otay

Update() (kuvette uygulayacak)

③ rb.AddForce (transform.forward \* 3);

$\downarrow$  float yotay = Input.GetAxis("Horizontal");

$\downarrow$  float dikey = Input.GetAxis("Vertical");

\* Edit  $\rightarrow$  project Settings  $\rightarrow$  Input Manager  $\rightarrow$  Axes  $\rightarrow$

Horizontal  $\rightarrow$  Left Right  $\rightarrow$  aild fusterl.

vertical  $\rightarrow$

④ kuvette hizup topa uygulayacak  $\rightarrow$  x, y, z  
kuvette = new Vector3 (yotay, 0, dikey);

(yotay, 0, dikey)

$\begin{matrix} \leftarrow & \uparrow & \rightarrow \\ x & y & z \\ \downarrow & \downarrow & \downarrow \\ \text{yotay} & \text{yukari} & \text{per} \\ & \text{c. sap} & \text{ceri} \end{matrix}$

⑤ rb.AddForce (kuvette \* hiz);

⑥ kuvette hizup acisi rm  $\rightarrow$  normlenu  $\rightarrow$  y ne

yukenkobir  $\rightarrow$  x'e tikla  $\rightarrow$  ⑦ brez asefylemek