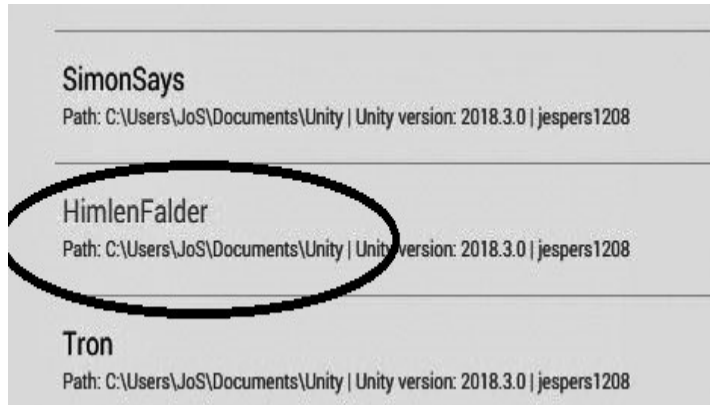


Himlen Falder 2.!

Først åbner du det Gamle projekt “Himlen Falder”.

Nu skal vi opdatere projektet, og tilføje nogle seje ting til dit fly..

Den lille firkant som du styre, er et fly eller rumskib.
Den ting som falder fra himlen er enten bomber eller fjende fly..



For at gøre masser af plads for dit fly, så skal vi ændre lidt på opsætningen.

Tryk på din “Player” og på “Game”, så kan du se hvordan skærmen ser ud, når du har startet spillet.

Hvis du ændre på Y positionen på “Player”, så kan du flytte flyet længere ned. Hos mig er -10 den rigtigt, så flyt din “Player” ned lige over bunden.

Tryk på Play og se om du har bedre tid, når dit fly er længere ned på skærmen.

Så skal vi ændre baggrund til at være en sort baggrund.

Så det ligner at du flyver ude i rummet.

Først skal du trykke på “Main Camera”, over din “Player” til Venstre.

Så skal du ændre “Skybox” til “Solid Color”.(1)

Og så skal du finde en baggrunds farve ved at trykke på den blå firkant.(2)

Jeg vælger sort, som rummet.



Tryk på “Play” og test om det ser godt ud.

Skyde Funktion:

Nu skal vi have vores Fly, til at skyde.

Dobbelt klik på “PlayerControl”.

Først skal du lave en variable, som skal indeholde vores “Skud”.

Under public class og over Start, skal du tilføje en variable.

```
public Rigidbody skud;
```

Denne Variable har en Rigidbody, som giver vores “skud” nogle fysiske evner.

Så skal du skabe skudet, og det skal du gøre under, der hvor du styre dit fly.

```
if (Input.GetKeyUp(KeyCode.Space))
{
    var affyret = Instantiate(skud, transform.position, transform.rotation);
    affyret.velocity = transform.forward * 10;
}
```

Denne funktion gøre først, at du venter på om der bliver trykke på Mellemrum. (Mellemrum er Space på engelsk.)

Når der bliver trykke på “Space”, så skaber den et objekt, som kaldes “Affyret”. Dette objekt får så en retning (Transform.forward) og en hastighed (* 10).

Gem og gå tilbage til Unity.

Nu skal du skabe din kugle.

Jeg laver den til et Laserskud, men du kan opfinde den slags skud du vil.

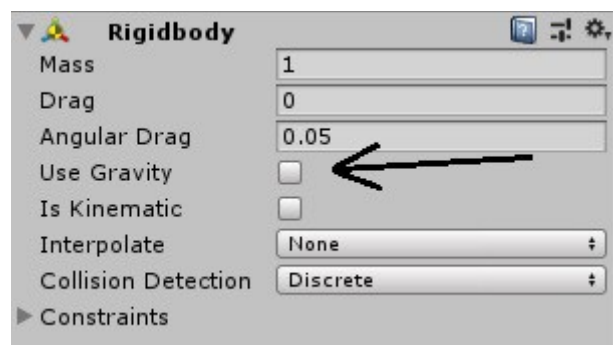
Først skal du op i venstre hjørne, og trykke på Create -> 3D -> Capsule.

Den skaber et objekt, som ligner et laserskud.

Først skal du ændre navnet til “Skud”.

Så skal vi ændre farven, som vi har gjort før...

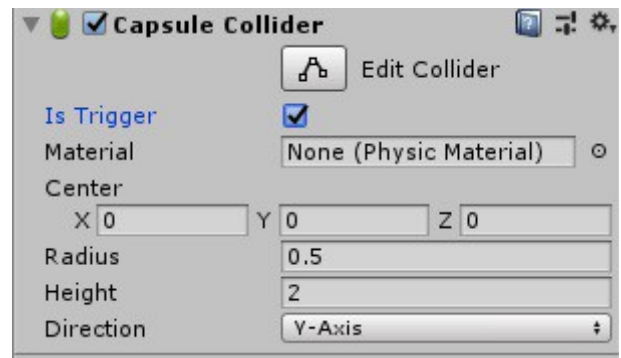
Og tilføje et Rigidbody, men Fjern Tyngdekraften.



Så skal du ændre dens krop til en trigger, i stedet for en collider.

Til sidst skal du trække den ned i “Assets”, og slet det i oppe til venstre.

Nu har du et skud, som er klar til at blive loaded i din players Våben holder.



Tryk på din “Player” til venstre, og kig på det skrib som sidder på den.

Der burde være en tomt kasse med navnet “None (Rigidbody)”

Træk det skud, som ligger i “Assets”, ind i den kasse.

Så det ligner dette billede.



Hvis der ikke er nogle fejl, så tryk på “Play”, og skud dit våben et par gange.

Mit skud går ind i skærmen, i stedet for opad, så jeg skal ændre i midt skript.

```
affyret.velocity = transform.forward * 10;
```

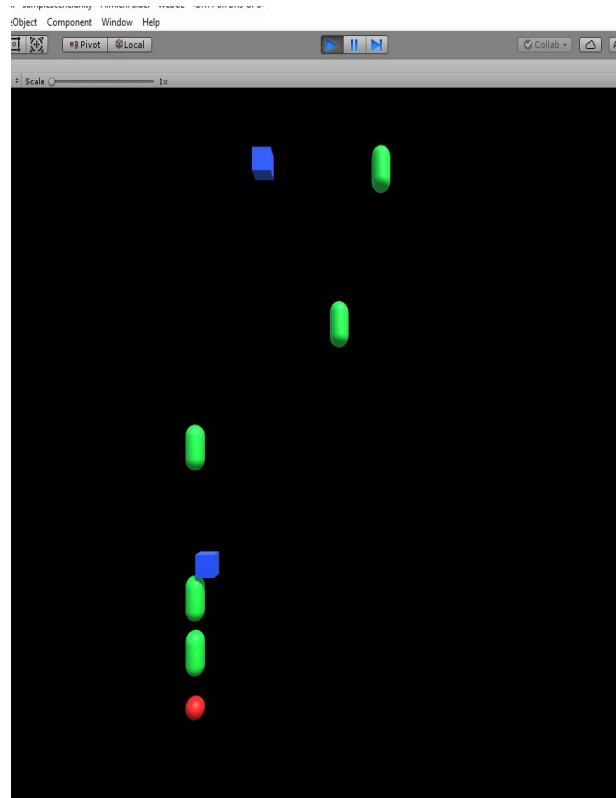
Retningen på skudet bliver bestemt af Transform.forward.

Der er 6 retninger... forward, right og up, og så Minus de retninger.

Eks. `-transform.forward * 10;`
Skyder i modsatte retning, fordi det er “Minus forward”.

Find den retning, som får dit skib til at skyde op ad.

Her er mit skærm billede...



Der sker intet, når vi rammer de bokse som falder fra himlen.
Fordi de bokse bare er tomme ting, som ikke ved hvad de skal gøre, når de bliver ramt.

Nu laver vi et skript, som skal sætte på boksene, så de dør, når de bliver ramt.

Peg musen ned på “Assets”, og skab et nyt skript.
(Højre klik - > Create – C# Script)

Kald det “DødsDansen”, fordi det lyder sejt, og dobbelt klik på det.

Den skal kun have to linjer, som skal placeres mellem Start og Update.

```
private void OnTriggerEnter(Collider other)
{
    Destroy(gameObject, 0);
}
```

Da vi kun har en Trigger, så kan vi bare sige..
Hvis dette skript rammer en “Trigger”, så skal den Død.

Gem skriptet, og gå tilbage til Unity.
Tryk på Play og se om du kan skyde boksene ud af himlen.

Hvis du også tilføjet at du kan kontrollere “Player”, med musen,
så skal du vel også skyde med musen...
(Det var Master level del 1)

I dit “PlayerControl” skript, så skal du ændre den linje med “Space”.

```
if (Input.GetKeyUp(KeyCode.Space) || Input.GetKeyUp(KeyCode.Mouse0))
```

Dette betyder at du enten kan trykke på “Space” eller på “Mouse0”.
Det er venstre knappen på musen, som er “Mouse0”.

De to vertikale linjer, sidder yderst til højre på tastaturet.
Hos mig skal jeg trykke på “Alt Gr” + “ ’ “
Som er knappen ved siden af “ + “.

Master Level 1: Tælle point.

Vi skal tælle hvor mange fjender, som vi har skud.
Og så skal det skrives på skærmen.

Først skal vi tælle hver gang en fjende dør, og det kræver to ting.

Først en variable, som husker hvor mange er død,
også skal vi sende data til den variable.

Så åben dit “PlayerControl”, og tilføj denne variable, der hvor du satte din skud variable. Se øverst side 2.

```
public static int død;
```

Denne variable er lidt anderledes end de andre, fordi den er en “Public Static”, som betyder at der kan ses og skrives til af andre.

Så skal du åbne det andet skript, som jeg kaldte “DødsDansen”.
Lige under der hvor du skrev Destroy, så skal vi sætte en ny linje.

```
PlayerControl.død++;
```

Denne linje ligger et tal til død, hver gang en fjende bliver dræbt.

Nu skal vi så skrive det tal på skærmen, med denne linje fra “Maze.”
Denne funktion, skal du tilføje i “PlayerControl” mellem start og Update.

```
private void OnGUI()  
{  
    GUI.Label(new Rect(1, 1, 100, 100), "Fjender døde: " + død);  
}
```

Hvis du vil have en forklaring, så skal du kigge i “Maze”

Men den korte version er...

Den skriver en linje i øverste venstre hjørne...med hvor mange du har Dræbt.

Master Level 2: Progressive Sværhedsgrad...

Eller sagt på en nemmere måde.
Jo flere du dræber, jo hurtigere skabes de næste.

Vi har en tæller, som holder styr på hvor mange som er død.
Den værdi, kan vi bruge til at trække fra hvor mange mikro-sekunder der går imellem at den laver en ny Fjende.

Går ind i dit “FjendeControl”, og find denne linje.

```
affyringsTid = Time.time + venteTid;
```

Den ligger venteTid, (som er 1 sekund), til hvad klokken er lige nu.
Altså vent 1 sekund med at affyre næste fjende.

Hvis vi trækker antallet af død fra... så bliver de affyret hurtigere.
Men vi kan ikke bare trække antel af død fra, ellers bliver værdien lav.

```
affyringsTid = Time.time + venteTid - (PlayerControl.død/100);
```

Nu bliver spillet svære og svære,
jo flere du dræber.

Hvis du sætter tallet fra 100, ned
til 10, så bliver det meget svært.
Prøv det.

Ændre størrelsen, på dit skud og
på fjenderne.

Så det bliver svære at ramme
fjenderne, som falder.

Gå ned i “Assets”, og tryk på fjende
og ændre scalen til noget mindre.

Gem og tryk på “Play”.

