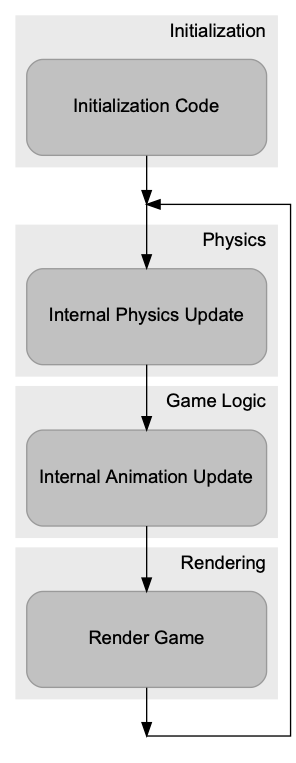
## Game loop in vogelvlucht

Alle code in Unity wordt in een specifieke volgorde uitgevoerd. Een vereenvoudigde weergave van deze 'Game Loop' is hieronder weergegeven.

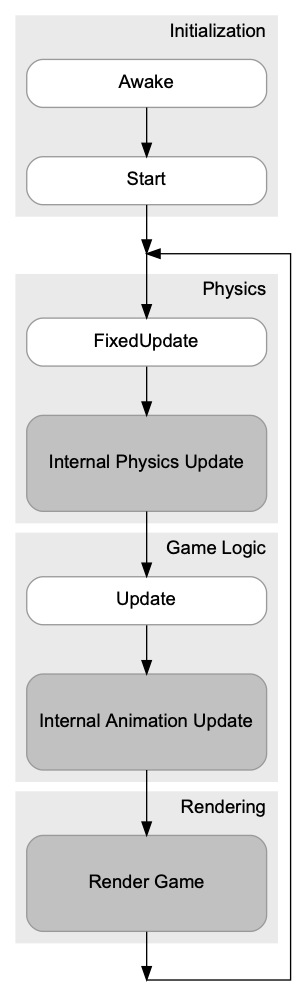


<<Images/game\_loop\_overview.png>>

* Een iteratie door de game loop wordt ook vaak een **frame** genoemd.
* **Render game** is een intern Unity proces waarbij de 3D wereld wordt vertaald naar pixels op een scherm (en dat is 2D). Pas na deze stap zijn alle veranderingen zichtbaar voor de speler.

## Relatie met Unity Scripts

Hieronder staan de functies die in de video tutorial worden gebruikt op de juiste plek in de game loop



<<Images/game\_loop\_with\_functions\_from\_video.png>>

Opmerkingen:

* De functies die door Unity worden uitgevoerd staan in het wit in het diagram en worden vaak **event functies** genoemd.
* Zoals te zien worden Awake en Start maar één keer uitgevoerd. Awake vindt plaats op frame 0, frame nummer 1 begint bij start en in dit geval beginnen frame 2 en alle volgende frames bij FixedUpdate

### Volgorde bij meerdere scripts

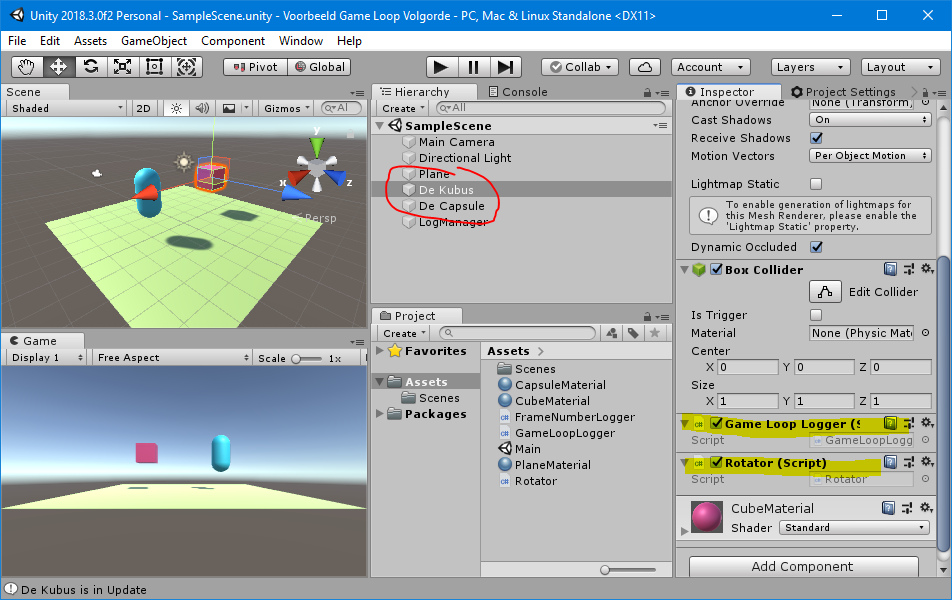
Unity voert per script en per game object alleen de event functie uit die op dat moment aan de beurt is en gaat daarna door naar een volgend script, of volgend game object. Pas nadat Unity de huidige event functie bij alle scripts uit alle game objecten heeft uitgevoerd, gaat Unity verdere naar de volgende event functie

Het is dus *niet* zo Unity per script alle event functies uitvoert voordat het doorgaat met het volgende script.

De volgorde waarin Unity de game objecten langs gaat, kun je beïnvloeden als programmeur, maar dit leidt snel tot onoverzichtelijke code. Daarom kun je er beter vanuit gaan dat je de volgorde niet kent.

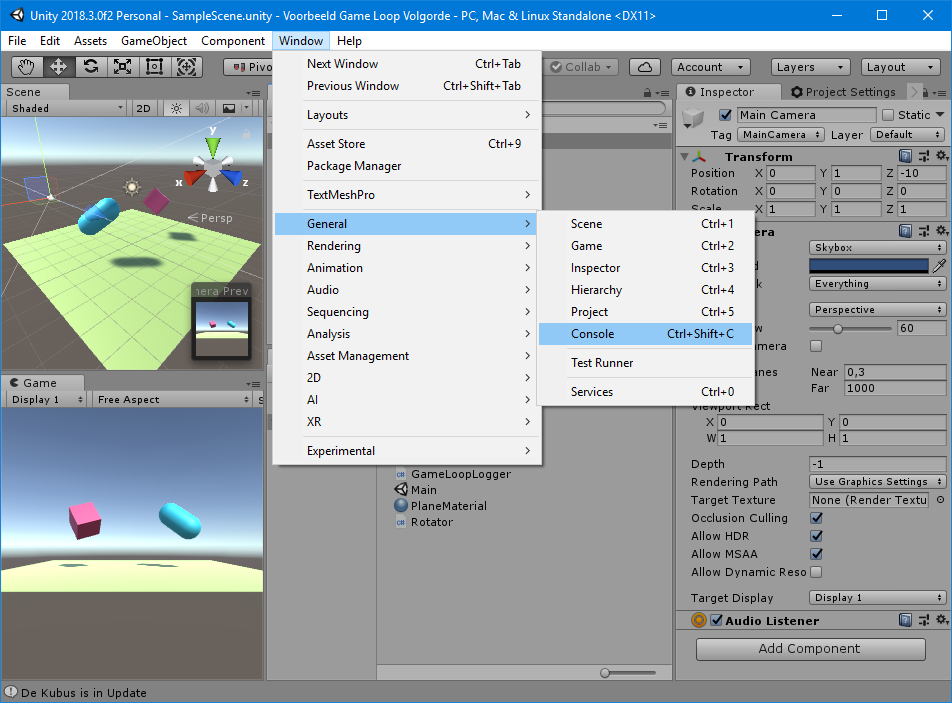
## Voorbeeld in Unity

In dit voorbeeld zijn er twee Game Objecten (De Kubus en De Capsule) die elk twee Scripts hebben: een script om te vertellen welke event functie wordt uitgevoerd (GameLoopLogger.cs) en een script om het object te roteren (Rotator.cs).



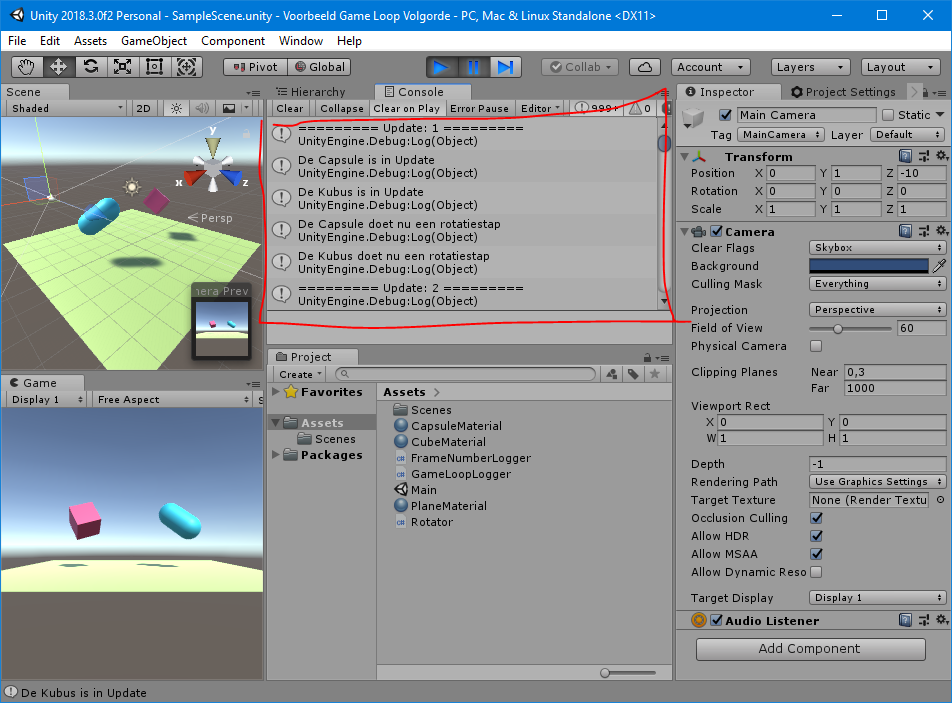
<<Images/project\_overview.png>>

Zorg dat het console window open is door (in Unity 2018.3) te klikken op Window > General > Console zoals hieronder te zien is:



<<Images/console\_window\_openen.png>>.

Wanneer je het project uitvoert zou je het volgende op de console moeten zien:



<<Images/console\_output.png>>

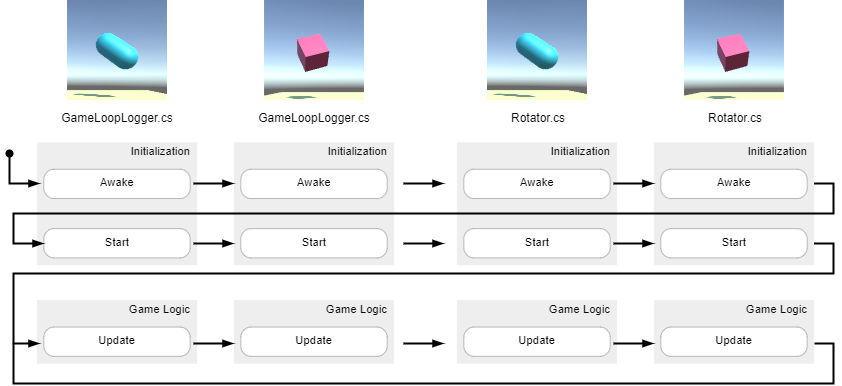
In de console zie je op elk moment uit de game loop welke event functie van welk game object wordt uitgevoerd. Het nummer achter de naam van de event functie is het volgnummer van het huidige frame.

Hieronder is de code van GameLoopLogger.cs te zien

| public class GameLoopLogger : MonoBehaviour {  void Awake()  {  Debug.Log(name + " is in Awake");  }   void Start()  {  Debug.Log(name + " is in Start");  }   void Update()  {  Debug.Log(name + " is in Update");  } } |
| --- |

Debug.Log schrijft een tekst naar het console window en de eigenschap name is de naam van het Game Object dat rechtsboven het Inspector window van dat Game Object te zien is.`Debug.Log` is onmisbaar voor een Unity programmeur.

De complete volgorde is hieronder te zien

****

<<Images/example\_order.png>>