

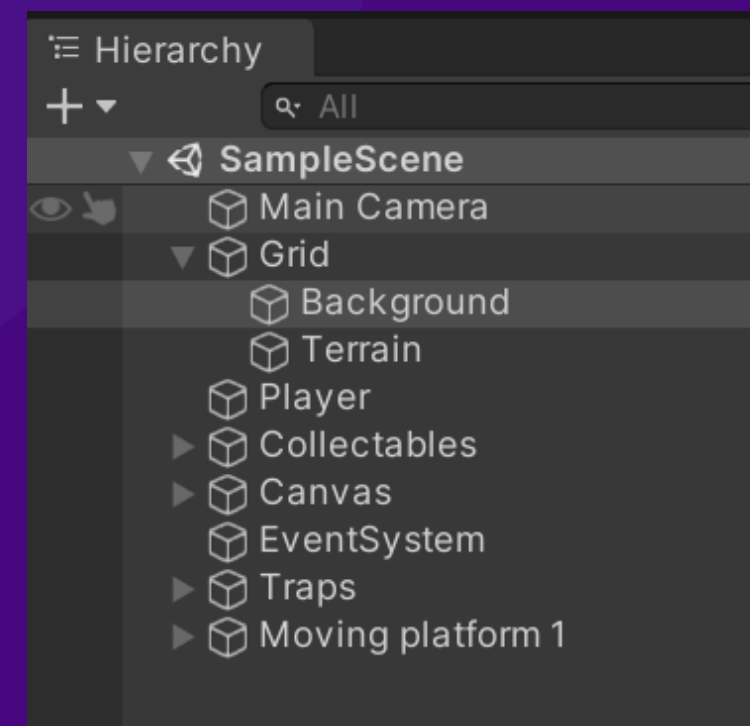


10 THINGS TO KNOW ABOUT UNITY

1. WHAT IS SCRIPTING IN UNITY?

Concept of coding in Unity

- ✦ Scripting tells GameObjects how to behave.
- ✦ In Unity, you don't need to create the code that runs the application. The Unity does it for you.
- ✦ The Unity runs in a big loop, reading all of the data that's in the game scene. It runs single discrete frames, one after the another.



2. LANGUAGE THAT CAN BE USED IN UNITY

C # (C Sharp)

C/C++

Rust

Object-Oriented Language





SCRIPTING LANGUAGE

Unity is compatible with Object-Oriented Language that is used to run the GameObject.



VARIABLES

Variables holds values and references to objects. They are like a box that holds something for us to use. Follows the camelCase rules.



FUNCTIONS

Functions are collection of code that compare and manipulate these variables. Function starts with an Uppercase letter.



CLASSES

Classes are a way to structure code to wrap collections of variables and functions together to create a template that defines the properties of an object.

3. VARIABLES

Declaring the variables

- ✦ Visibility Types: public and private
- ✦ When you make variables public, they're accessible to other scripts/classes, and can be changed in the inspector from the Unity Editor. So, we should make it private!
- ✦ Private variables allow your code to be cleaner, since the value of those variables can only be changed from inside that class. This makes debugging and maintaining the code easier.



VARIABLES

Define what kinds of value is the variable

- ✦ Types: number, text, or complex types.
- ✦ Naming Convention: variable can not start with number and can not contain any space in between
- ✦ Serialization: Using [SerializeField] in the Script allows Unity to store your data/objects for reconstruction. It allows feature like saving and loading.



4. FUNCTIONS

Basics of Functions

- ✦ Default Functions: Awake, Start, Update, FixedUpdate, LateUpdate.
- ✦ Start is called if a GameObject is active, but only if the component is enabled.
- ✦ Update is called once per frame. This is where you put code to define the logic that runs continuously, like Animations, AI, and other parts of the game that have to be constantly updated.
- ✦ Functions can do calculations and return a value. There are void type just as like an Object-Oriented Language.



5. CLASSES



CLASSES

CLASSES ARE COLLECTIONS OF THE VARIABLES AND FUNCTIONS.

Class name must match the file name of the C# script for it to work. Otherwise, it will not be accepted in the Unity and won't compile.



UNITY SPECIALTY- SERIALIZATION

It will be converted into simple data that Unity can look at in the inspector. It will see that you have the class will appear in the inspector.



“

**WE DECIDED TO SCRAP OUR
INDIVIDUAL ENGINES AND
MAKE AN ENGINE TOGETHER,
BECAUSE IT'S MORE FUN WHEN
THERE'S TWO OF YOU WROKING
ON SOMETHING.**

David Helgason



6. HISTORY OF UNITY

MAY 21, 2002

Nicholas Francis, a Danish programmer, posted on the Mac OpenGL board asking for assistance with a shader system he was trying to implement into his game engine.

FEW HOURS LATER

Joachim Ante, who lived in Germany at the time, responded to Nicholas' post. Their conversations resulted in the two developers collaborating to create a shader system that would work for both of their separate engines.

LATER

David Helgason heard about the project and thought they were "really" onto something, so he jumped aboard as the third developer.



“

AT FIRST THEY WANTED TO DEVELOP GAMES FOR A LIVING, BUT SAW THE NEED FOR A BETTER UNDERLINING TECHNOLOGY. THE THREE ENDED UP NOT MAKING THE GAMES, BUT RATHER MAKING A TOOL TO MAKE GAMES.



7. WHO AND WHERE IS UNITY BEING USED?

Usage of Unity

- ✦ Unity is a 2D/3D engine and framework that gives you a system for designing game or app scenes for 2D, 2.5D and 3D.
- ✦ Therefore, it was used in a lot of 3D games and 2D games. i.e. Pokemon Go, Hearthstone, Rimworld, Cuphead, and more.

8. HOW IS UNITY BEING USED?

Usage of Unity

- ✦ You go to Asset Store to download the initial set of Game Objects. That includes "Player", "Obstacle/Trap", "background", "Tiles", and "Animation Effect".
- ✦ Since the Unity is designed to build the 2D and 3D game, it's easier to switch between 2D game to 3D game, or 3D games to 2D games.



9. ASSET STORE



Explore the Item Store in Unity

ASSET STORE HAS A LOT OF PACKAGES AND ITEMS.

1. EASY ASSESS
2. FREE STUFF
3. UPDATED EVERYDAY
4. MANY USERS
5. VARIETY



CLICK IN THE LINK BELOW

2D Environments & Characters |
Unity Asset Store



10. HOW TO PLAY OUR GAMES?

Tutorial on our Game.



OUR OPINION ON UNITY

1. Did you enjoy learning/using the language? Why?
2. Do you think the language is useful? Why?
3. Compare the language with Java / C / Python
4. Would you recommend the language?



FURTHUR STUDY

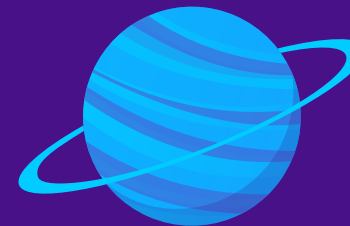
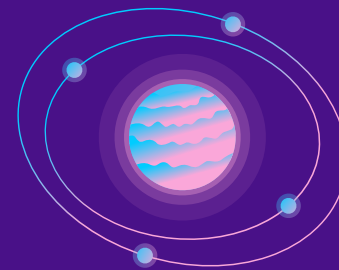
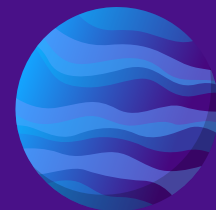
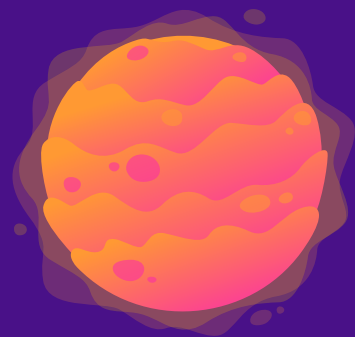
What are some aspects of the language you would be interested in investigating furthur?

REFERENCES

1. <https://unity.com/how-to/learning-c-sharp-unity-beginners>
2. <https://digital.wpi.edu/downloads/2f75r821k>
3. <https://unity.com>
4. <https://assetstore.unity.com/2d>
5. <https://docs.microsoft.com/en-us/archive/msdn-magazine/2014/august/unity-developing-your-first-game-with-unity-and-csharp#:~:text=Unity%20allows%20you%20to%20import,animation%20system%2C%20and%20much%20more.>

RESOURCE REFERENCED

Below planets are the resources (links are placed in the planets) we used to prepare our presentation.





GAME
ON