

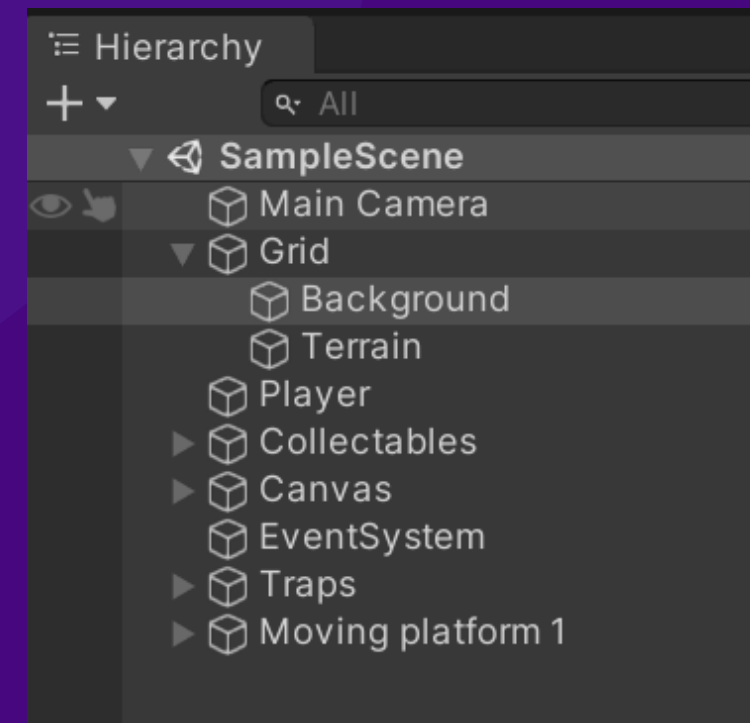


10 THINGS TO KNOW ABOUT UNITY

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 USED IN UNITY

C # (C Sharp)
Object-Oriented Language



A large, stylized letter 'S' in orange with a pink outline, positioned at the top of the first column.

SCRIPTING LANGUAGE

Unity can understand
Object-Oriented
Language to run the
GameObject.

A large, stylized letter 'V' in pink with a blue outline, positioned at the top of the second column.

VARIABLES

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

A large, stylized letter 'F' in blue with a pink outline, positioned at the top of the third column.

FUNCTIONS

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

A large, stylized letter 'C' in yellow with a blue outline, positioned at the top of the fourth column.

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 it public, it's accessible to other scripts and other 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 you know that value of those variables can be changed only 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.
- ✦ Name: naming variable
 - Cannot start with number
 - cannot contain spaces
 - camelCase
- ✦ Using [SerializeField] in the Script makes it visibel in the Unity to change the values of the variables.



4. FUNCTIONS

Basics of Functions

- ✦ Default: 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 compile, and not accepted in the Unity.



UNITY SPECIALTY- SERIALIZATION

It will 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, Healthstone, 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?

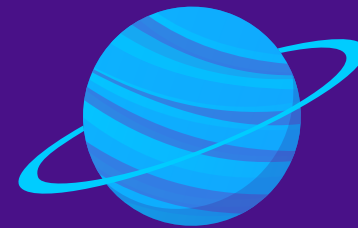
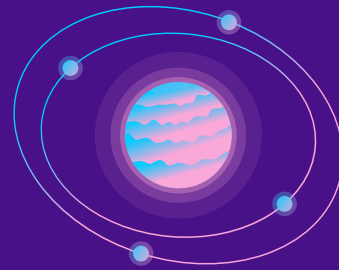
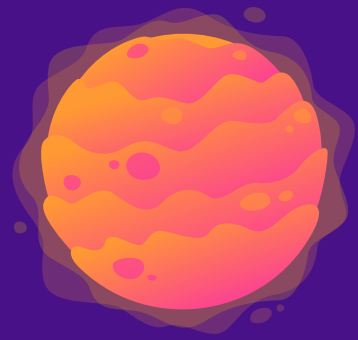


FURTHUR STUDY

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

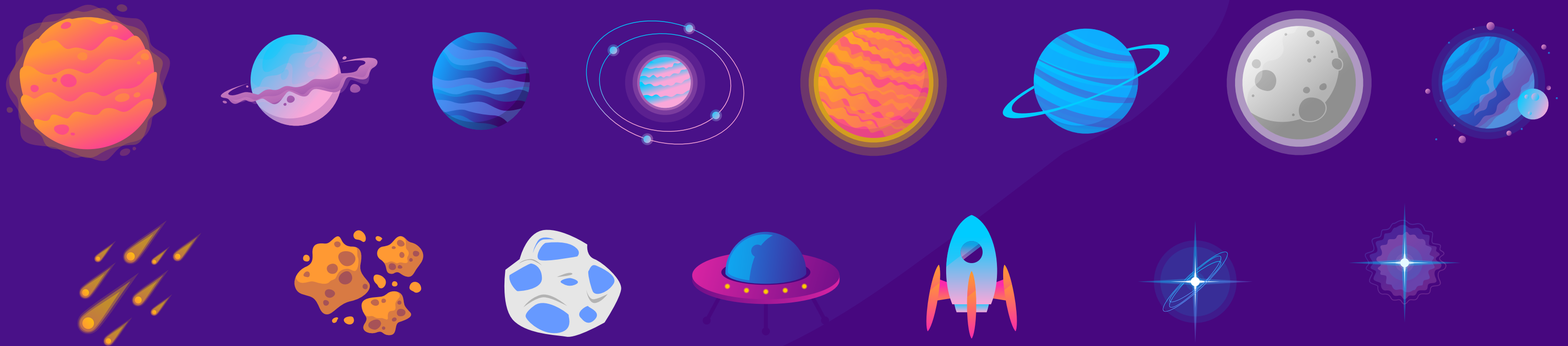
RESOURCE REFERENCED

Below planets are the resource(Link) we used to prepare our presentation.



RESOURCE PAGE

Use these icons and illustrations in your Canva Presentation. Happy designing! Don't forget to delete this page before presenting.





GAME
ON