Camera Free Minimap

Camera Free Minimap

Description:

"**Camera Free Minimap**" is a free Unity asset designed to simplify the creation of dynamic and camera-independent minimaps. Say goodbye to the complexities of setting up a second camera or manually updating your minimap. With "Camera Free Minimap," you can effortlessly integrate your minimap images, player icons, and camera FOV, leaving the asset to handle the rest for you.

Key Features:

- **Camera Independence:** No need for a second camera setup, ensuring a lightweight and performance-optimized solution.
- **Image-Based Rendering:** Use custom minimap images and player icons for creative freedom in your game world.
- **Automatic Updates:** The minimap dynamically reflects player movements and camera FOV in real-time gameplay.
- **User-Friendly Setup:** Seamlessly and intuitively set up with minimal configuration for developers of all levels.
- **Customizable Icons:** Tailor player and camera FOV icons to match your game's style and convey crucial information effectively.

Use Cases:

- **Performance Oriented:** "Camera Free Minimap" ensures smooth gameplay performance by removing the overhead of a second camera setup.
- **Seamless Integration:** Easily integrate "Camera Free Minimap" into any Unity project, regardless of the game genre or complexity.

How does it work?

Camera Free Minimap utilizes a GameObject with sprite renderer in the world that should perfectly align with your world's environment. This sprite renderer helps capture the X and Z coordinates, representing the minimum and maximum boundaries of your world. By calculating the world's size and tracking the player's position, Camera Free Minimap effectively maps the player's location on the minimap, so make sure to align the sprite renderer with your environment perfectly as it is crucial.

(The "Minimap Manager" acts as the sprite renderer, although you can choose any other gameObject for this purpose, it's essential to assign its reference to the Minimap Manager.)

IMPORTANT:

- Avoid scaling the minimap or its holder as it can interfere with the calculations, leading to incorrect player icon positions on the minimap, to change the size, you can change the width and the height of the Rect Transform.
- The current version is optimized for Canvas with a Canvas Scaler set to "Scale With Screen Size" and "Constant Pixel Size." It won't function correctly with the "Constant Physical Size" mode.
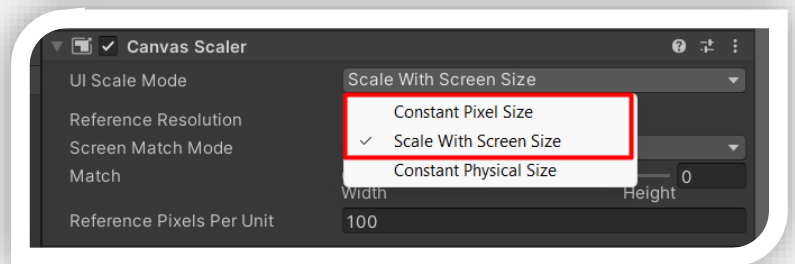
Limitations and Future Plans:

Current Version Limitations:

- The current version of the minimap is static and best suited for dungeons, mazes, RPG maps, and other static environments. It is not designed for games where the camera moves with the player. To enable dynamic movement, modifications to the code are necessary.
- Expanding the elements on the minimap must be done manually, as the present version supports only the player in the "Minimap Manager."

Future Plans:

- Creating a dynamic version of the minimap that moves along with the player, revealing more of the world map as they explore.
- I aim to enhance the minimap's functionality by allowing the addition of more UI elements beyond the player representation.
- I am working on implementing a zoom in and out feature for better map navigation and exploration.
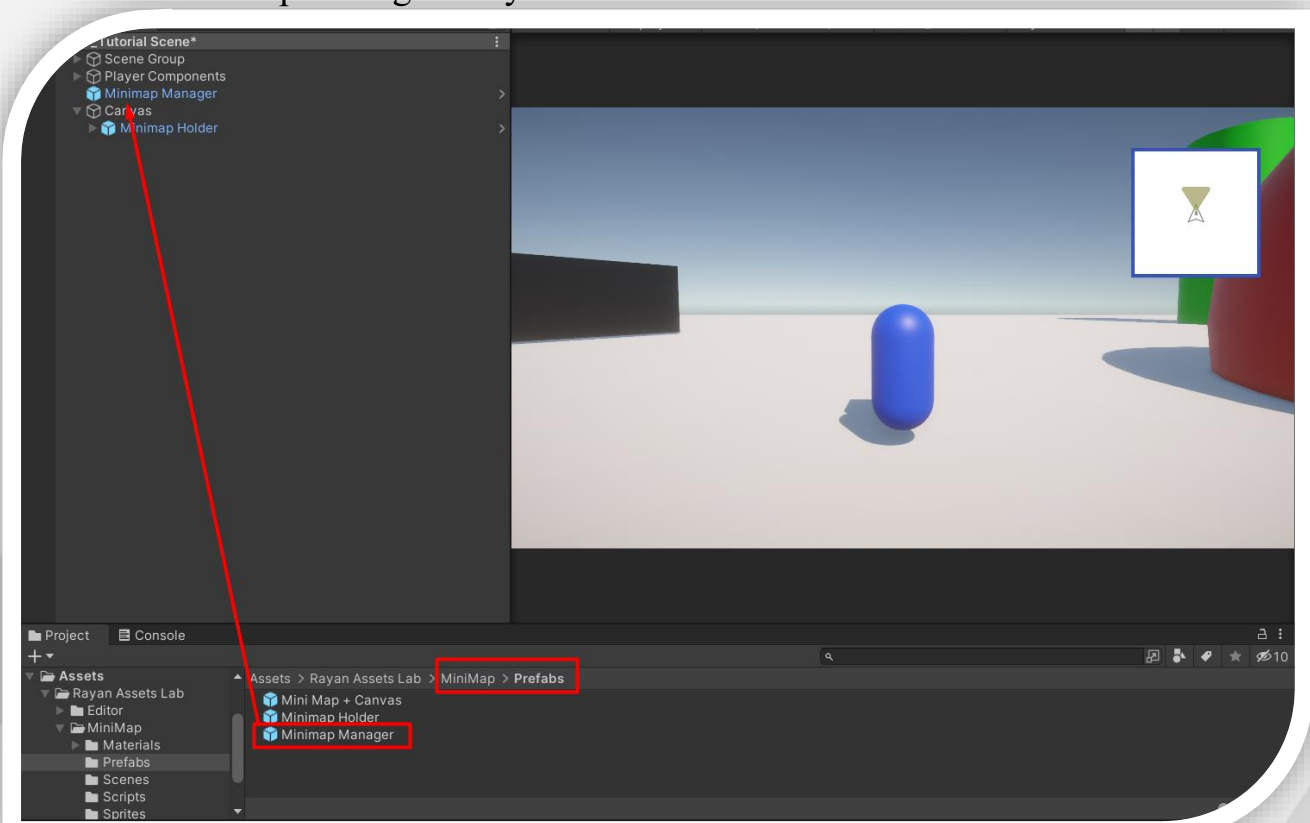
How to use it?

Method (1):

STEP1:

- Open prefabs folder as shown below and add "Minimap" prefab to your **canvas** in the desired scene.
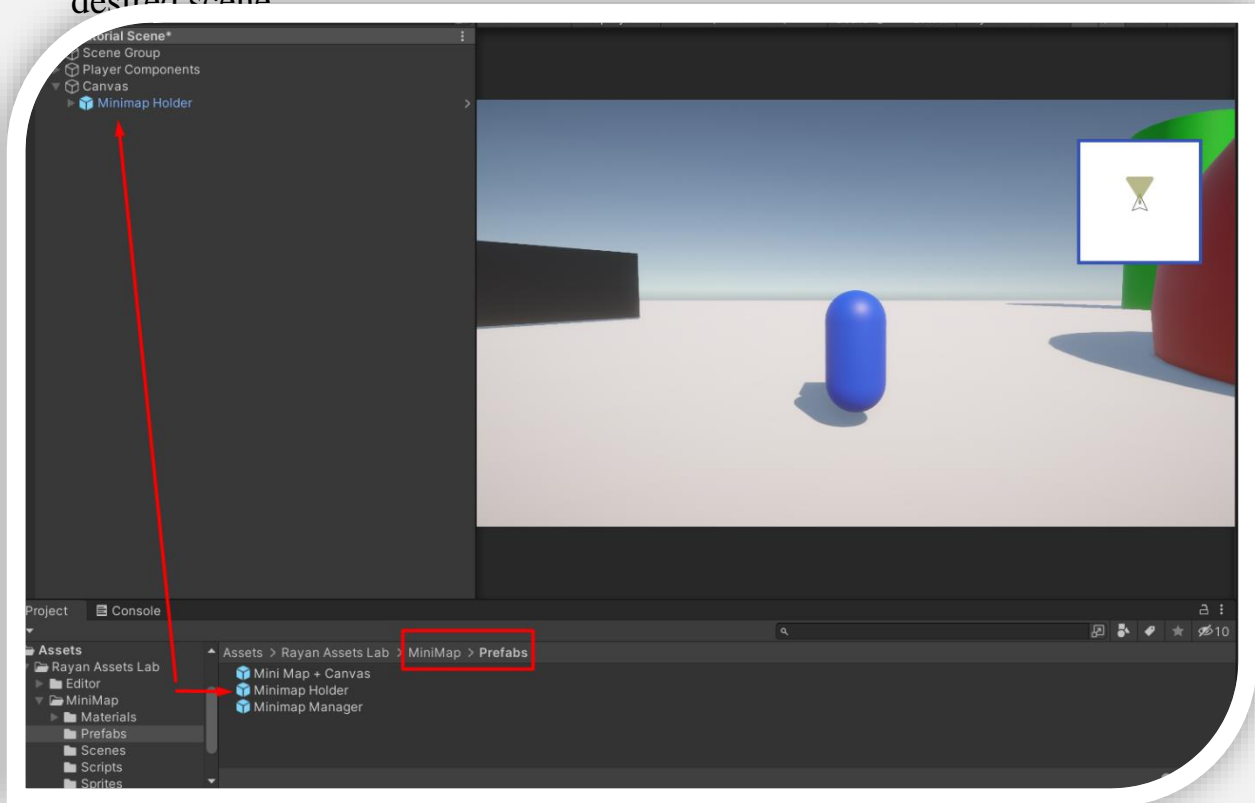
STEP2:

- Add "Minimap Manager" to your scene.

STEP3:

Minimap UI in Canvas:
Reference to your minimap in the UI in
the canvas.
Minimap Sprite Renderer in World:
Reference to the sprite of the minimap image in the world,
If "Minimap Manager" component is on "Minimap Sprite Rendere in World" is a component of the "Minimap Sprite Renderer in World" it will atuomatically take the reference from its components.
if not, you'll have to assign it yourself.
Minimap Sprite:
This is your minimap image.
Player Transform in World:
A reference to your player, this is the part
of the player that moves and rotates in
the world.
Player Icon in Minimap:
Reference to the player's icon on the UI
in the canvas.
Player Camera in World:
Reference to the player's camera. (it is
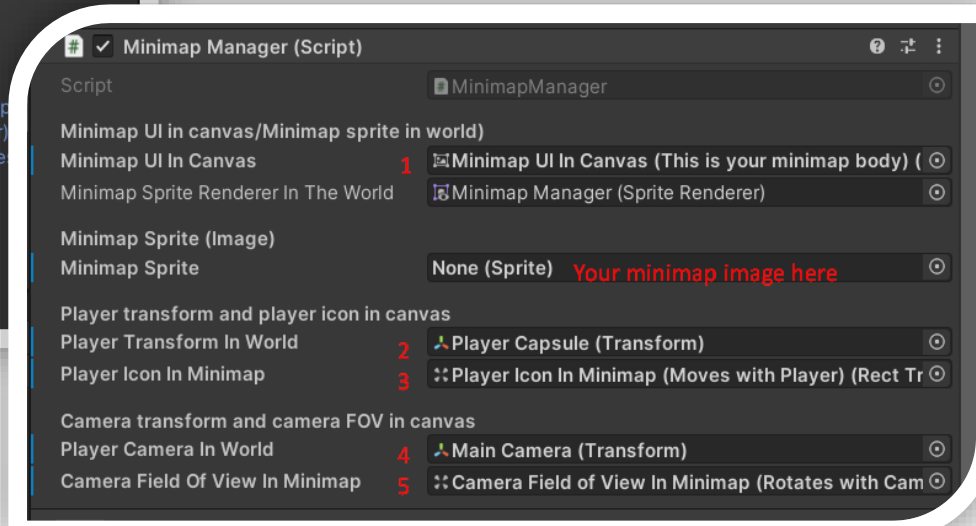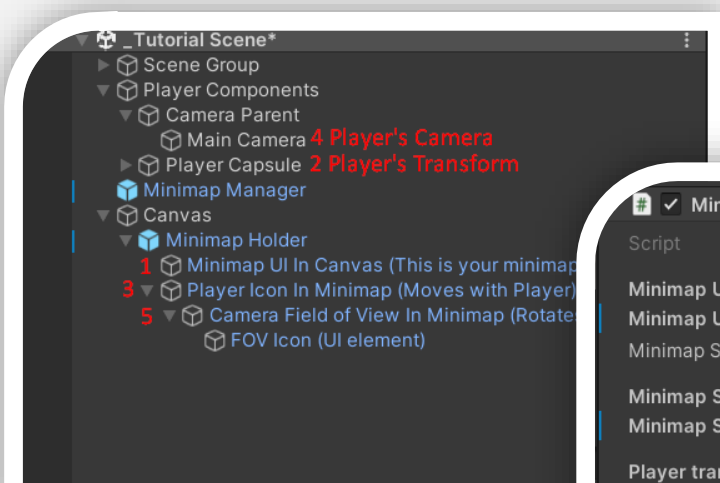the object that has camera component)
Camera Field of Viw in Minimap:
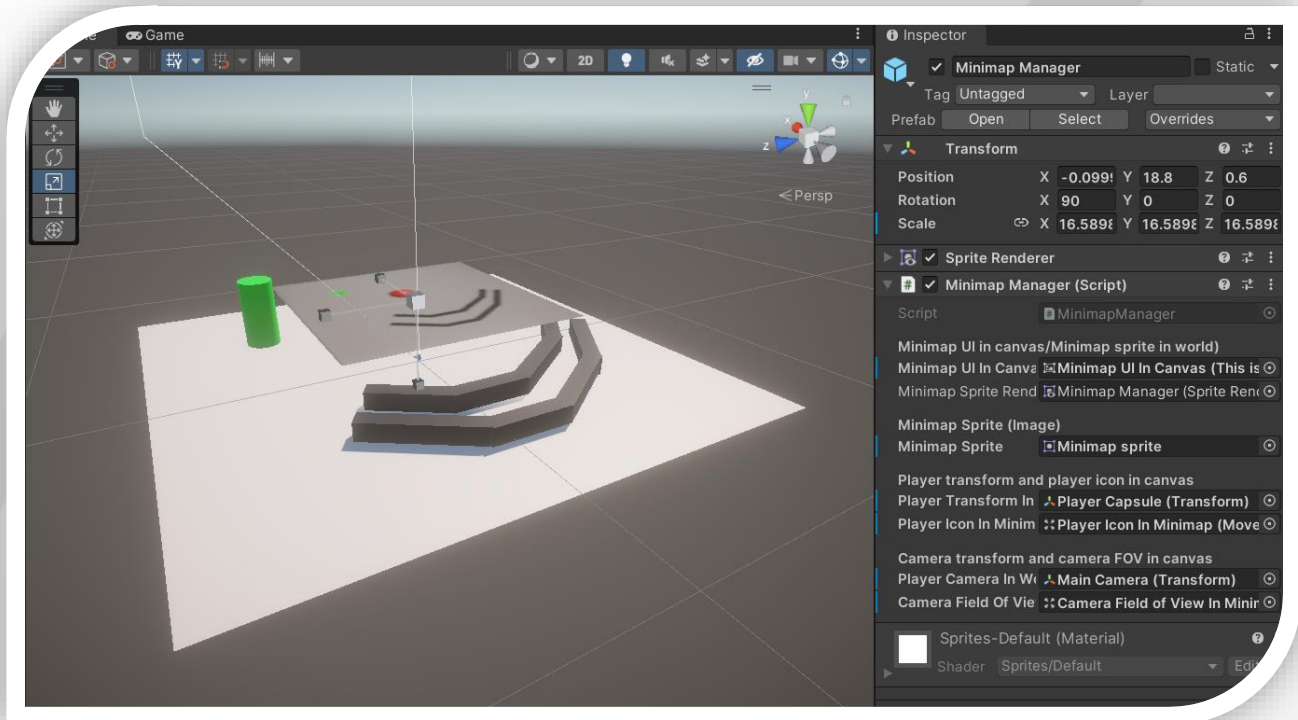Reference to the FOV in the UI in the
canvas.
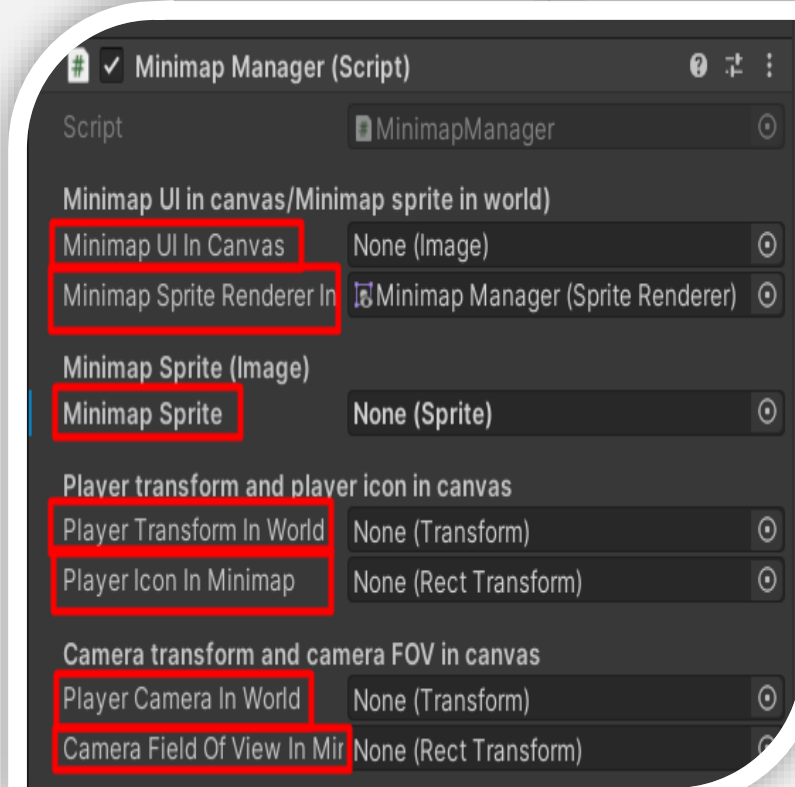
Assigning the References:
- Assign the references as elaporated on the screen shots below
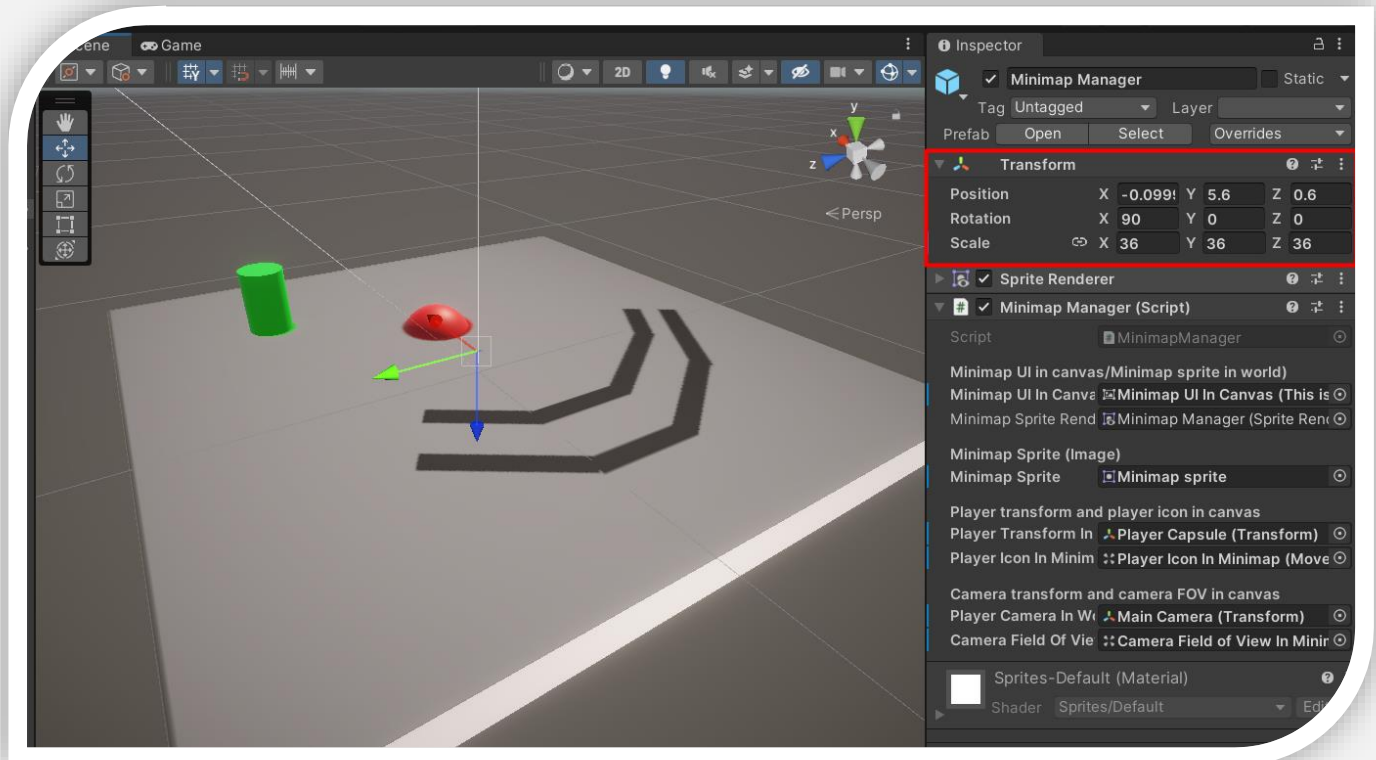
STEP4:

- Once you set all of the references, check your scene view, you will find you minimap in the world as shown below.
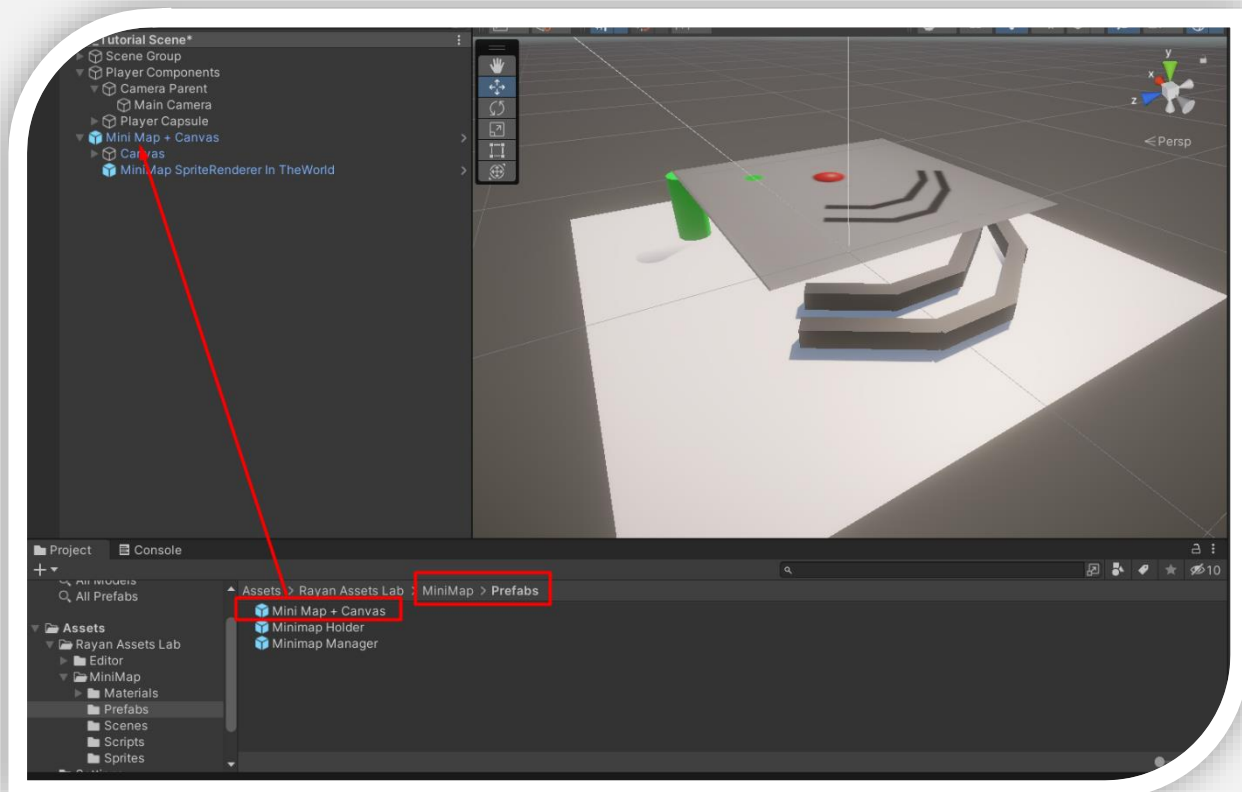

- <u>Modify the transform of the **"Minimap Manager"** to fit the environment. Make sure it aligns with your environement (THIS IS IMPORTANT).</u>
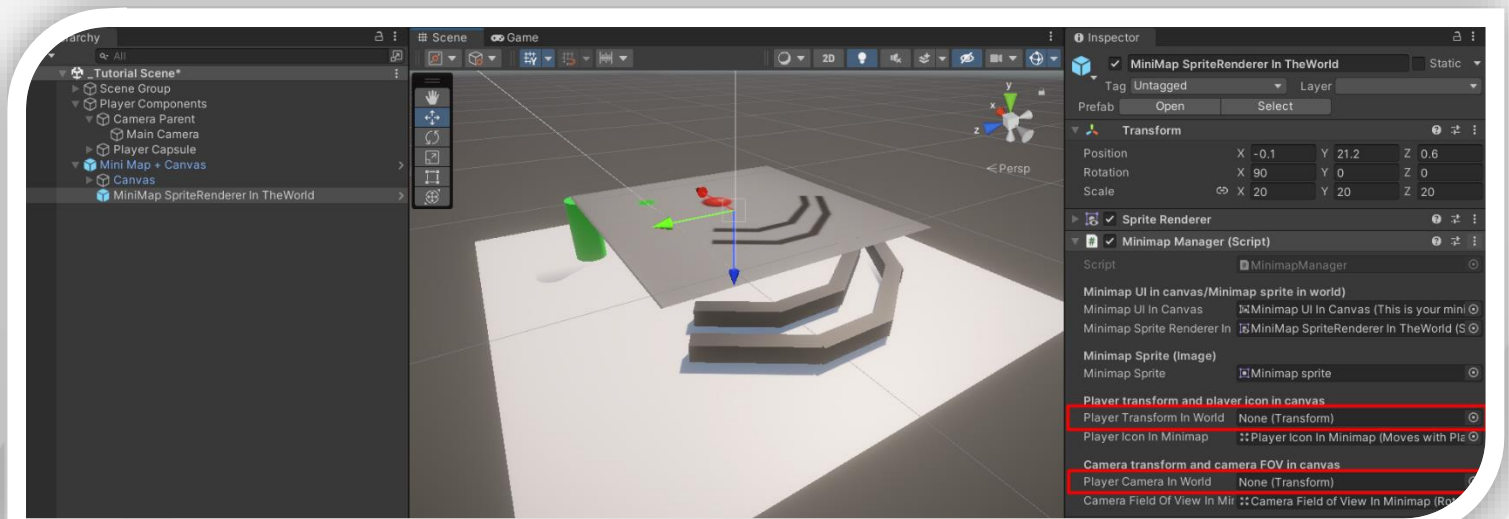

- Now the minimap should work perfectly.

Method (2):

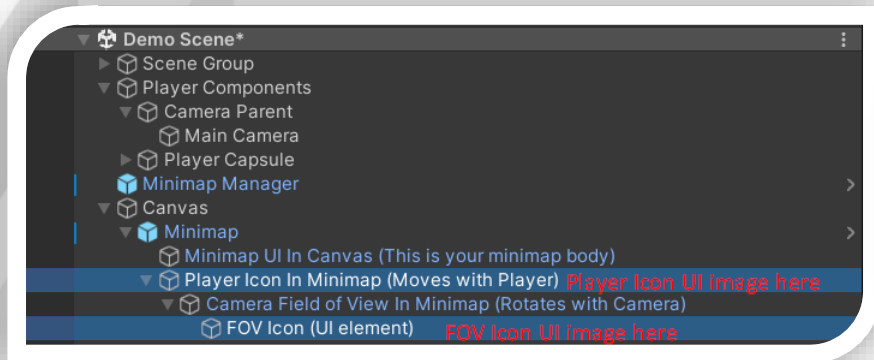- You can use the prefab "Minimap + Canvas", drag and drop it to your scene.

- You have to give the Minimap Manager the reference of both the player and its camera.

- Now follow the instructions of Method (1) STEP4

Notes:

1. You can change both the player Icon and FOV by changing the source image of "**Player Icon in Minimap**" and "**FOV Icon**"

Camera Free Minimap

If you have any suggestions or bugs please contact me so I can improve the asset or add more features.

You can contact me on:
Discord: rayan_games
Email: RayanYoussef1993@gmail.com

Thank you for using my asset!