About the project

I decided to approach the project in the most modular way possible, utilizing the capabilities of ScriptableObjects and Prefab variants to the best of my ability.
The final deliver features a shop with over 12 items spread into 4 categories: Torso, Legs, Head and Face (accessories) and plenty of colors to choose from. This allows users to mix and match clothing to create their own fashion and express themselves in their avatar.

ScriptableObject Clothing Items

New clothing items can be easily inserted into the project by a professional without coding experience thanks to the exposed inspector variables. Items are set using a numeric ID (in which they're ultimately ordered in the shop screen list), a thumbnail icon and a big preview art (used in the full-body preview), as well as its name, description and colors. Colors are laid out int a list and are used to recolor sprites for that clothing piece. More importantly, colors are considered different products altogether, so buying the same piece of clothing twice with different colors is possible and fully supported.

Shop Screen and Inventory functionality

The shop screen loads clothing item ScriptableObjects directly from the Resources folder and populate them into a Unity UI Grid inside a ScrollView. The screen also shows a full body preview art that showcases items every time they're selected. Once a clothing color is purchased, the shop will indicate it is "Sold Out", but internally it's just checking whether the player already possess that item or not.
The Sell and Inventory screens derive from the Shop Screen prefab as prefab variants, both to save time and avoid duplicating resources needlessly. Items are equipped by clicking on their thumbnail or color widgets from the Inventory screen, and the full body preview art will reflect the changes.

Final considerations

- Apart from the animated humanoid character spritesheet found on https://es.pixilart.com/art/gb-topdown-animation-base-4cd32b0857dbd88, everything else was drawn by men so expect a kinda ugly art 😅
- Characters in this project just walk or idle. In a more robust setting, animations should be handled with State Machines, but to save time and focus on the important parts, I've used just a bool instead, which is responsible for switching Animator states.
- Clothing animation frames are based on Editor-set enum drop-downs. A more robust system could use custom-drawn Editor Dictionaries to avoid setting up animation frames for the same directions more than once.
- I like to make a convention of never setting variables as **public** just as a means to serialize them. In my opinion, this is a beginner's habit, when people are still unaware you can serialize variables using attributes. For that, I believe it's more organized to set variables that are used only locally on the script but assigned from Unity Inspector as **[SerializeField] private** instead; hopefully this also helps whoever reviews the code to understand that the given variable isn't used in any other class despite being assigned in the inspector.
- Clothes are basically monochromatic as is. It's is totally doable to create clothing with differently colored patterns, but for that I'd rather use a sprite shader in order to make it more flexible and easy, but I actively avoided diving too deep in shader territory for this test.
- The scenario/map is very crude, built with simple Unity primitives just for testing purposes to save time. I've worked with tilemaps before and, given more time, I'd have done a more presentable environment, but ultimately I didn't feel like it was necessary to import and mess around tile sets in such a short time for this test.