

Input Actions for Player interaction with UI

10/11/25

- I just established a very basic pause menu in the GS-Factory.unity Scene.
- Right now, the menu is called using the letter 'T' on the keyboard, although I will update this to be the Escape key in a later stage.
- It looks like I can copy the UI input part of the default input Actions Map that Unity provides, rather than creating it all from scratch.
- The mouse cursor is NOT visible during game play because it is being used to control the player & the camera's orientation.
- A tutorial online said I can switch the maps in C# using something along the lines of:

```
InputManager.PlayerInput.SwitchCurrentActionMap("...");
```

This creator added a public variable: "public PlayerInput PlayerInput;"

The name of the specific Action Map in the Input Actions Asset.

- Note that the "InputManager" part of that line is just due to the fact that the creator of this tutorial chose to have a game object of the same name be in the scene.
- I have instead chosen to create a Player.cs script that is attached to the player game object, which holds a reference to the PlayerInput component.

- And then, inside of the Pause / Play logic in the UI Manager, I can Switch the Action Map from "PlayerMove" to "UI."
- Additionally, I'm able to make the mouse cursor available for use on these key bindings with the following:

Cursor.visible = true;

Unity automatically recognizes this keyword

Toggles mouse visibility

Cursor.lockState = CursorLockMode.None;

This unlocks the cursor to allow for free movement

- When resuming the game, I just flip the State/value of these Settings.
- I also Switch the current PlayerInput Action Map to be "PlayerMove" again.