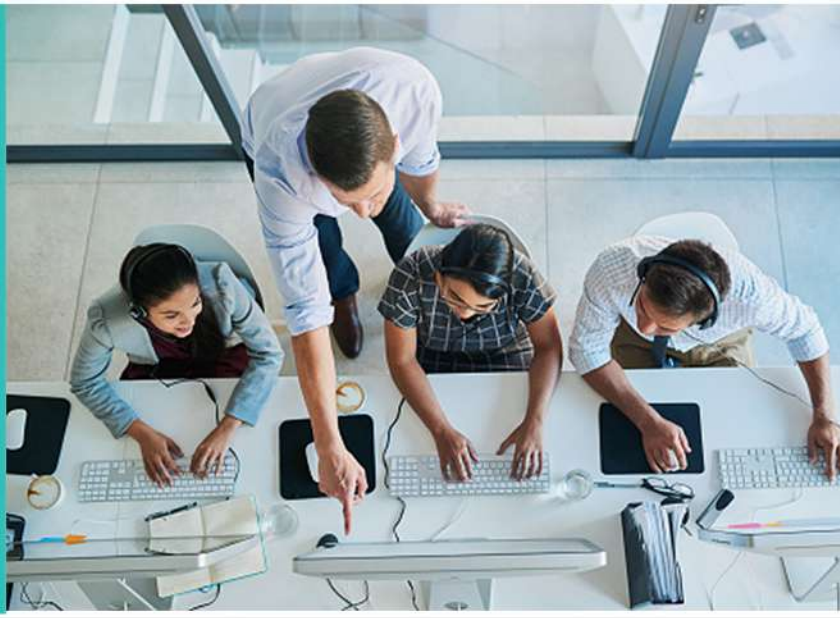
Andhra Pradesh State Skill Development Corporation

Skill AP
APSSDC

Learn Anytime Anywhere

Andhra Pradesh State Skill Development Corporation



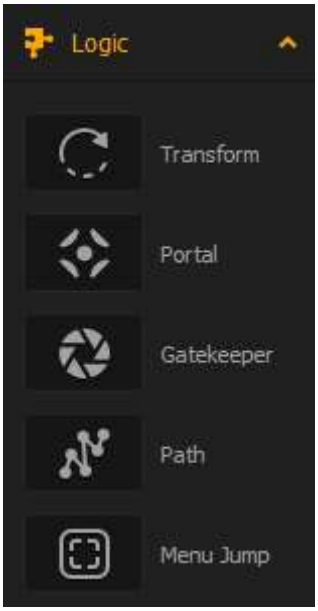
Game Development Using Buildbox

Logic Items

LOGIC ITEMS

7.1 What is Logic

This is where objects that dynamically control game objects appear. It appears under Effects and above Labels on the main bar to the left. There is currently a single Logic item available. They are not the only thing that can be used to alter game play during game play. Actions can also be used to alter game play settings.

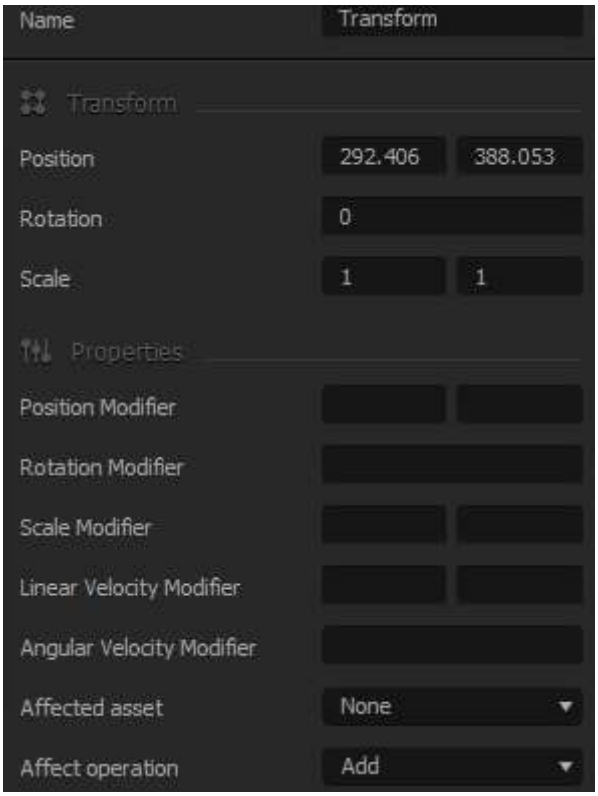


7.1 Logic Items

1. Transform:

A Logic Transform is an invisible game piece that allows you to modify the properties of your existing asset in your game. Specifically you can modify the position, rotation, scale, linear velocity or angular velocity of an asset within your game. To use a Transform item, simply drag onto the active scene.

Once this is done a single instance will be created, and settings can be configured on the right (when selected) as shown below. There is no effective limit to the number of Logic items that can be used on a scene.



7.2 Transform Logic item Settings

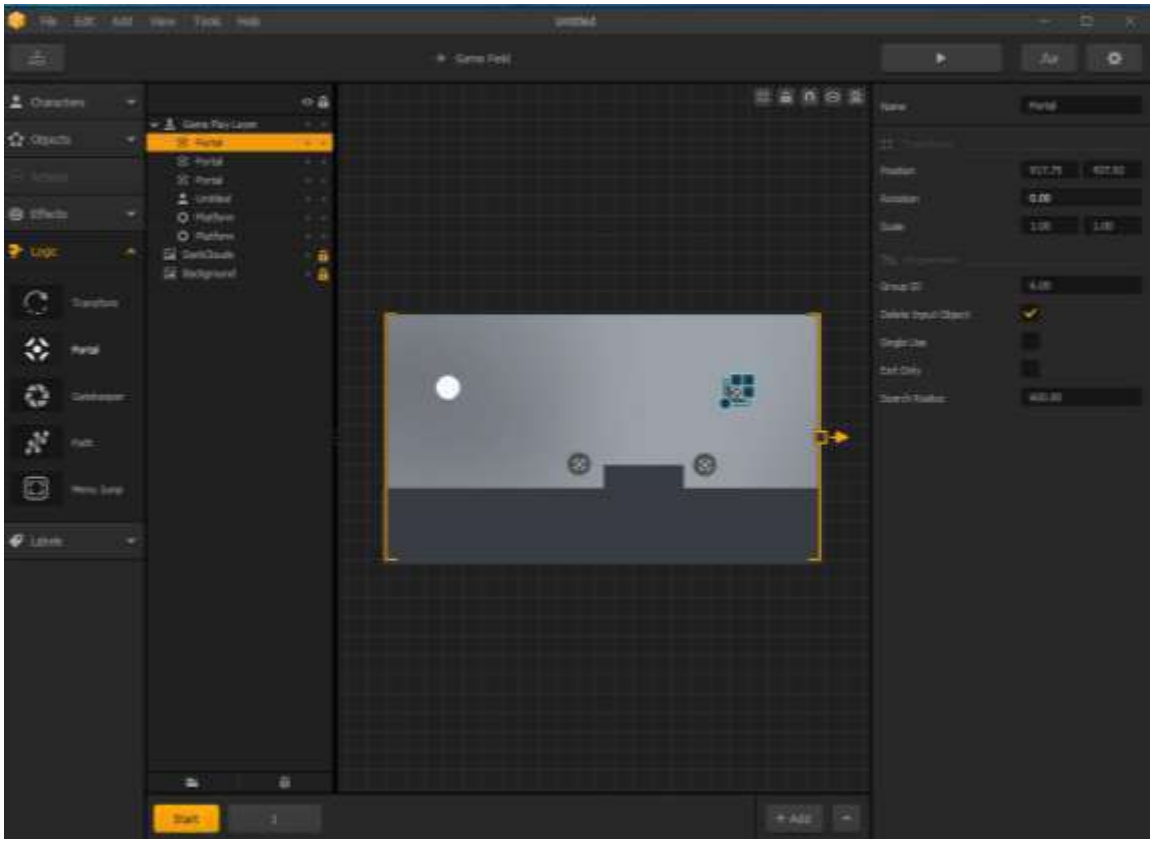
Objects colliding with this will have their setting changed. The available settings are:

- **Position (x,y):** As for any other placed item.
- **Rotation:** Angle in degrees that this should be rotated. Not normally used.
- **Scale (x,y):** Does not have an actual effect.
- **Position Modifier (x,y):** Move the affected object in the x and y directions.
- **Rotation Modifier:** Angle in degrees to rotate the object.
- **Scale Modifier (x,y):** Change the size of the object. Negative values can be used to invert the object – useful for an object that needs to turn around the go back in the opposite direction so the object does not end up moving backwards.
- **Linear Velocity Modifier (x,y):** Change to the speed the object is moving
- **Angular Velocity Modifier (x,y):** Change to the speed the object is rotating.
- **Affected asset:** A dropdown of all Objects in the game. One modifier is needed per Object you wish to affect. This is also where a good naming convention will make your job as a game builder much easier! If you want to affect the player Character or World, you need to look at Action, Gameplay type (see Gameplay Properties).
- **Affect operation:** How to apply the settings listed here, to the Object values in-game.
    - **Add:** Add values to the existing values for the affected object.

- **Multiply:** Multiply existing values for the affected object. Useful when turning an object around by applying -1 values for example.
- **Replace:** Change all values you have set. If you set all values then all values will change. If you set only some values – leaving the rest blank – then only the changed values will alter.

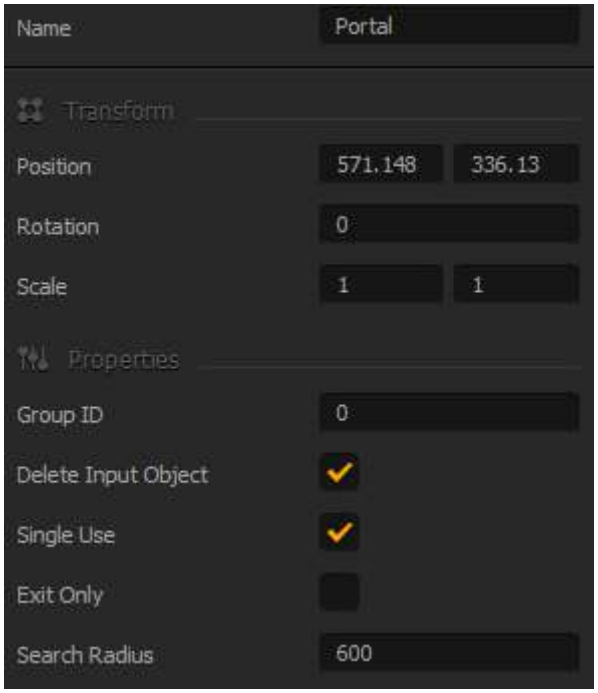
2. Portal:

Portals are used for jumping the Character from one place in a World, to another – and optionally duplicating the Character (something Agent 86 would be familiar with).



7.3 Portal Item placing into a scene

Properties:



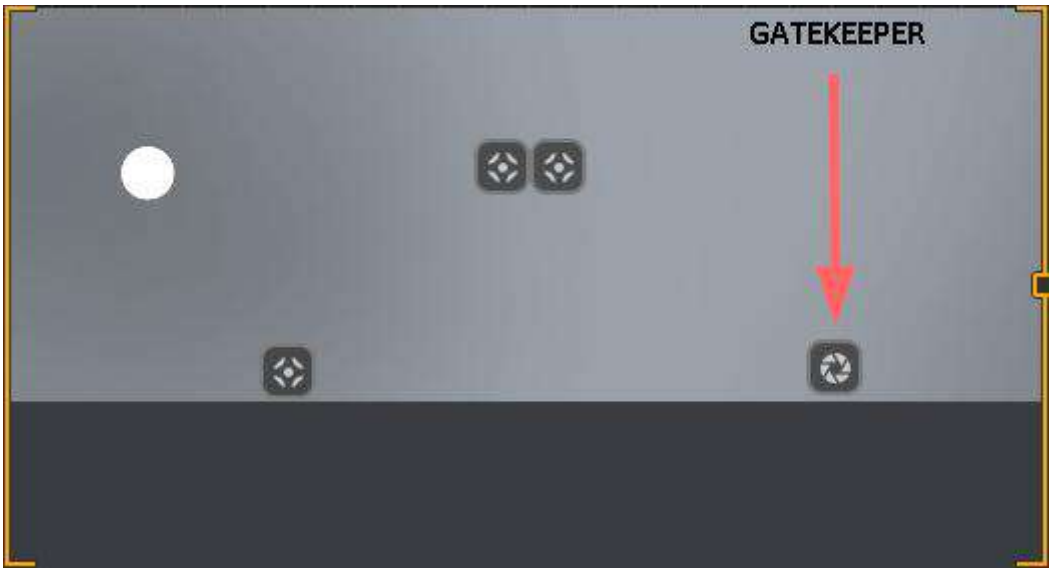
7.4 Portal Logic item Settings

The available settings are:

- **Position (x,y):** As for any other placed item.
- **Rotation:** Angle in degrees that this should be rotated. Not normally used.
- **Scale (x,y):** Does not have an actual effect.
- **Group ID:** This portal will transport to(/duplicate) or from other portals with this group ID. If there are more than two portals with the same id, the Character will be duplicated.
- **Delete Input Object:** If this is not ticked, the portal will be a duplicator. The original will still be at the entering Portal.
- **Single Use:** Can only be activated once in the game.
- **Exit Only:** Portal will not activate when moved into. It will still act as an exit, however.
- **Search Radius:** How far away the character can be and still trigger the portal open.

3. GateKeeper:

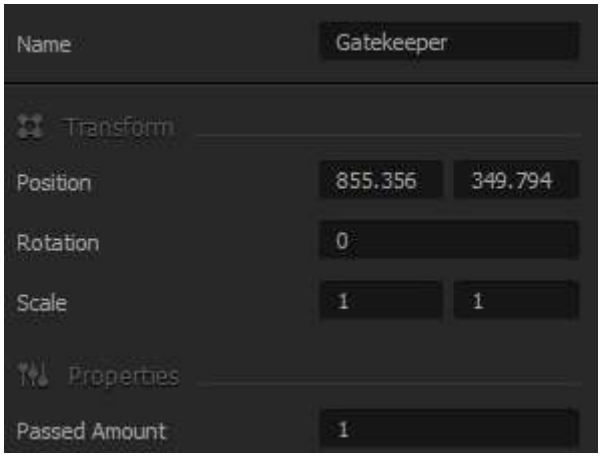
Gatekeepers are used to prevent a player Character from passing more than a number of times through an area. Used to create, for an example, a one way door (though they are more flexible than that).



7.5 Scene View of Gatekeeper

Properties:

Once you select it you will see its available options on the right hand side.



7.6 Gatekeeper Logic Item Settings

The available settings are:

- **Position (x,y):** As for any other placed item.
- **Rotation:** Angle in degrees that this should be rotated. Not normally used.
- **Scale (x,y):** Does not have an actual effect.
- **Passed Amount:** Number of times the Character will be allowed to pass through the GateKeeper. Every time the Character passes through it, this number will reduce by one until it reaches 0. Once it is zero, it will be impassible.

4. Path:

A Path will let you take the Character on a ride. Control by the player ceases, while the Character moves along the line you have defined. If there are physical Objects, or Logics, etc, they will interact (or interrupt) the Character while on the Path.

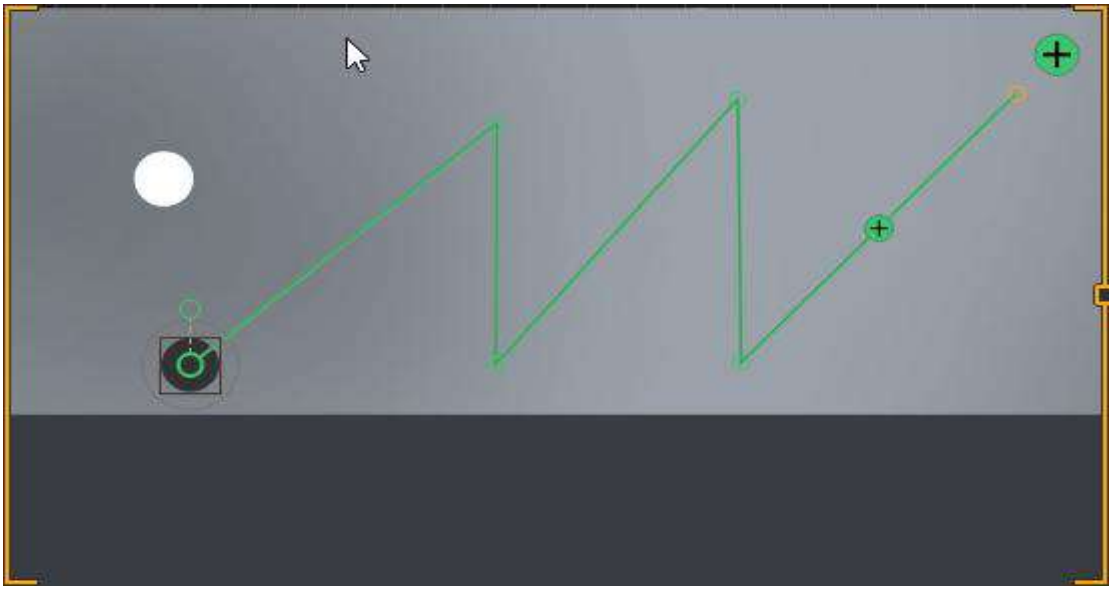
If we drag a Path Logic piece to our scene editor, we will have the option to create a path for either our character or any object to follow. The + sign lets you add joints and the small circles let you control the direction.

After dragging onto the design screen, you will see the Path represented as a green line. Note the following elements:

 Click this to select the Path

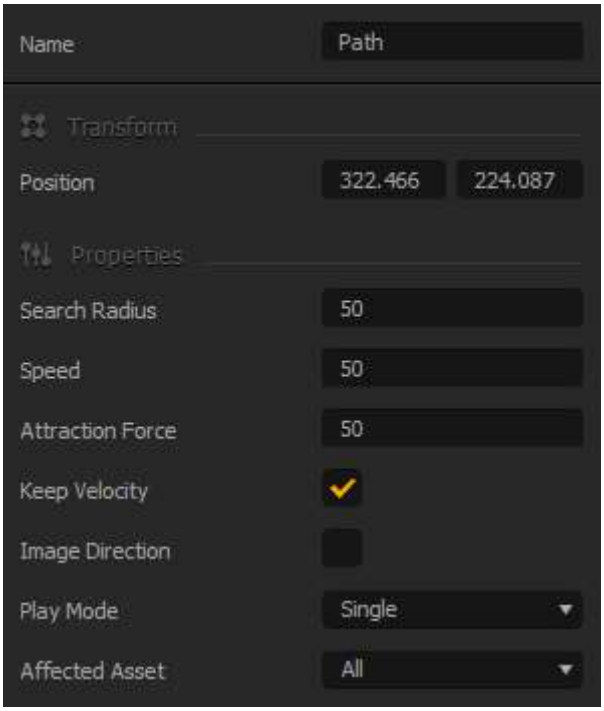
 Click the larger version of this to add a segment to the end of the line. Click any of the smaller versions in the middle of a segment, split a segment in two (if they are not currently visible, the split action will still occur).

 Click and drag a corner like this to move it. Let's look at the below illustration.



7.7 Scene View of Path

Properties:



7.8 Path Logic Item Settings

The available settings are:

- **Name:** For internal reference, only.
- **Position (x,y):** As for any other placed item.
- **Search Radius:** How close the Character needs to be to activate this path.
- **Speed:** How fast the Character should move. If this was set to a very high value, it would look a lot like the actions of a Portal (provided there was nothing in the way to interact or stop the Character of course).
- **Attraction Force:** How much force each point in the path will attract the Character, in turn. This is a setting that is a lot of fun to play with. With very low values you can get effects similar to Angry Birds™ Space around each point – that can be resisted by the player controls and other forces including the World's gravity. Allowing the player to fight the Path, means they can eventually move out of its influence, when they get far enough away.
- **Keep Velocity:** When Character finishes a path, it should have its original velocity.
- **Image Direction:** Tick this to keep the player pointing the 'right' direction on the Path.
- **Play Mode:** How the Character will move along the Path
    - Single: From beginning to end, and exit.
    - Loop: From beginning to end, then jump back to the start.

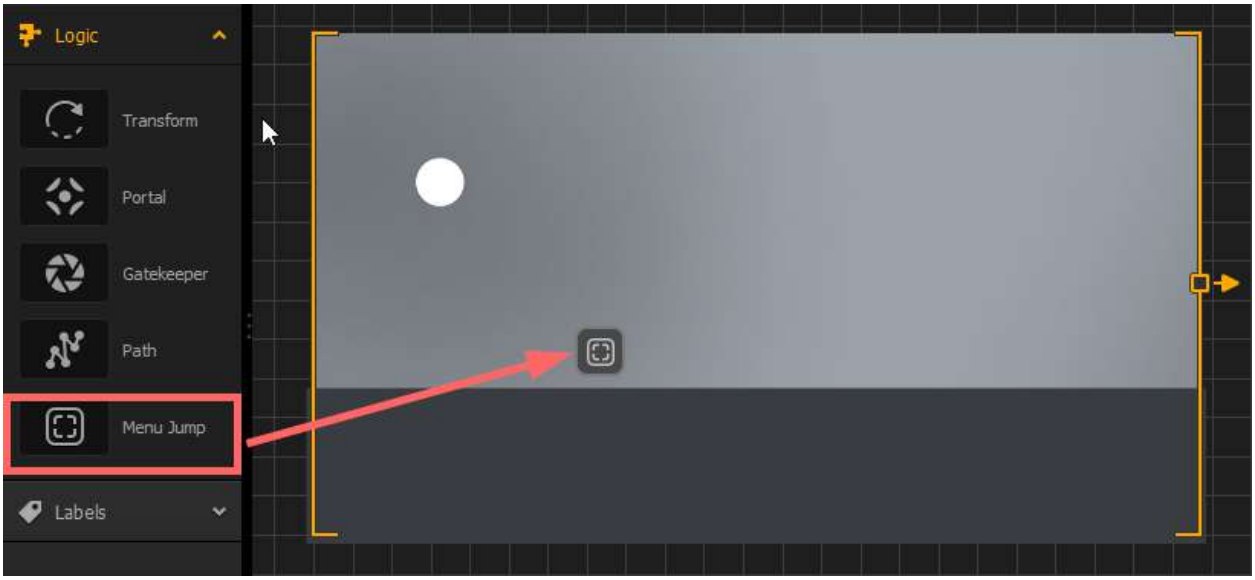
- Ping Pong: From beginning to end, then end to beginning.

- **Affected Asset:**
  - All: For all Objects and Characters.
  - Characters: For Characters only.
  - Assets: For Objects only.

5. Menu Jump:

The Menu Jump is a very easy yet extremely powerful tool. This will allow you to jump to any world or menu. Here is how it works

After dragging a Menu Jump Logic piece in the scene editor, edit its name to something like secret world. Also remember that this logic piece is invisible so you may want to add a decoration or a particle emitter to it.



7.9 Scene View of Menu Jump

Properties: Once you select it you will see its available options on the right hand side.

7.10 Menu Jump Logic Item Settings

The available settings are:

- **Position (x,y):** As for any other placed item.
- **Rotation:** Angle in degrees that this should be rotated. Not normally used.
- **Scale (x,y):** How big the jump point will be.
- **Pause Current World:** Keep current world in a frozen state, so that when the Character returns to this world, they can continue from the current location and game state.