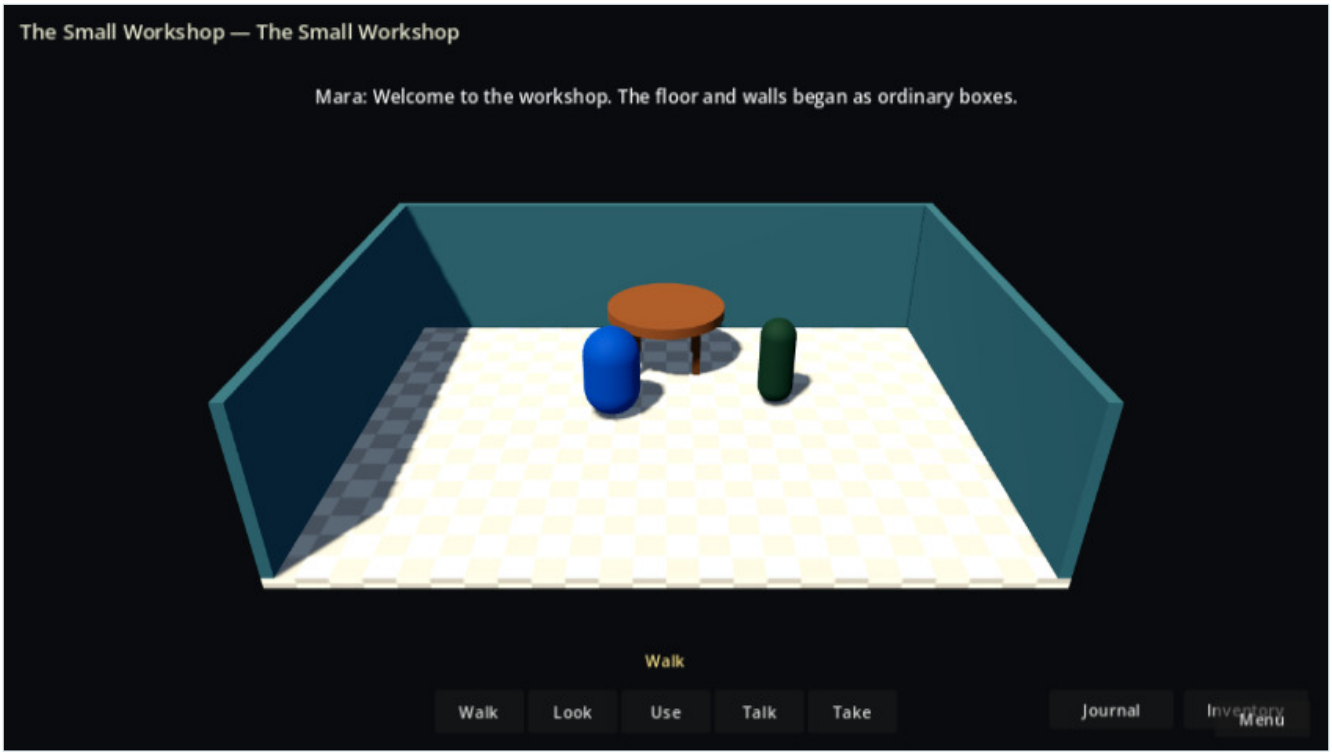
PCA 3D ADVENTURE STUDIO

The Scene Designer

Build your first 3D room, one piece at a time.



A floor. Three walls. Two characters. One working camera.

A guided workshop lesson using actual editor screenshots. Build the room with simple shapes, add colours or textures, give the player somewhere to walk, and finish with a playable scene.

For version 0.86.3.4 • Windows, macOS and Linux

No companion project or practice files are required.

0.86.4.0 / Move a native 2D object > p. 61

How to use this guide

Start at Chapter 1 and build the same small room as you read. Each part explains what to click, what to enter and what to check before carrying on. The measurement reference at the back collects the important values in one place.

The pictures come from a running copy of the editor. Some show the whole workspace; others are cropped to keep a setting readable. The theme and panel widths may differ from your copy. Follow the labels, selected record and tab names.

Part	What you will finish
Chapters 1–4	A new 3D room and a working understanding of selection, transforms and primitives.
Chapters 5–12	A floor, walls, materials, lighting, navigation and a simple workbench.
Chapters 13–20	An NPC, the player, project startup and the room camera.
Chapters 21–24	A play test, room organisation and answers to common problems.
Reference pages	Exact measurements and a final checklist.

Use solid colours if you do not have a suitable texture image. The room still works. Imported character models are optional too: the built-in placeholders let you check movement and interaction first.

Three habits will save you time

Read the Inspector heading before editing. Apply Changes after a finished set of fields. Save Project when you reach a useful stopping point.

When a screenshot shows more settings than the step mentions, change only the fields named in that step. Other tabs are explained in the reference chapter when they become relevant.

Contents

Click a chapter title or page number to jump to it. PDF bookmarks provide the same chapter list, with section headings underneath.

1. Start with a new room . . . 4
2. The three parts of the Scene Designer . . . 7
3. Looking around without moving the room . . . 8
4. Add a primitive . . . 10
5. Turn the box into a floor . . . 12
6. Build the walls . . . 15
7. Give the room its colours . . . 18
8. Texture the floor or a wall . . . 21
9. Add real room lighting . . . 25
10. Set the walkable area . . . 28
11. Blocking, colliders and event volumes . . . 31
12. Add a workbench with a generator . . . 34
13. Create an NPC definition . . . 35
14. Put the NPC in the room . . . 38
15. Place and identify the player . . . 40
16. Choose the project's starting player . . . 42
17. Set up the room camera . . . 44
18. Make it the default camera . . . 46
19. Copy an editor view into a camera . . . 47
20. Give Mara one simple response . . . 49
21. Play-test the room . . . 50
22. Keep the room manageable . . . 53
23. The rest of the scene tools . . . 55
24. Fix the common problems . . . 57
R1. The workshop measurements . . . 59
R2. A final room check . . . 60

1. Start with a new room

This guide builds a small, usable workshop in PCA 3D Adventure Studio. You will make the floor and three walls from ordinary boxes, give them colours or textures, place a workshop keeper and the player, and choose the camera the game should use. Then you will walk around the room and try an NPC interaction.

The instructions follow version 0.86.3.4 on Windows, macOS and Linux. The pictures are actual captures from the running editor. File pickers and window borders can differ between operating systems, but the scene settings have the same jobs.

You can complete the building lesson with solid colours. The optional texture steps use an image of your choice; it does not have to match the tile pattern in the pictures. There are no project files or asset packs to download with this book.

Create the project

1. From the opening screen, click **New Game**.
2. Enter **The Small Workshop** as the title, then click **Create Game**.
3. Save the new adventure in your own working folder when the save window opens.
4. Open the **Scene Designer** if it is not already visible.
5. Select the starter room in **Scene Objects**. Select the room name itself, rather than the player or camera beneath it.
6. In the Inspector's General tab, set Scene Name to **The Small Workshop** and Scene ID to `workshop`. Click **Apply Changes**, then Save Project.

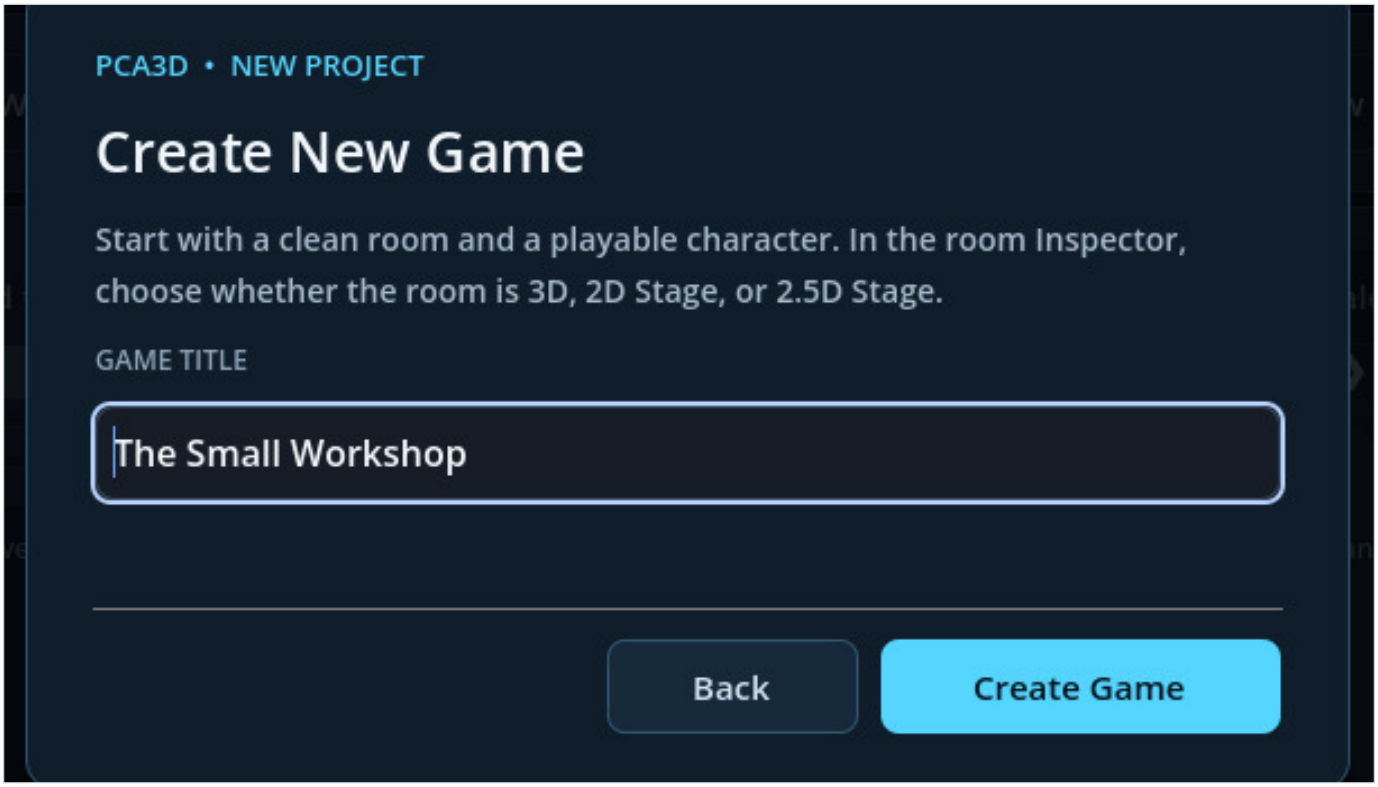


Figure 1. New Game starts a clean adventure with a starter room and a playable character. Enter your own title before creating it.

A new project already contains a player definition and a player placed in its starter room. Keep that player. Later chapters show how to move it, check its player settings and add it again if you have removed it. Adding a second copy is unnecessary.

Choose a 3D room

In this version, the one-time room-family choice is inside the Inspector's **2D / 2.5D Room** tab, even when the room you want is fully 3D.

1. Keep the room selected.
2. Open **2D / 2.5D Room**.
3. Under **Permanent Room Family**, choose **3D Room** in **Room Family / Stage Variant**.
4. Click Apply Changes and save.

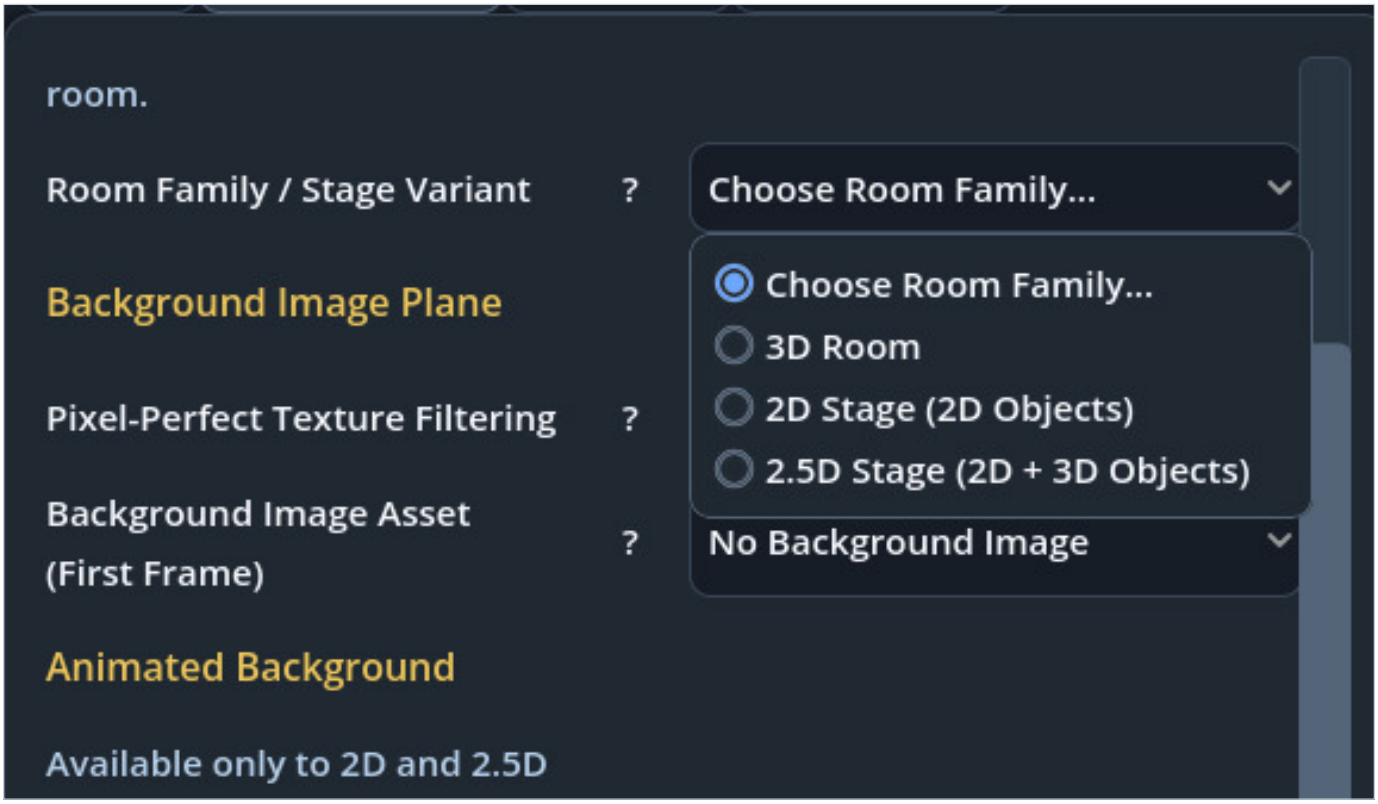


Figure 2. Choose 3D Room here. The tab title is broader than the particular choice being made.

Applying this choice locks the room to the 3D family. The Scene Designer's separate 3D, 2D and 2.5D viewing controls are not a way to undo that room-family decision. Create another room if you need a different family later.

> **Before you continue**
> You should have one room called The Small Workshop and the supplied Player character. Save before adding geometry.

2. The three parts of the Scene Designer

The Scene Designer works best when you keep three things in view: the room's object list, the scene itself and the Inspector. Each answers a different question.

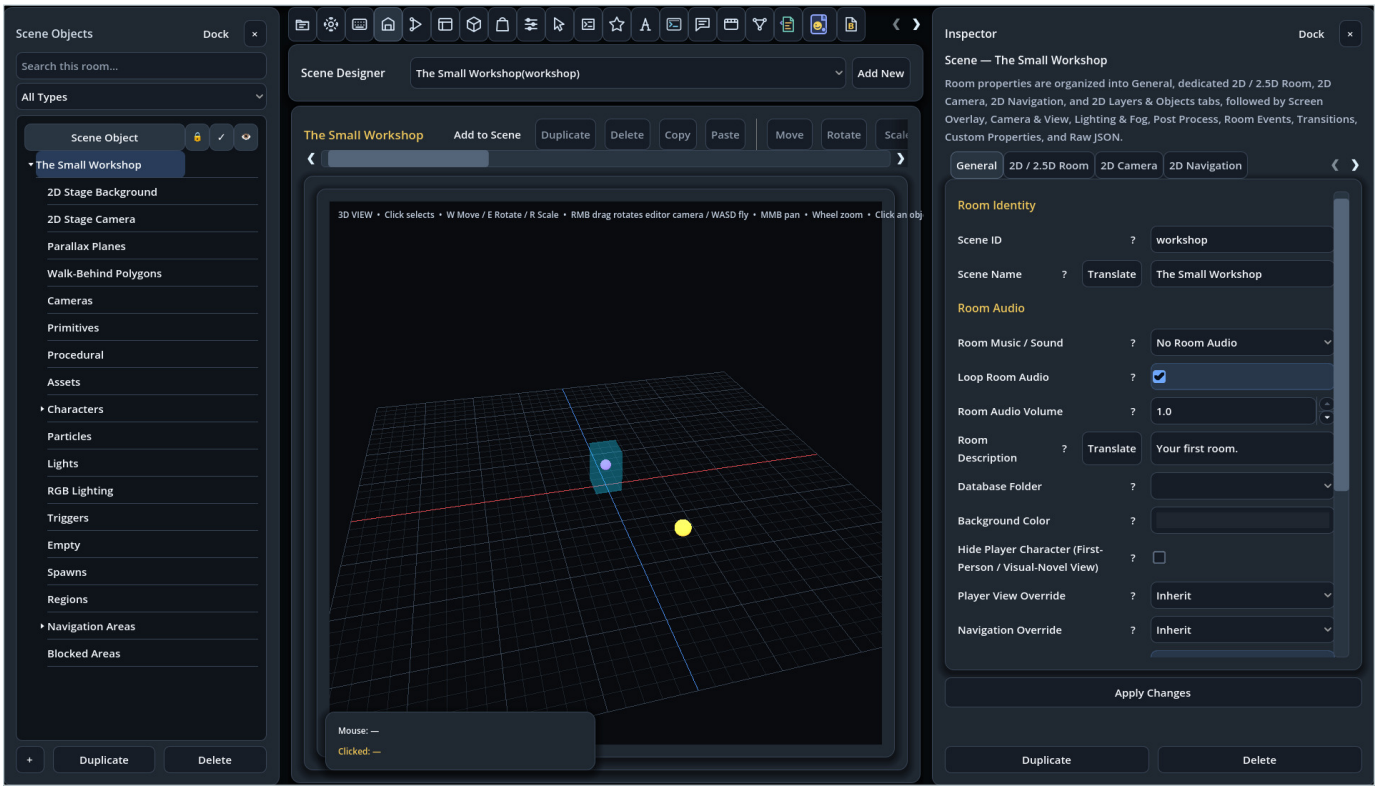


Figure 3. The 3D workspace before construction. Scene Objects is on the left, the editing view is in the middle, and the selected record's Inspector is on the right.

Area	What it tells you	Typical use
Scene Objects	What belongs to the current room.	Select a wall that is hard to click in the view, or find a camera, character or navigation area.
Editing view	Where those things sit in 3D space.	Move, rotate, resize and inspect the room.
Inspector	The selected item's actual settings.	Enter an exact size, position, colour, character reference or camera target.

The **Adventure Database** is the wider project catalogue. It contains reusable characters and assets, as well as all rooms. **Scene Objects** shows the contents of the current room. A character in the database and a placement of that character in the room are related records, rather than the same editing job.

Select the right record

Click a visible object in the editing view, or select it by name in Scene Objects. Read the Inspector heading before typing. **Scene - The Small Workshop** means room properties. **Entity - Floor** means the floor. A camera or character has its own heading and tabs.

After using Add to Scene, select the newly added item in Scene Objects if its properties are not already showing. This simple check prevents a surprisingly common mistake: changing the room's name when you meant to name a box.

Search this room narrows the object list. The type filter beneath it helps when a room contains many lights or props. Expand the little arrows beside categories to see their contents.

Apply and Save

Inspector edits can preview live, but the working record still needs **Apply Changes**. **Save Project** writes the adventure to disk. Use both after each completed part of this lesson.

Gizmo movements are committed through the designer. You should still save the project after a useful set of changes. Undo is helpful during editing; a saved checkpoint is better protection when you close the editor or make a large experiment.

If the Inspector is cramped, drag the divider to give it more width. Scroll within the Inspector to reach lower fields. Its tabs also scroll sideways; use their arrows when a tab such as Visual Effects is off the right edge.

3. Looking around without moving the room

The editor's inspection camera is different from the camera the player will see in the game. You can orbit the editing view freely without changing your authored Room View camera.

Control	What it does
Right mouse drag	Rotates the editor camera.
W, A, S and D while navigating with the right mouse button	Flies around the editor view.
Middle mouse drag	Pans the editor view.
Mouse wheel	Zooms the editor view.
Perspective, Top, Front, Right or Isometric view menu	Gives a useful inspection angle.
Frame Scene	Fits the room into the editing view.

Control	What it does
Frame Selected, or F with the view focused	Centres the selected item for inspection.

Use Top when checking a floor plan. Use Front or Right to check whether walls and props meet the floor. Return to Perspective or Isometric to judge the room as a whole.

Move, Rotate and Scale

Select an object, then choose **Move**, **Rotate** or **Scale** in the designer toolbar. The shortcuts are W, E and R when you are working in the view. Drag the appropriate coloured gizmo handle. When entering numbers, keep the keyboard focus in the field instead.

Position moves the object. **Size** describes a primitive's underlying dimensions. **Scale** multiplies those dimensions. For the floor and walls in this lesson, leave Scale at **1, 1, 1** and use Size for the measurements. That keeps the arithmetic straightforward.

Axis	Meaning in this room
X	Left and right. Negative X is left of the origin.
Y	Height. The floor's upper surface will be at Y 0.
Z	Front and back. The back wall is on the negative-Z side.

The current 3D Rotation fields hold radians. All floor and wall rotations in this lesson are zero, so you do not need to convert angles. When you do need a quarter turn, about 1.5708 radians equals 90 degrees. The light chapter supplies the exact values for its fields. A field explicitly labelled Degrees, such as some animation or facing controls, should use degrees instead.

Grid and Snap

Grid shows a measuring aid. Snap helps movements land on regular increments. The number beside Snap is the spacing; it is not the room's width or the limit of player movement.

For coarse layout, a spacing of 0.5 is convenient. Enter the exact values from this book in the Inspector for the final placement. If an object seems to jump while you drag it, check Snap before assuming the object is broken.

4. Add a primitive

Primitives are simple shapes. They are ideal for establishing the size and layout of a room before worrying about decorative models.

1. In the Scene Designer, click **Add to Scene**.
2. Open **Primitives**.
3. Choose **Box**.
4. Select the new box under Primitives in Scene Objects.
5. In General, set Entity ID to `floor` and Name to **Floor**.
6. Turn **Blocks Other Characters** off for this floor.
7. Apply Changes.

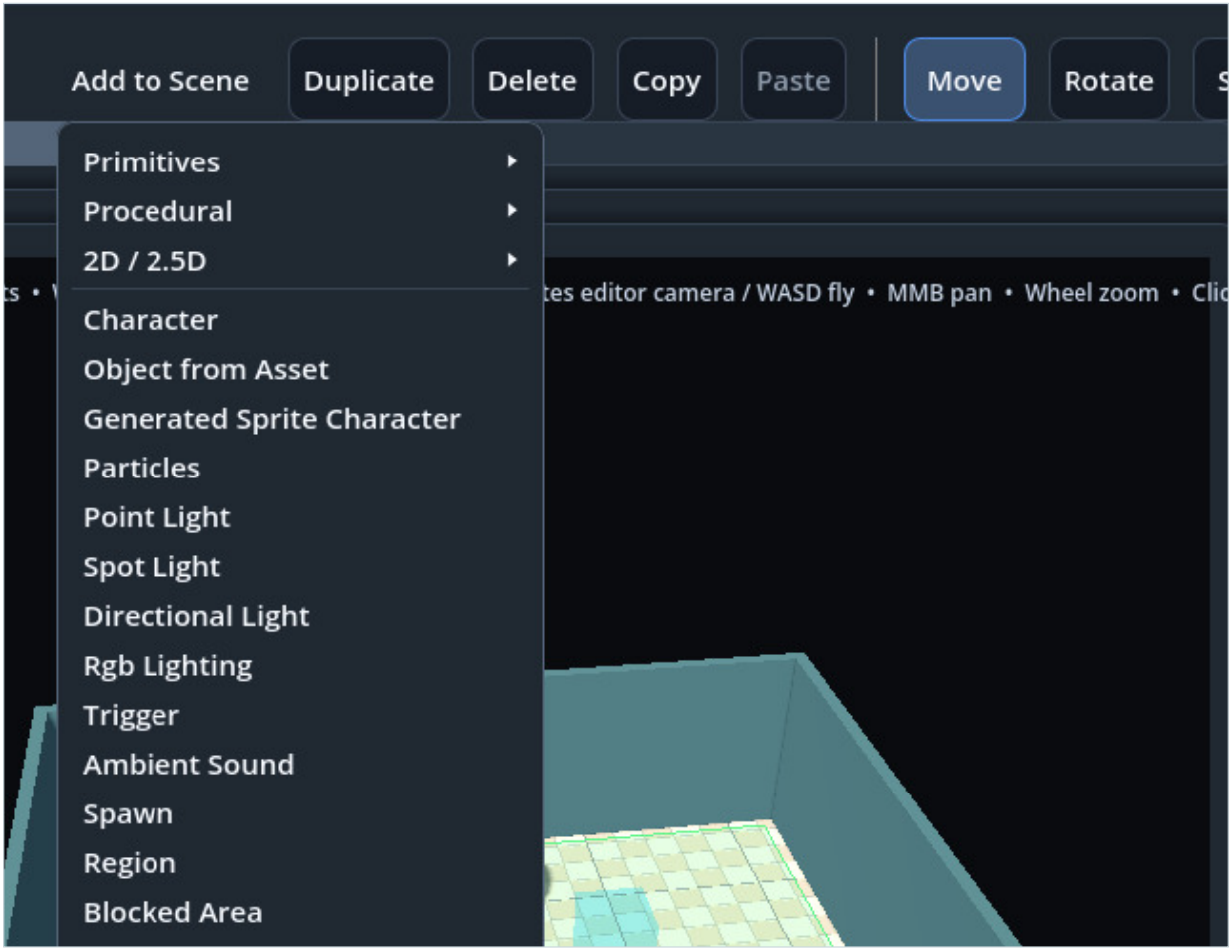


Figure 4. Open Add to Scene, then open its Primitives submenu to choose Box, Sphere, Cylinder or Plane.

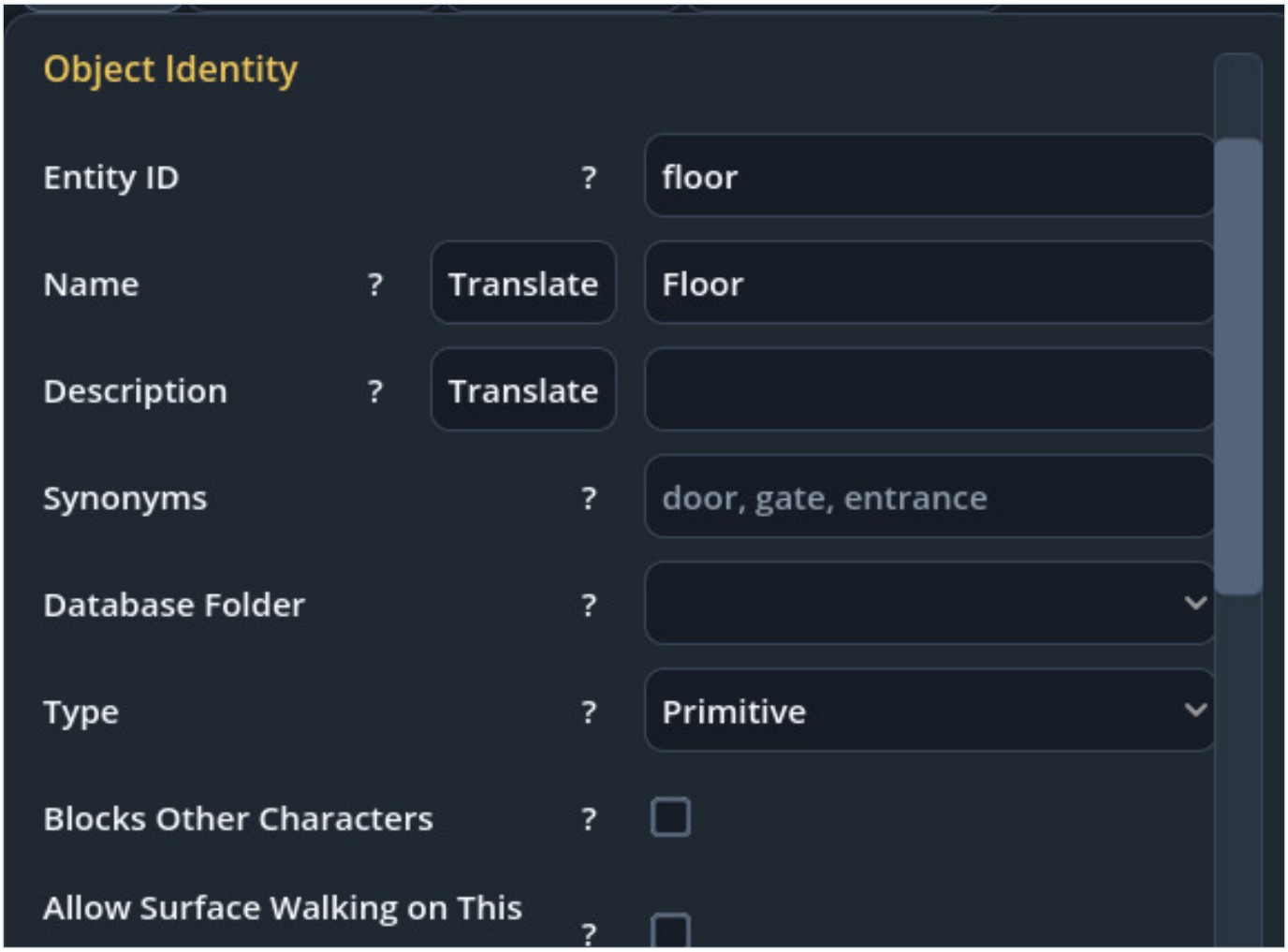


Figure 5. Select the new box and name the object before changing its geometry. Check the Inspector heading as you work.

Shape	A useful first job
Box	Floors with thickness, walls, counters, steps and rough furniture shapes.
Sphere	A ball, a decorative object or a stand-in marker.
Cylinder	Columns, posts, barrels and round supports.
Plane	A flat visible surface with no box-like thickness.

This lesson uses a thin Box for the floor. It is easy to see its thickness and calculate where its top surface is. A Plane is also useful, but it does not replace the navigation setup which tells the actor where it may walk.

Do not change the new object's Type to Character just to turn it into a player. A character needs a reusable Character definition. We will use that workflow in the NPC and player chapters.

5. Turn the box into a floor

A box is centred on its position. If its height is 0.2, half of that height extends above its centre and half below it. We will put the centre at Y -0.1, making the top surface exactly Y 0.

Set the dimensions

Keep Floor selected. In General, scroll to **Additional Geometry Settings**. Set Primitive to Box if necessary, then enter:

Size field	Value
Size X	12
Size Y	0.2
Size Z	9

Click Apply Changes.



Figure 6. The floor is a 12-by-9 box with a thickness of 0.2. Size is near the bottom of the object's General tab.

Put the top surface at walking height

Open **Visual Adjustments**. Set these World Transform values:

Setting	X	Y	Z
3D Position	0	-0.1	0

Setting	X	Y	Z
Rotation	0	0	0
Scale	1	1	1

Apply and save.

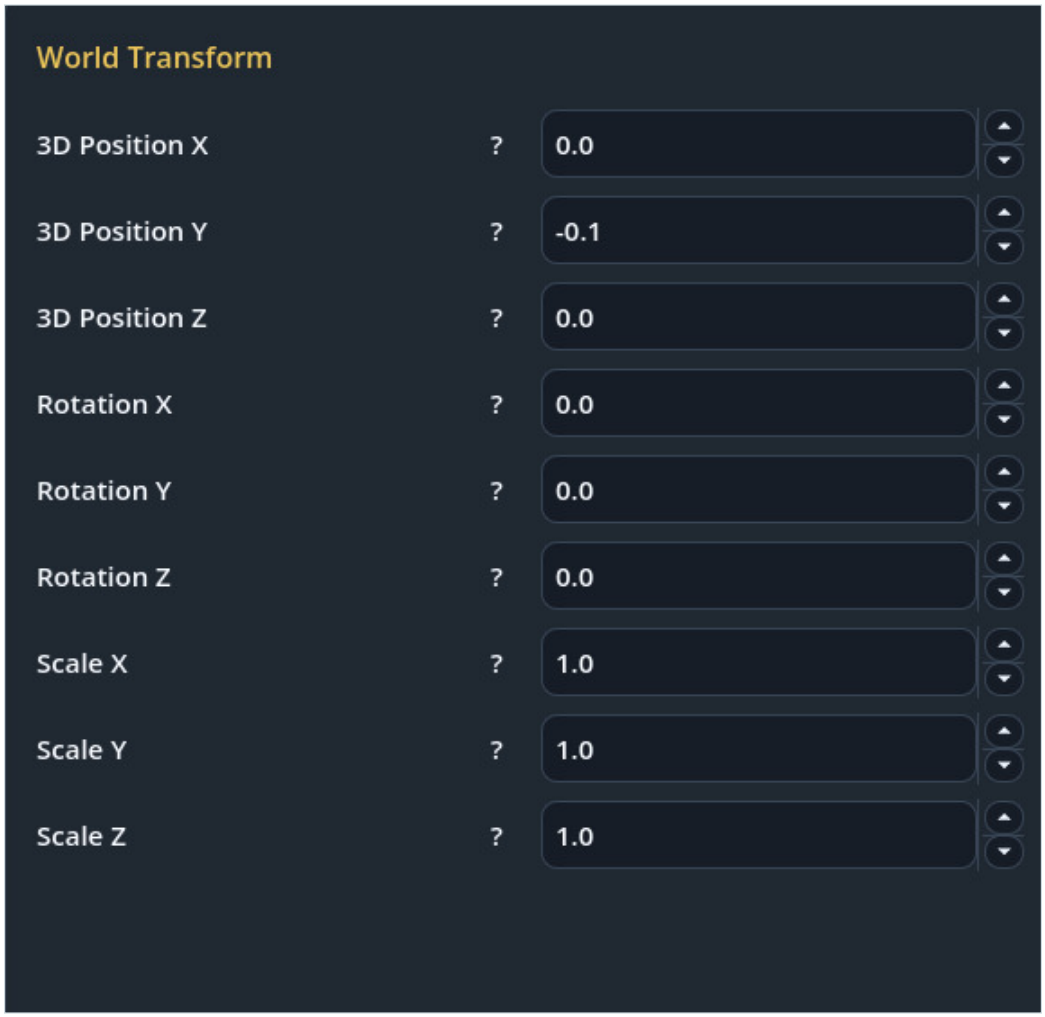


Figure 7. Moving the centre down by half the thickness puts the floor's top at Y 0.

The visible floor now runs from X -6 to +6, and from Z -4.5 to +4.5. The walking area will sit slightly inside those edges. That leaves room for the walls and for the character's body radius.

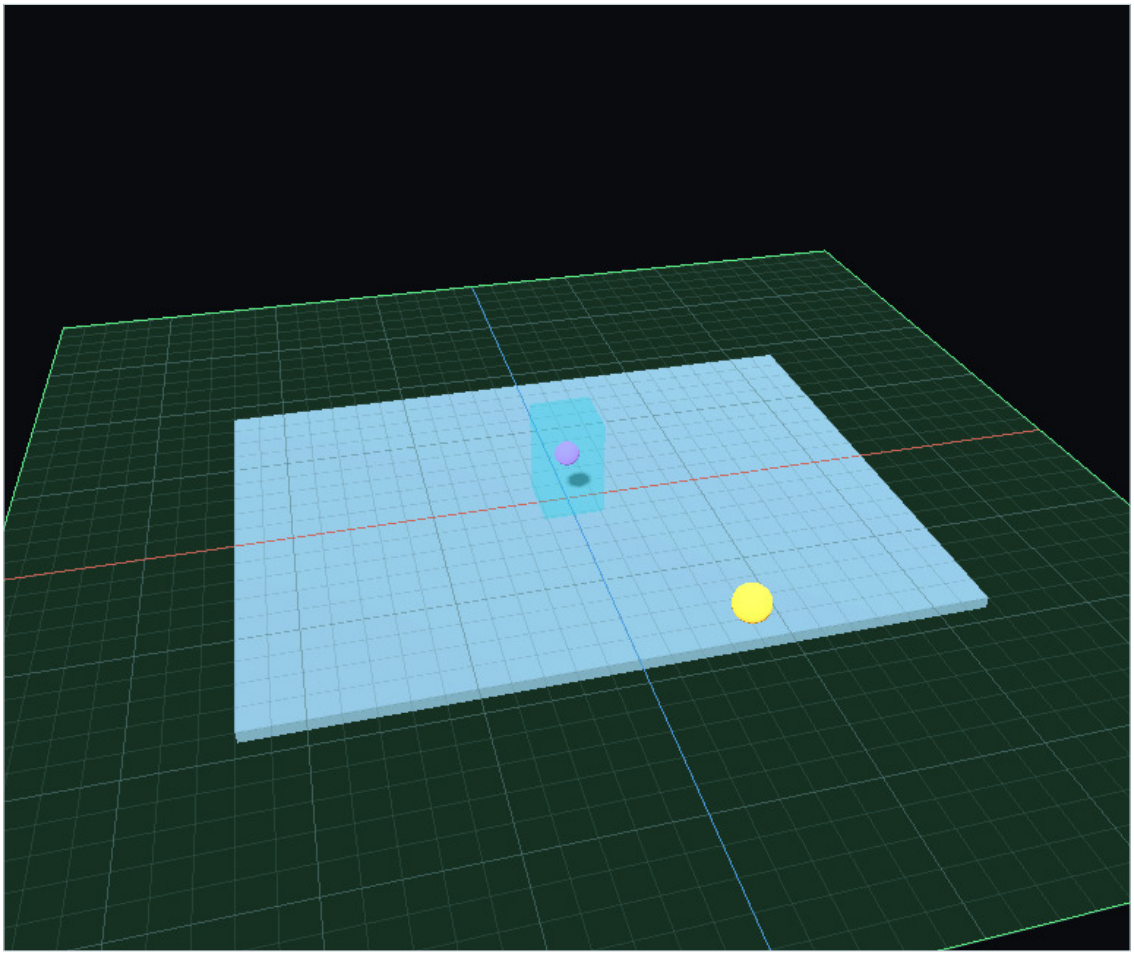


Figure 8. The first building piece in place. A visible floor and a walkable navigation area are separate parts of the room.

If the green navigation overlay or grid makes the floor colour hard to judge, use Editor Visibility or the Grid switch to simplify the editing view. Those inspection switches do not change the floor's material in the game.

> **Before you continue**
> Select Floor again. Its Size should be 12, 0.2, 9 and its Position 0, -0.1, 0. Save the project before adding walls.

6. Build the walls

We will add a back wall and two side walls, leaving the front open. This gives the camera a clear view and makes the room feel like a small stage set. A front wall would stand between this camera and the player.

The back wall

1. Add another **Primitives > Box**.
2. Select it in Scene Objects.
3. Set Entity ID to `wall_back` and Name to **Back Wall**.
4. Leave Blocks Other Characters on.
5. Under Additional Geometry Settings, set Size to **12, 3, 0.2**.
6. In Visual Adjustments, set 3D Position to **0, 1.5, -4.4**.
7. Leave Rotation at 0, 0, 0 and Scale at 1, 1, 1.
8. Apply and save.

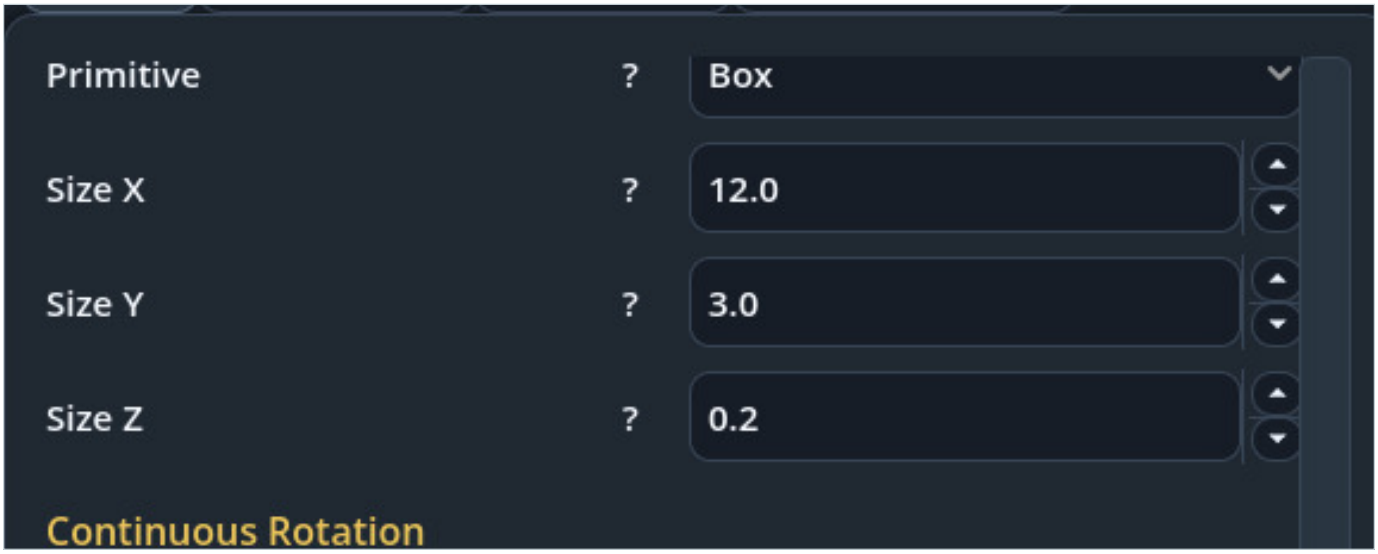


Figure 9. The back wall is wide and thin. Its height is 3, while the 0.2 thickness is along Z.

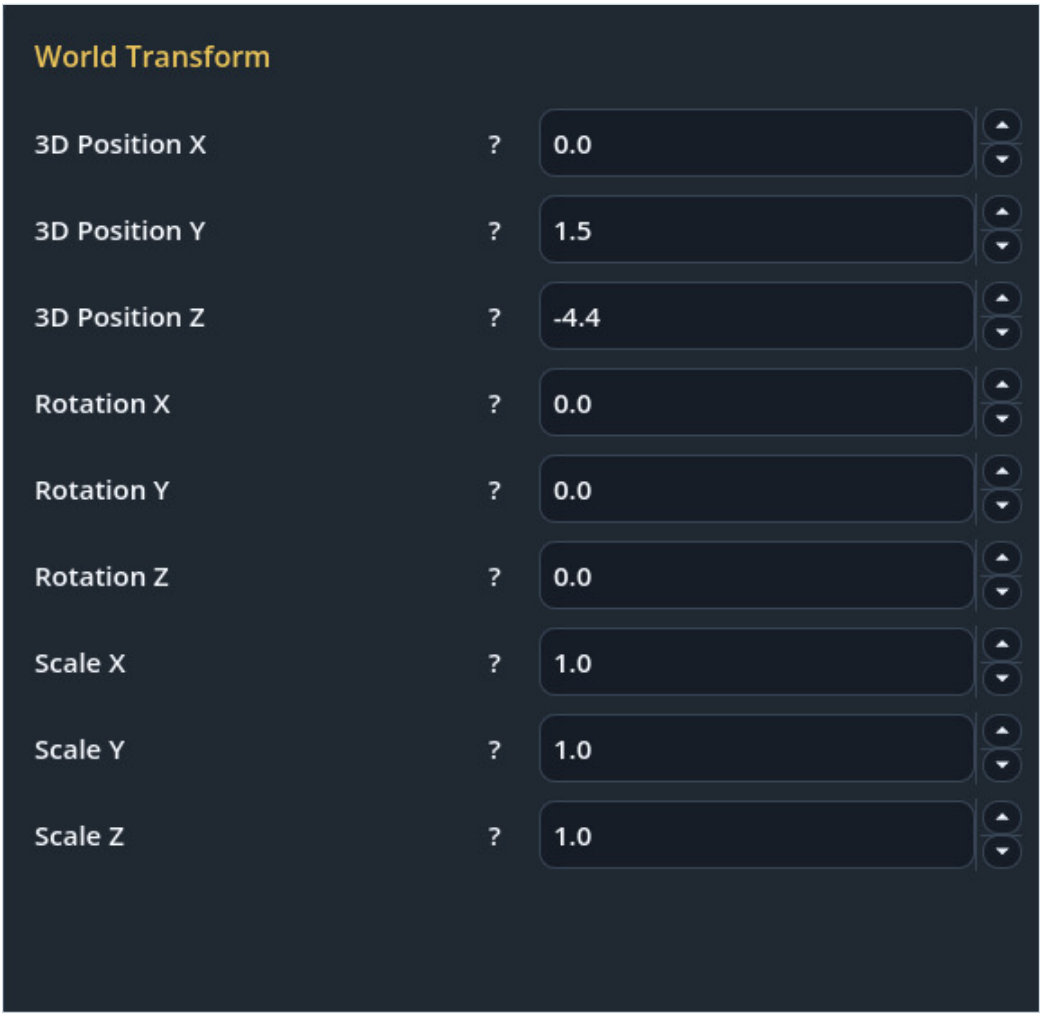


Figure 10. The back wall's centre is halfway up its height. Its bottom therefore meets Y 0.

The two side walls

Add two more boxes using the same process. Enter the following values. The names matter more than the order in which you add them.

Object	ID	Size X, Y, Z	Position X, Y, Z
Left Wall	wall_left	0.2, 3, 9	-5.9, 1.5, 0
Right Wall	wall_right	0.2, 3, 9	5.9, 1.5, 0

Leave both walls' Rotation at 0, 0, 0, Scale at 1, 1, 1, and Blocks Other Characters on. Apply each wall before selecting the next one.

You can Duplicate a wall to save work, then change the duplicate's ID, name, dimensions and position. Duplication makes a separate scene object, so check the new selection before entering values. For a first attempt, adding each box separately is perfectly reasonable.

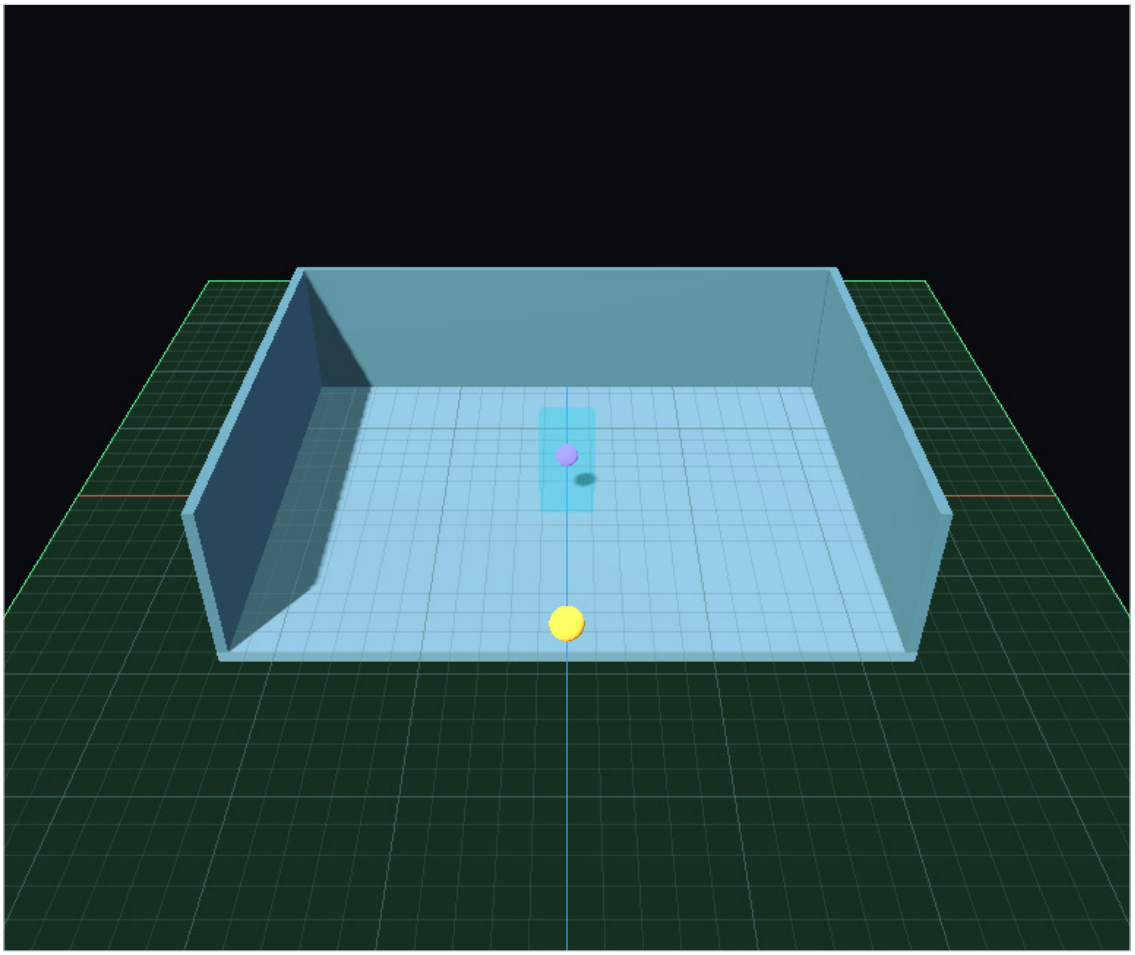


Figure 11. The floor and three walls form the room shell. The open front is deliberate.

Why the positions use halves

The wall is three units tall, so its centre is Y 1.5. The side walls are 0.2 thick, so their centres sit at X -5.9 and +5.9 within the floor's outer edges. The back wall's centre is at Z -4.4 for the same reason.

If a wall floats, check Position Y and Size Y together. If a wall is unexpectedly huge, check Scale as well as Size. A Size X of 12 with Scale X of 2 produces a wall twice as wide as intended.

7. Give the room its colours

Each primitive has its own material settings. Changing the floor does not automatically recolour all of the walls.

1. Select Floor.
2. Open **Visual Effects** in the Inspector.
3. Scroll down to **Additional Material Setting** and find **Material Color**. It is below the shader-layer controls in this version.
4. Click the colour swatch and choose a warm grey, or enter the example colour `#918b7c` in the picker's colour-code field.
5. Close the picker, click Apply Changes and save.

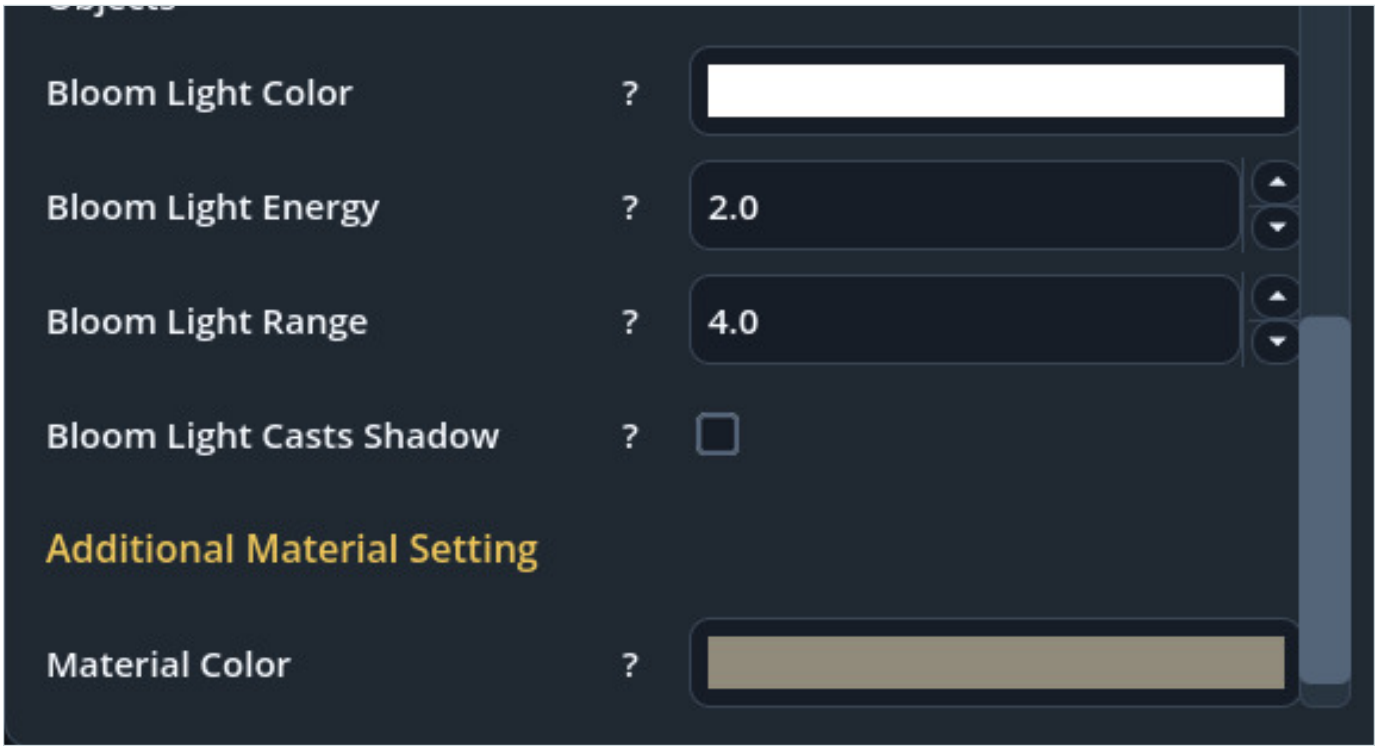


Figure 12. Material Color is in Visual Effects. Scroll past the shader controls to reach this setting.

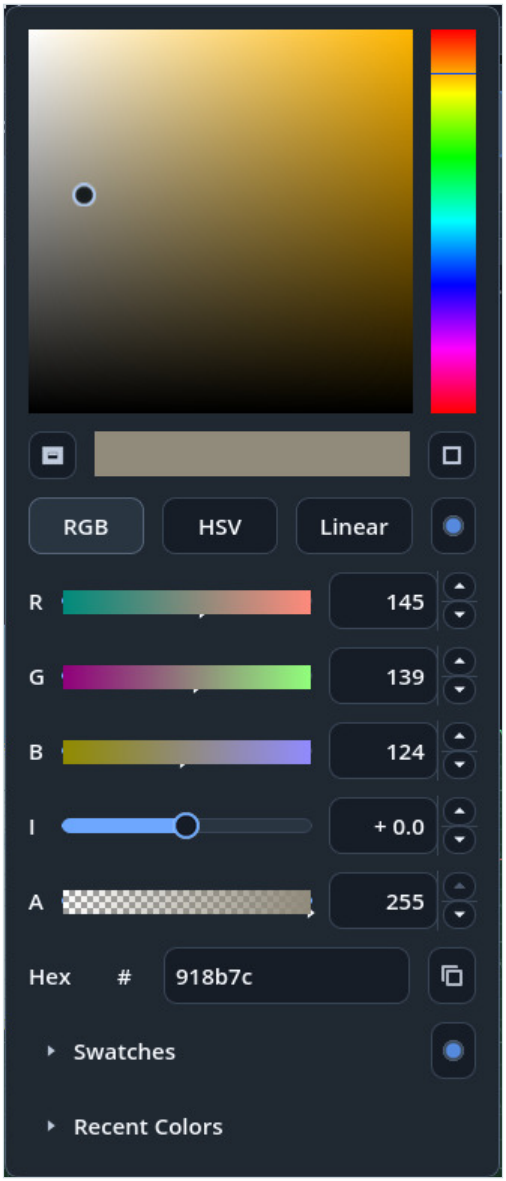


Figure 13. The colour picker lets you choose visually or enter a precise colour code. You can use a different palette and keep the same room dimensions.

Repeat those steps for Back Wall, Left Wall and Right Wall. The example wall colour is **#486d71**, a muted blue-green.

For ordinary painted surfaces, use Metallic **0** and Roughness about **0.85**. These settings are near the top of Visual Effects, under Material and Reflection. A high roughness gives a softer surface response; metallic is for metal-like material response rather than a general brightness control.

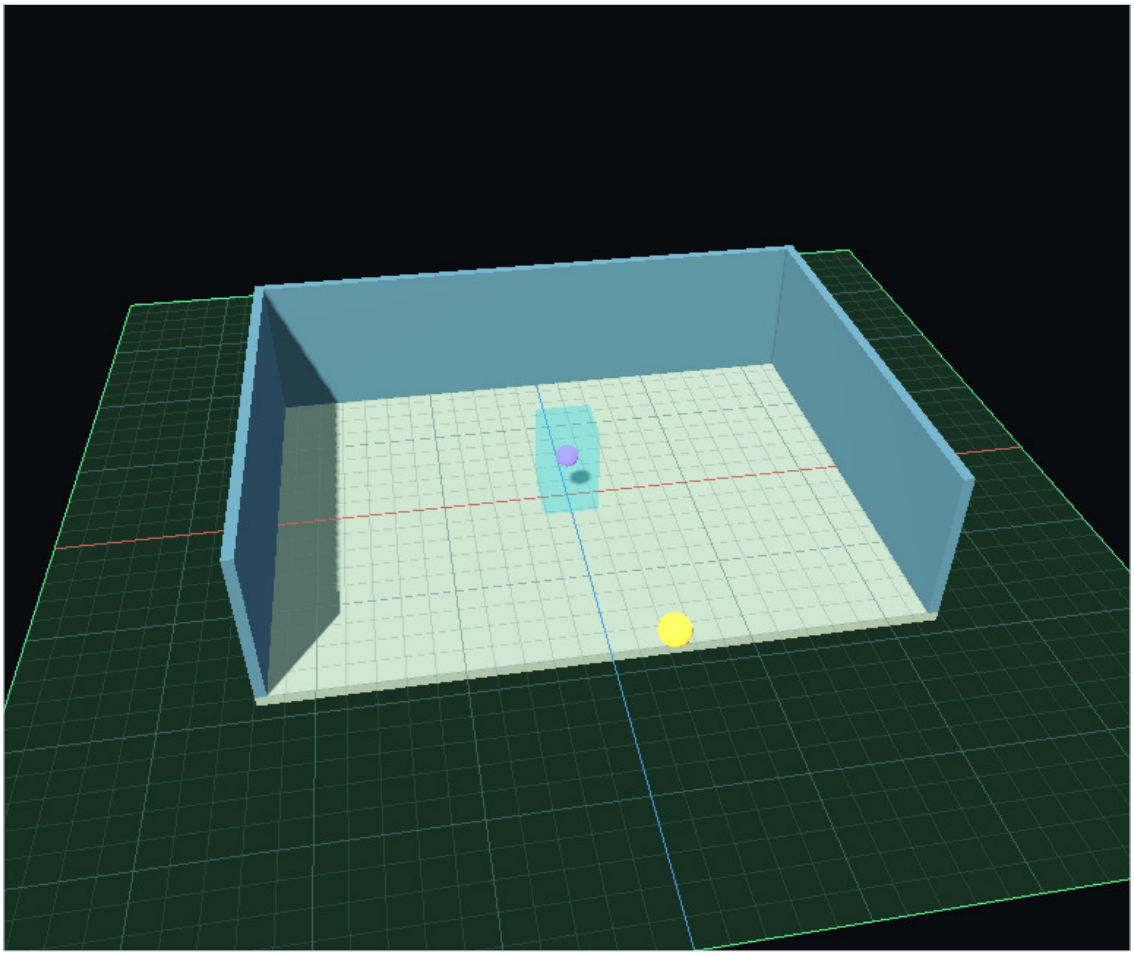


Figure 14. The same room shell with separate floor and wall colours.

Light changes how a colour looks. A warm light can make a grey surface appear warmer, and a shadowed wall will look darker than its colour swatch. Judge the material in a play test as well as in the editor's inspection lighting.

Colour, tint and shader effects

Material Color is the basic surface colour. Effect Tint belongs to the visual-effect stack. Model Tint belongs to a character's appearance. They are not interchangeable labels for one shared colour.

Start with no shader layers while learning material colour. Once the plain surface looks right, add an effect if you need it. This makes it easier to see what the effect actually changed.

8. Texture the floor or a wall

Use an imported **Image** asset as a primitive's Base Color Texture. A plain texture can give the same box a tiled, wooden, brick or painted-panel appearance without changing its geometry.

The pictures use a simple tiled pattern to make repetition easy to see. Choose any suitable image you already have permission to use. For a floor, a seamless or tileable image gives a better result than a photograph with an obvious edge.

Import your image

1. Open the Asset Designer and add an asset.
2. Set Kind to Image.
3. Give it ID `floor_tiles` and Name **Floor Tiles**.
4. Browse for your chosen floor image.
5. Apply Asset Changes and Save Project.
6. Return to the Scene Designer and select Floor.

If you already imported the image, use that record instead. Its ID and name can differ from the example. The important part is selecting the correct Image asset in the material field.

Assign it to the floor

1. Open Floor's **Visual Effects** tab.
2. At the top, find **Material and Reflection**.
3. Set **Base Color Texture** to your image, such as **Floor Tiles(floor_tiles)**.
4. Set **Texture Mapping** to **Triplanar** for this first test.
5. Start Texture Scale at **1**.
6. Leave **Tint Texture with Material Color** off.
7. Apply Changes and save.

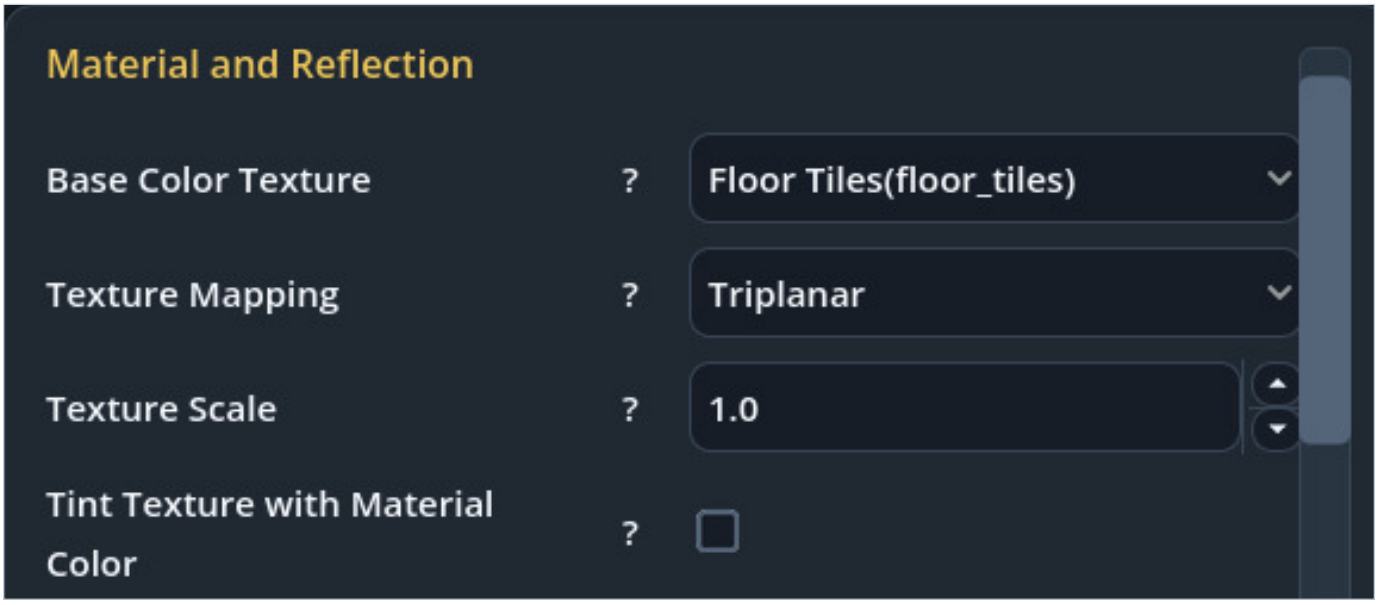


Figure 15. These are the primitive texture controls in the main Inspector. They are separate from model-file texture replacement in the Asset Designer.

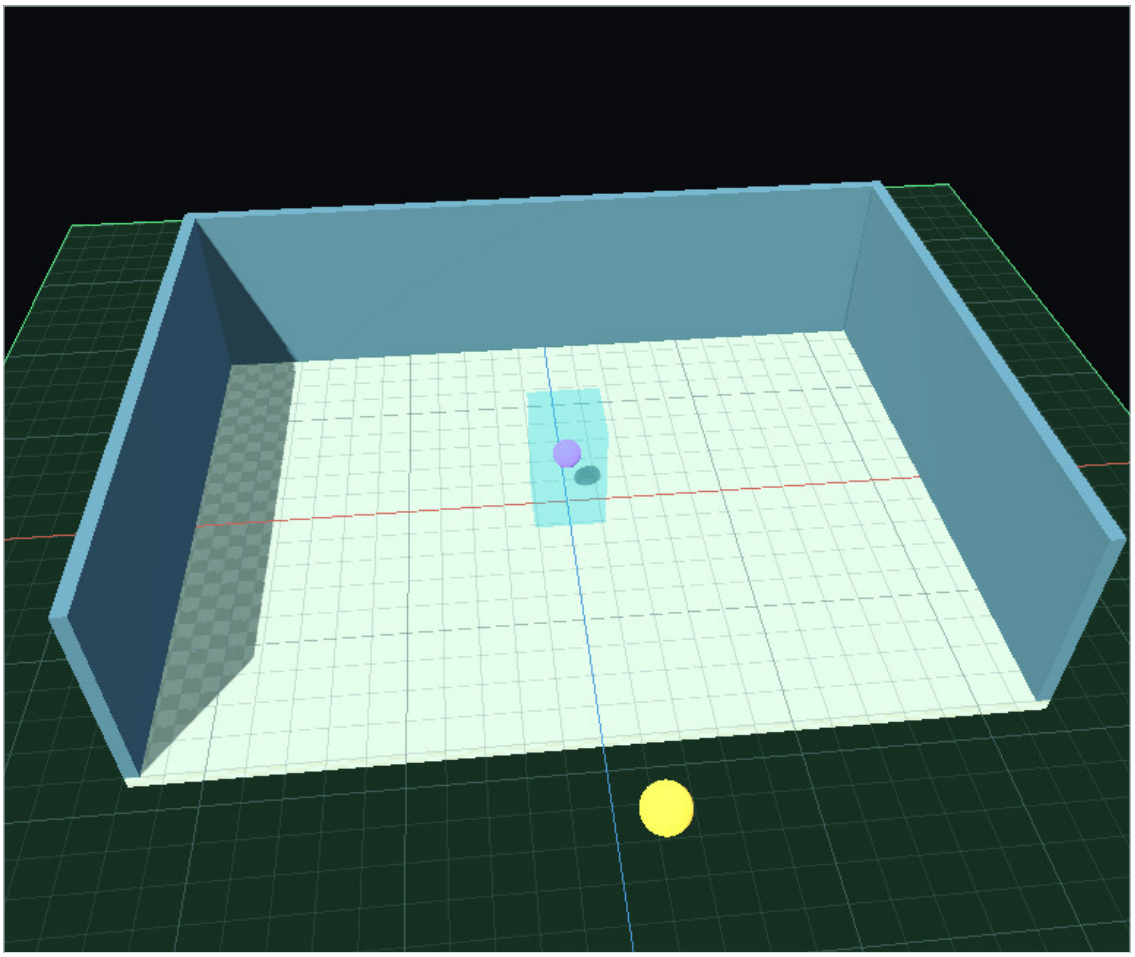


Figure 16. The Image asset supplies the floor pattern. The floor is still the same box with the same dimensions.

Control	What it changes
Base Color Texture	The Image asset placed on the generated geometry. None returns to the object's colours.
Model UV	Uses the primitive's own texture layout. A large, resized face can stretch the image.
Triplanar	Projects the image from three directions. Useful for repeated surfaces on floors, walls and generated parts.
Texture Scale	Changes repetition. A larger number makes the pattern repeat more often. It does not enlarge the room.
Tint Texture with Material Color	Multiplies the image by the object's material/generated colours. Turn it off to see the image's own colours.

Find a sensible repeat size

Try Texture Scale **0.5**, then **1** and **2**. Apply the change and look at the size of a tile beside the player. In the example, 0.5 gives a more spacious pattern than 1. Your source picture may contain a different number of tiles, so its best value can be different.

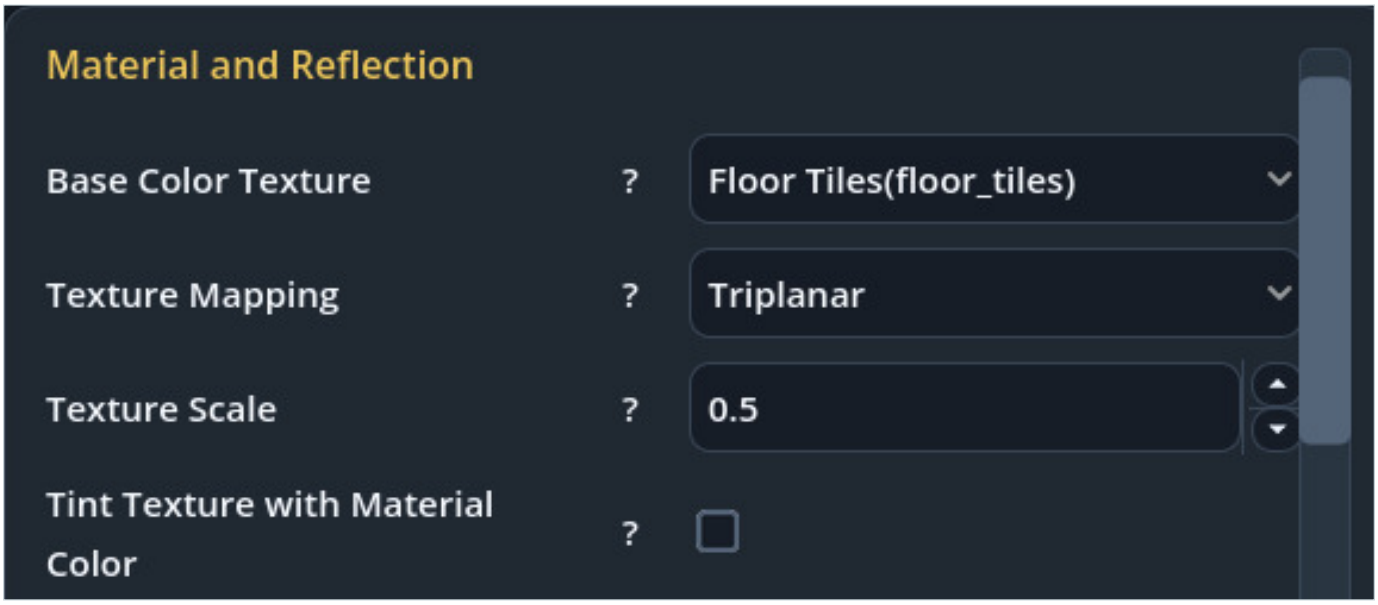


Figure 17. Change Texture Scale while keeping Size and Position unchanged. Judge repetition against the character and room.

The range is 0.01 to 100. Triplanar uses local model units, so an object's transform scale can also affect how the pattern reads. Keeping the construction Scale values at 1 makes the result easier to predict.

Apply a texture to a wall

Select Back Wall and repeat the same Visual Effects steps. Choose a wall image, use Triplanar, adjust Texture Scale, then Apply and Save. Select each side wall separately if it should use the same material.

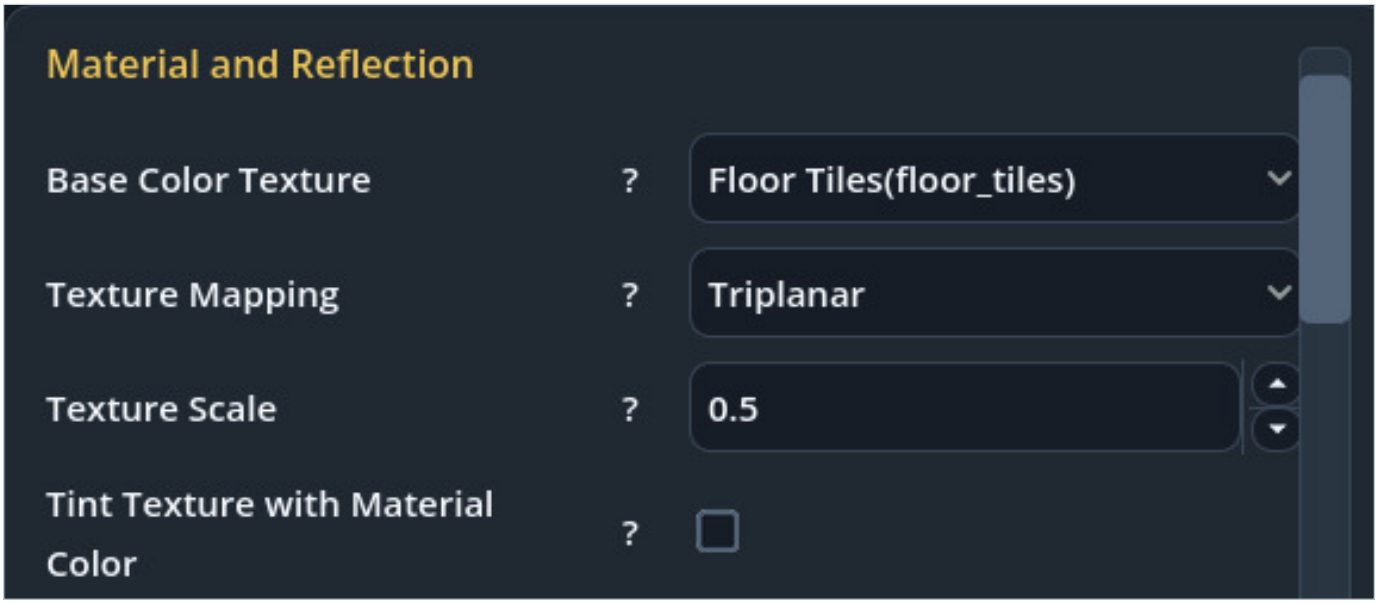


Figure 18. The same material controls work on a wall. This comparison deliberately uses the tile image so that the mapping is easy to recognise.

You can use a different Image asset for the wall, or choose None to return it to a solid colour. A texture assignment is per object. It is not a global paint operation for every wall in the scene.

If the image looks oddly blue or brown, check Tint Texture with Material Color. If it is absent from the selector, return to the Asset Designer and confirm Kind is Image and the asset changes were applied.

9. Add real room lighting

The Scene Designer uses inspection lighting to help you see objects while editing. A room can look visible there and still need its own light for gameplay. Add an authored light and test the game view.

1. In Add to Scene, choose **Directional Light**.
2. Select the new light in Scene Objects.
3. Set its ID to `key_light` and Name to **Room Key Light**.
4. Open **Light Settings**.
5. Set Light Kind to Directional, turn Light Starts On on, choose Light Color **#fff1d8**, and set Intensity to **0.7**.
6. Leave Cast Shadows on.
7. In Visual Adjustments, enter Position **0, 5, 1** and Rotation **-0.96, -0.59, 0**.
8. Apply and save.

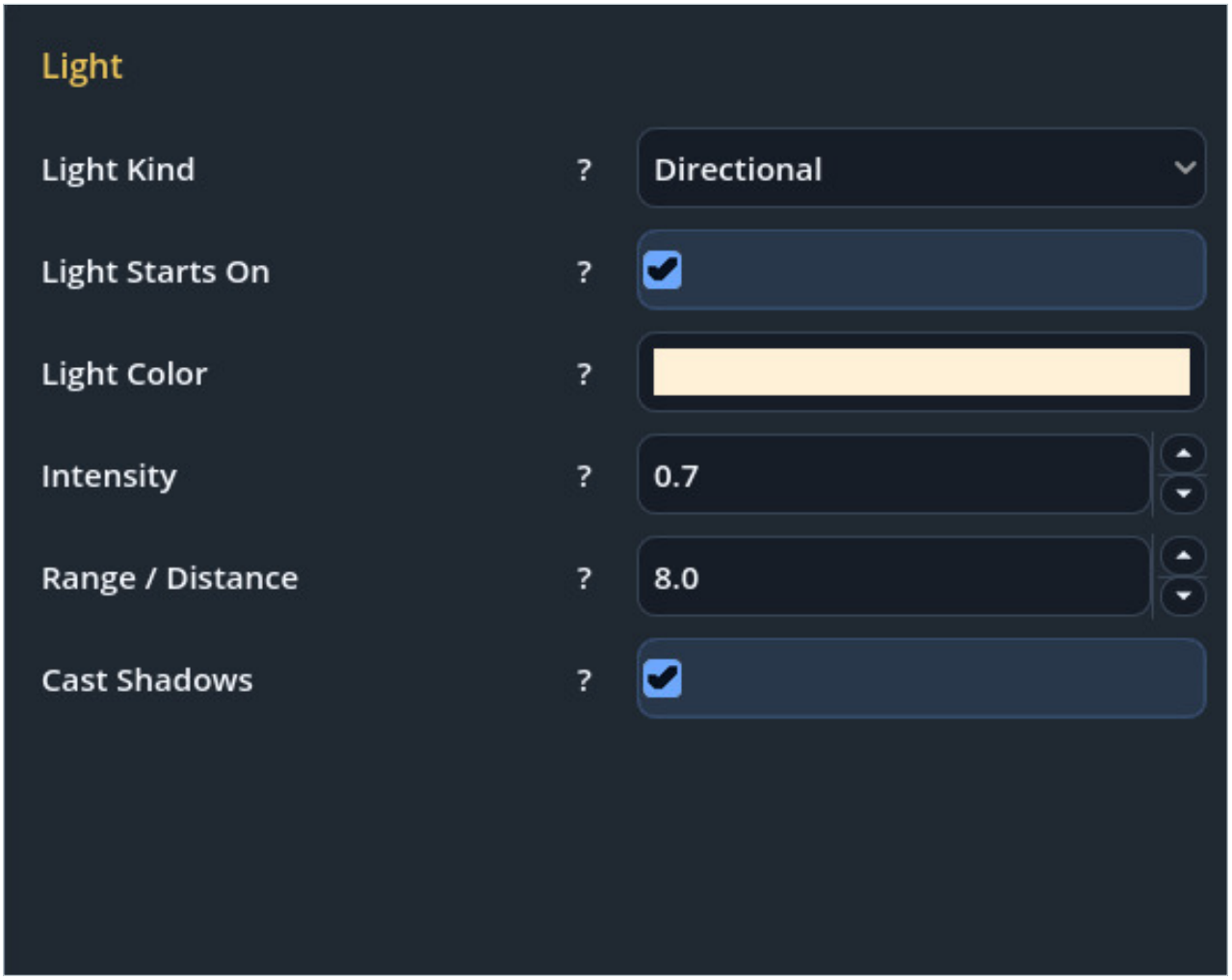


Figure 19. An authored light belongs to the room and is used by the game. Its settings are different from the editor's temporary inspection lights.

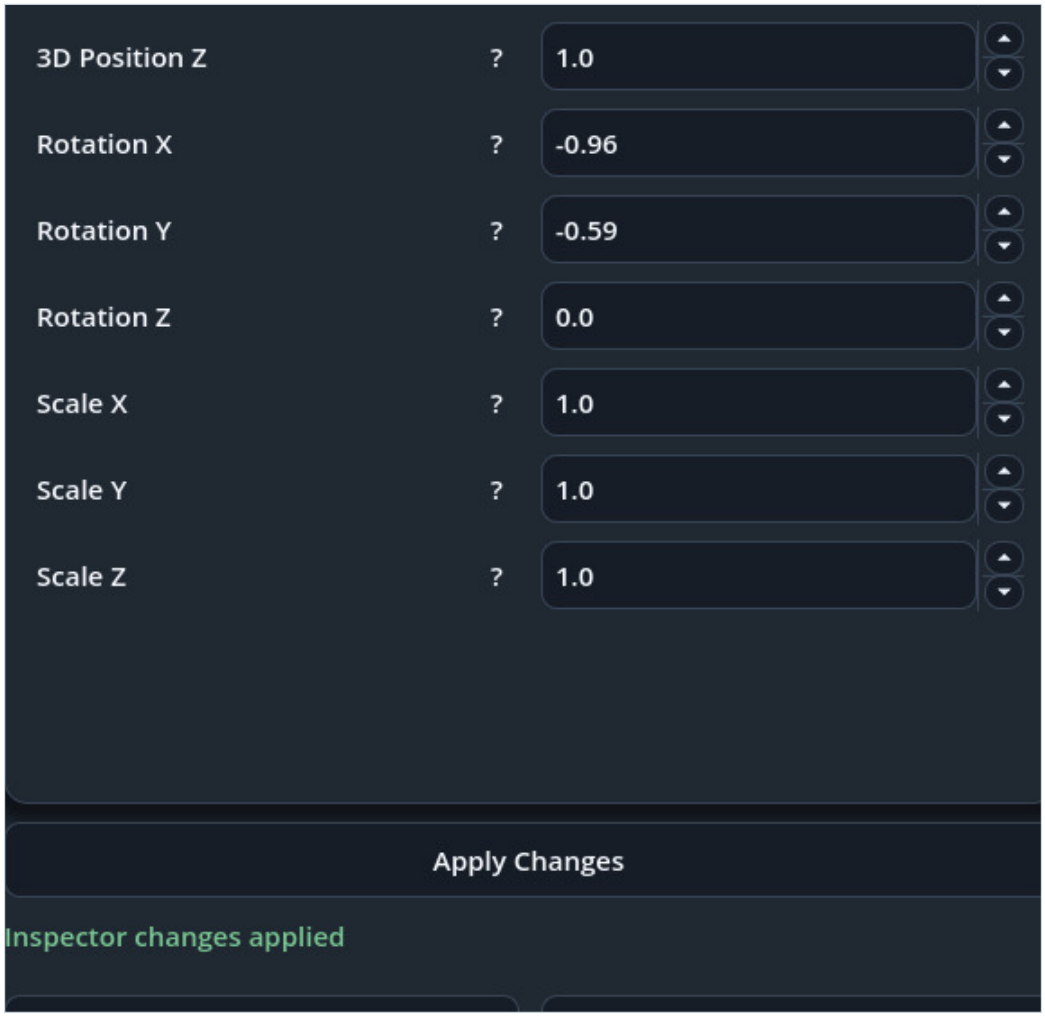


Figure 20. This directional light's Rotation values are radians. They aim the light down into the open room.

A directional light supplies parallel rays, like sunlight. Its rotation determines the direction; its position is mainly useful for organising and selecting the marker. A point light has a position and range, making it more suitable for a lamp in a corner. A spot light also has an aim and cone angle.

Leave advanced lens flares, god rays, fog and post-process effects alone until the basic room is lit. Their availability and appearance can also depend on the renderer. This room needs only straightforward lighting to make the geometry and actors readable.

10. Set the walkable area

The visible floor does not tell the navigation system its limits. We need a walkable rectangle sitting inside the walls. This is the answer when a floor looks large enough but the player stops at an invisible boundary.

1. Expand **Navigation Areas** in Scene Objects.
2. Select **Navigation Area 1**.
3. Open its **Shape** tab.
4. Leave Shape as **Bounds Rectangle**.
5. Set the four bounds as follows.
6. Apply Changes and save.

Shape field	Value
Minimum X	-5.6
Maximum X	5.6
Minimum Z	-4.1
Maximum Z	4.1

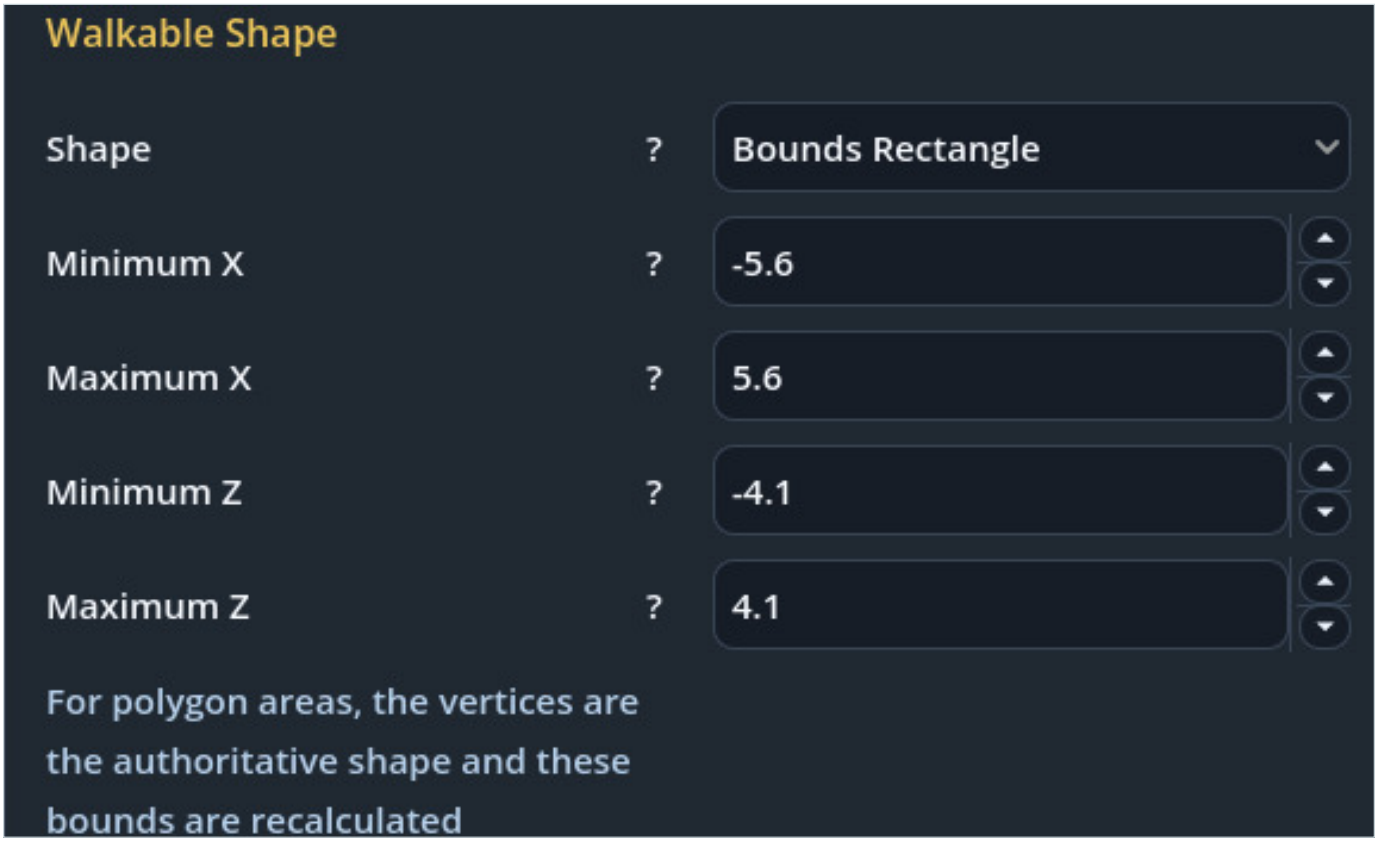


Figure 21. The navigation rectangle uses its own bounds. Enlarging the visible floor or changing grid spacing does not change these fields.

The starter's flat navigation is at walking height Y 0. Our floor's upper face was deliberately placed at the same height. The small inset from the walls avoids asking a character's centre to walk directly onto an outer edge.

The character also has a body radius. Navigation must leave clearance for that body, so the centre may stop short of the rectangle's exact numeric edge. Do not treat a sensible clearance gap as proof that the floor failed to resize.

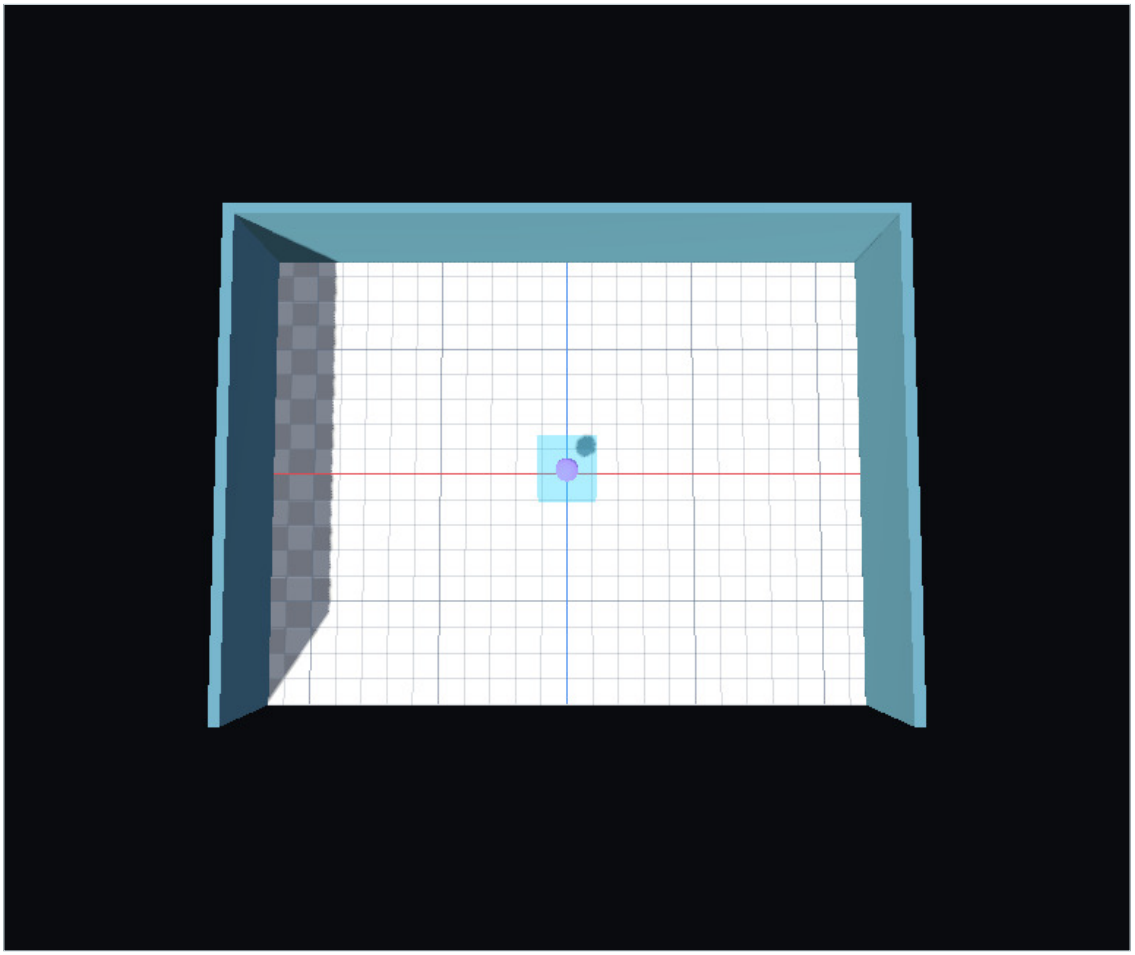


Figure 22. Top view is useful for comparing the walkable area with the room shell. Navigation overlays are editor aids.

Keep these four sizes separate

Size or boundary	Its job
Primitive Size	The visible shape of the floor or wall.
Navigation Area bounds	The region used to build walkable navigation.
Character Collision Radius	Clearance for the actor's body.
Grid/Snap spacing	An editor measuring and movement aid.

If you later enlarge the room, update the architecture and navigation together. Camera framing may need another adjustment too.

More complicated floor plans

For an L-shaped room, select a navigation area, use **Navigation > Convert Bounds to Polygon**, then **Edit Navigation**. Work in Top view. Drag vertices to reshape the outline; double-click an edge to add a vertex where supported by the editing tool.

The Navigation menu also offers Add Vertex, Delete Selected Vertex, Reverse Polygon Winding and Reset to Bounds Rectangle. Multiple enabled navigation areas can form a connected surface when they overlap. Separate islands still need a valid route between them; two distant rectangles do not create a bridge.

For this workshop, keep the simple rectangle. It is easier to check before introducing more complicated navigation.

11. Blocking, colliders and event volumes

For the flat workshop, turn **Blocks Other Characters** off on Floor and on for the walls and solid furniture. The flat navigation surface already handles where the character can walk. A floor accidentally treated as an obstacle can obstruct the whole room.

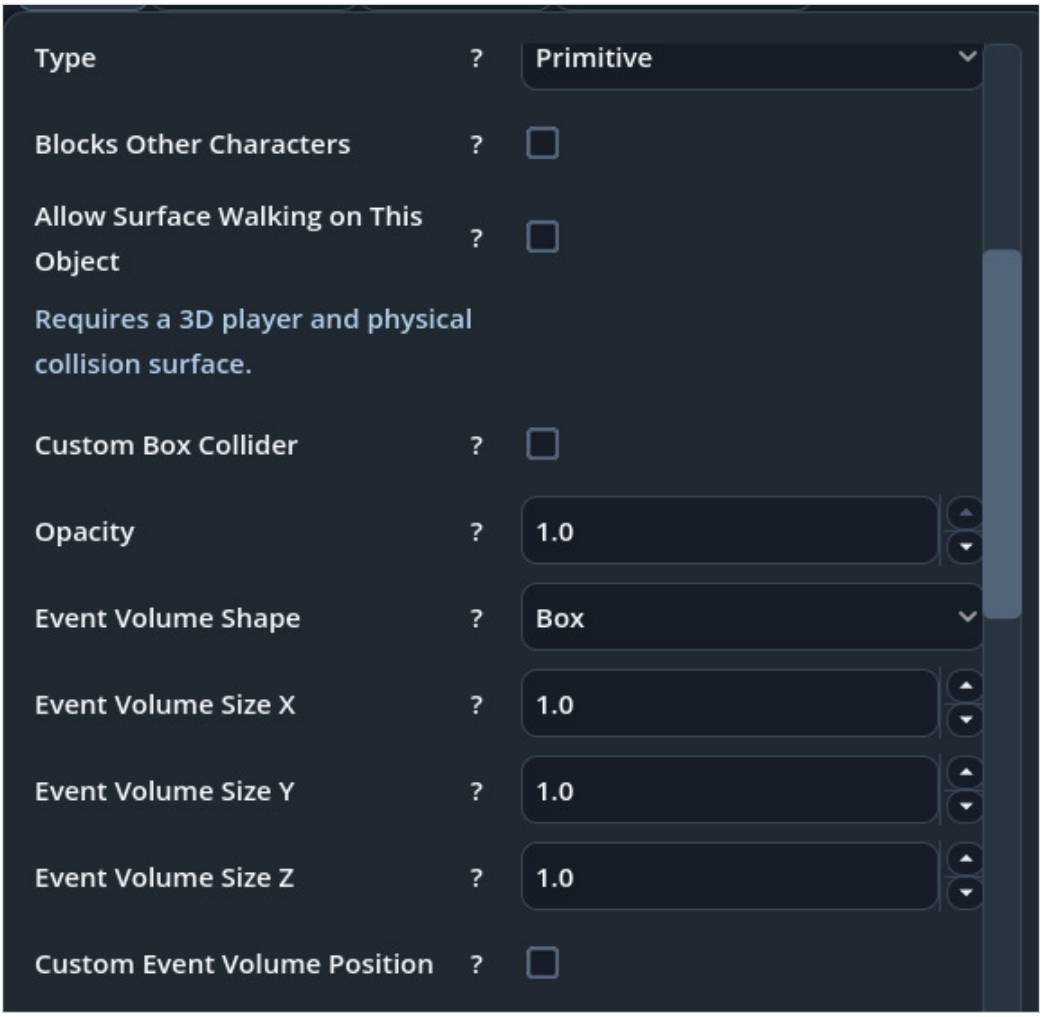


Figure 23. Floor's blocking switch is off for this flat-room setup. Walls and furniture have their own blocking settings.

A collider is the physical shape

Some objects need a collision box different from their visible model. In General, turn on **Custom Box Collider**, then edit **Collider Size** and **Collider Position**. These fields are especially useful for an imported door or a prop whose automatic bounds are too generous.



Figure 24. Custom Box Collider exposes a separate physical box. Its position is an offset relative to the object.

For a primitive box, a matching collider normally uses the primitive's size and a local position of 0, 0, 0. A changed collider should still cover the solid part of the object. Making it too small lets the character clip through the visible surface.

Event Volume controls where entry, exit or similar event tests happen. It is not a universal replacement for the physical collider. **Stop Distance** controls an interaction approach. It does not shrink an obstacle either.

Problem	The setting to inspect
Player is blocked far away from a door	Blocking, the collider's size/offset, character radius and nearby navigation obstacles.
An entry event fires too early	Event Volume shape, size and position.
A Talk/Use approach stops farther away than desired	Walk-to-Interact behaviour and Stop Distance, after checking that a collider is not preventing the approach.
Actor can walk through a solid object	Blocks Other Characters and the physical collider.

Keep gravity and surface-walking experiments out of this first flat-room lesson. If you later enable physical gravity, stairs or walking on marked surfaces, the collision and movement requirements need another deliberate check.

12. Add a workbench with a generator

Primitives are good building blocks. A procedural generator is a shortcut for a recognisable prop assembled from generated parts.

1. Choose **Add to Scene > Procedural > Table**.
2. Select the table in Scene Objects.
3. Set ID to `workbench` and Name to **Workbench**.
4. Leave Blocks Other Characters on.
5. In Visual Adjustments, put it at **0, 0, -2.5** with Rotation 0, 0, 0 and Scale 1, 1, 1.
6. Use a brown Material Color, such as **#8b5832**.
7. Apply and save.

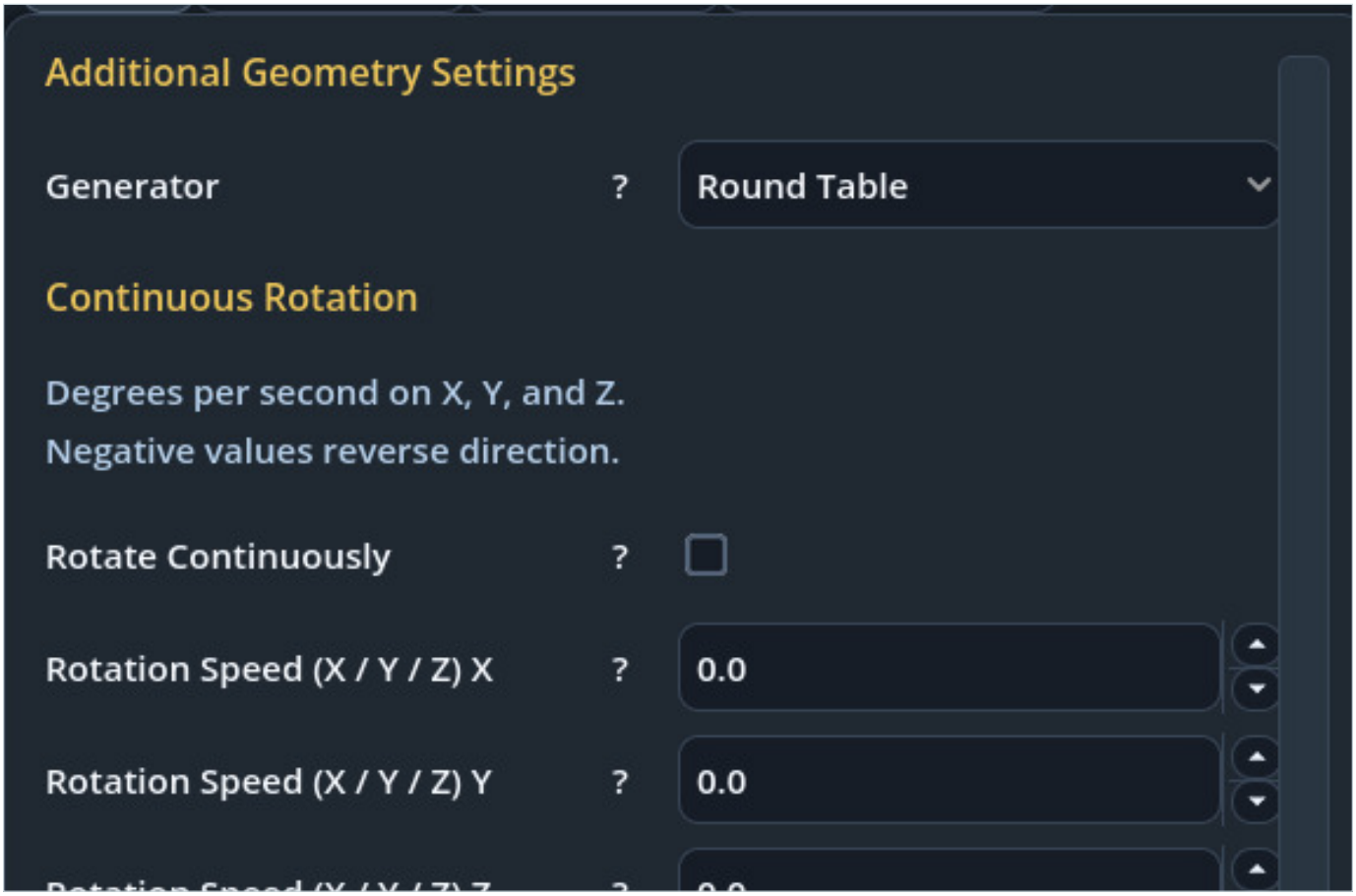


Figure 25. A procedural table supplies several generated parts without requiring you to assemble every leg by hand.

The generator's origin and finished dimensions need not match the centre convention of a simple Box. Inspect its feet in Front or Right view. Do not automatically give a generated table the floor's Position Y of -0.1.

You can texture generated parts with the same Base Color Texture controls used for primitives. One image can be projected onto all of the generated parts. That is useful for a tiled material; detailed lettering or carefully painted furniture may be better handled by an imported model with its own UV layout.

13. Create an NPC definition

An NPC is a character the player does not control. Start with its reusable definition, then place an instance in the room. Naming a box Mara does not give it character behaviour.

Make Mara, the workshop keeper

1. Open the **Character Designer**.
2. Click **Add New**.
3. On the Character identity page, set Character ID to `mara`, Name to **Mara**, and Description to **The workshop keeper.**
4. Leave Model / Asset empty for this first version. The built-in placeholder is enough to test placement and interaction.
5. Open **Character Settings**. Set Display Name to Mara, Role to **Non-Player Character (NPC)**, and leave Character Enabled on.
6. Keep an ordinary standing/default state while building the room. Set Model Height to about **1.8** and Collision Radius to **0.35** if you are adjusting the movement defaults.
7. Open **Player Settings**. Leave Playable Character and Default Starting Character off.
8. Apply the character changes and save.

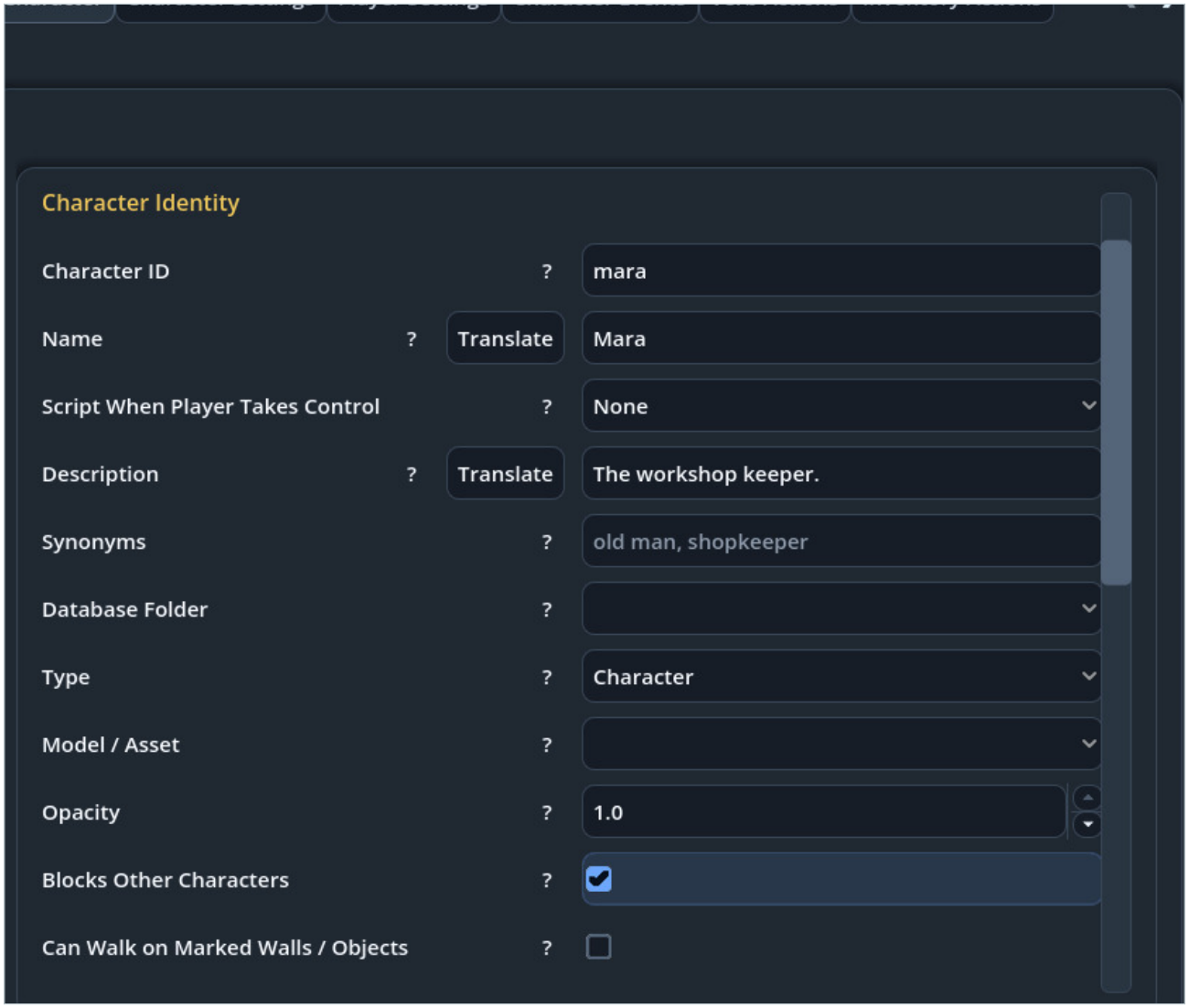


Figure 26. Mara's reusable Character definition. A source model can be assigned later without rebuilding the room layout.

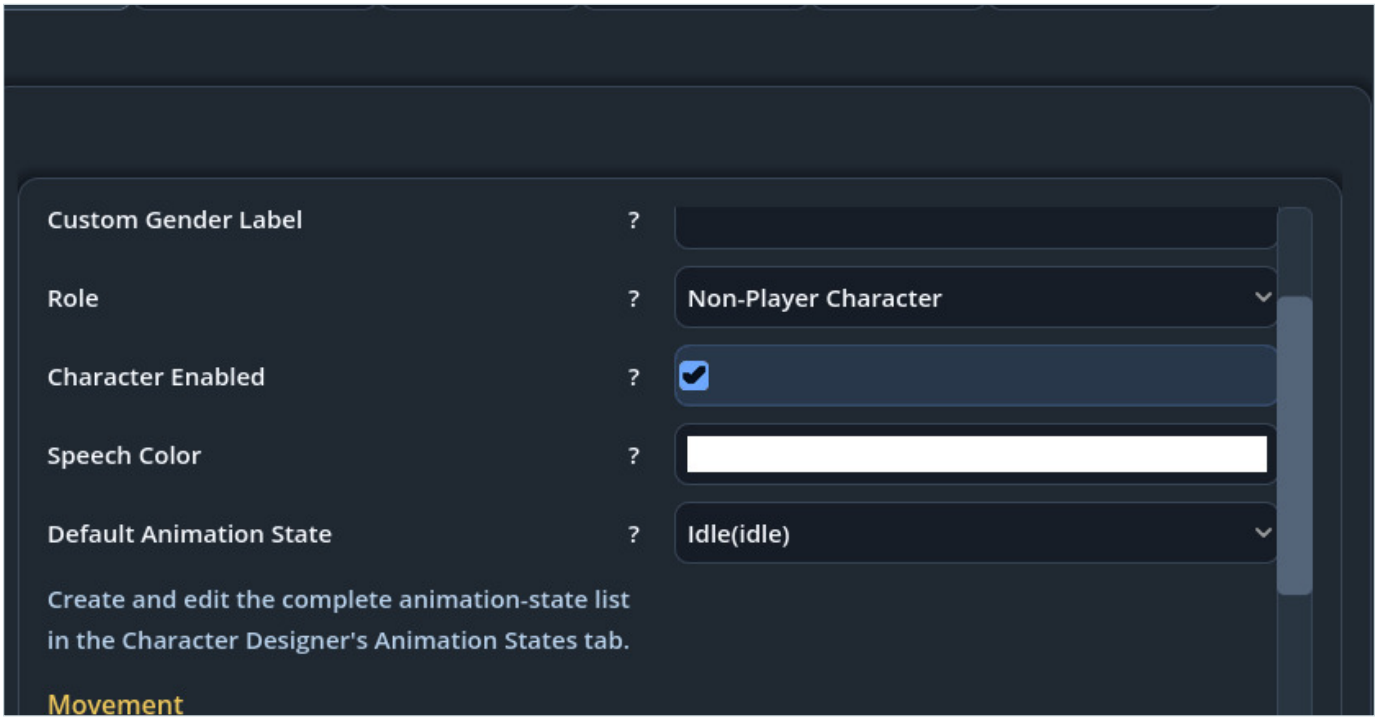


Figure 27. Role and display name belong to the reusable character's Character Settings.

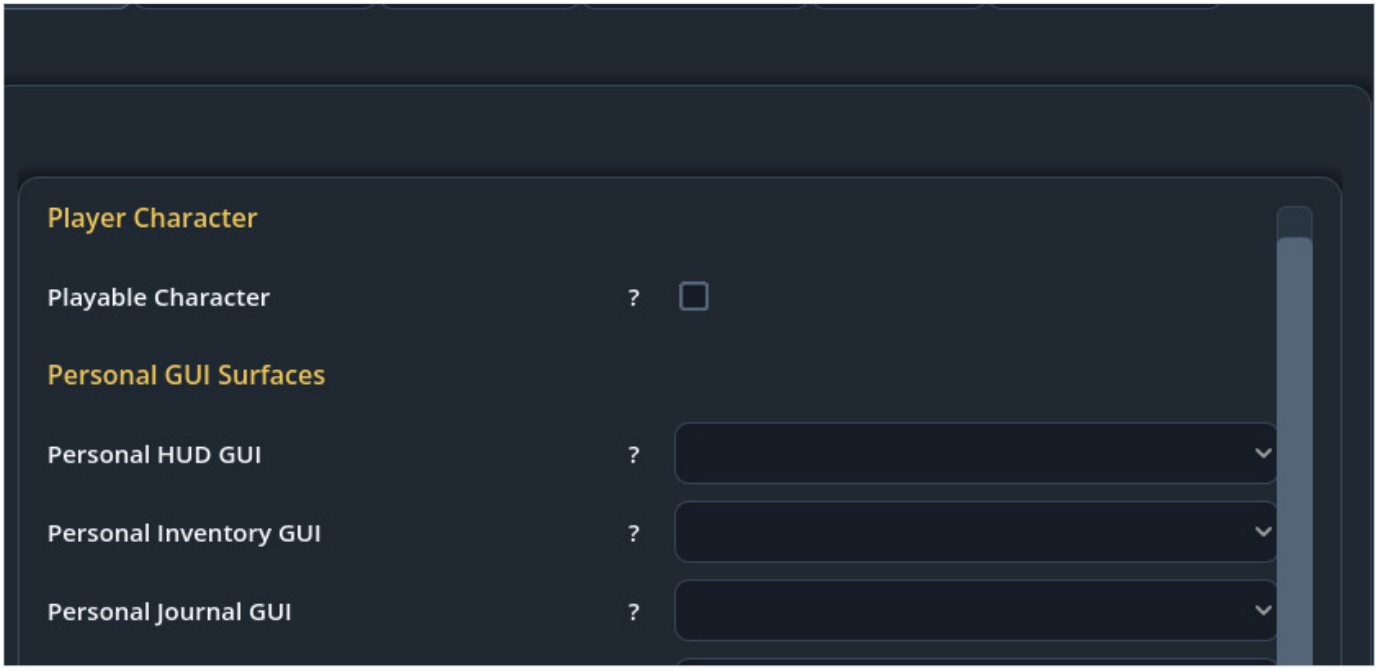


Figure 28. An ordinary NPC is not selected as a playable or default starting character.

The placeholder is a layout and gameplay stand-in. It is not meant to be the final animated person. If you already have an imported character model, choose it in Model / Asset and map its actual idle, walk and talk clips in Animation States. An imported model asset alone does not make those state assignments.

Definition versus placement

The definition holds reusable identity, movement and character behaviour. A room instance has a position in that particular room and can have appearance or other overrides. Use the definition for changes which should follow Mara wherever she appears. Use the room instance for where she stands in this workshop.

14. Put the NPC in the room

1. Return to the Scene Designer and select The Small Workshop.
2. Choose **Add to Scene > Character**.
3. In the character chooser, select **Mara(mara)** and confirm.
4. Place her in the room when the placement tool is active.
5. Select Mara in Scene Objects and open **Visual Adjustments**.
6. Set 3D Position to **2.2, 0, -0.8**, Rotation to 0, 0, 0 and Scale to 1, 1, 1.
7. Apply and save.

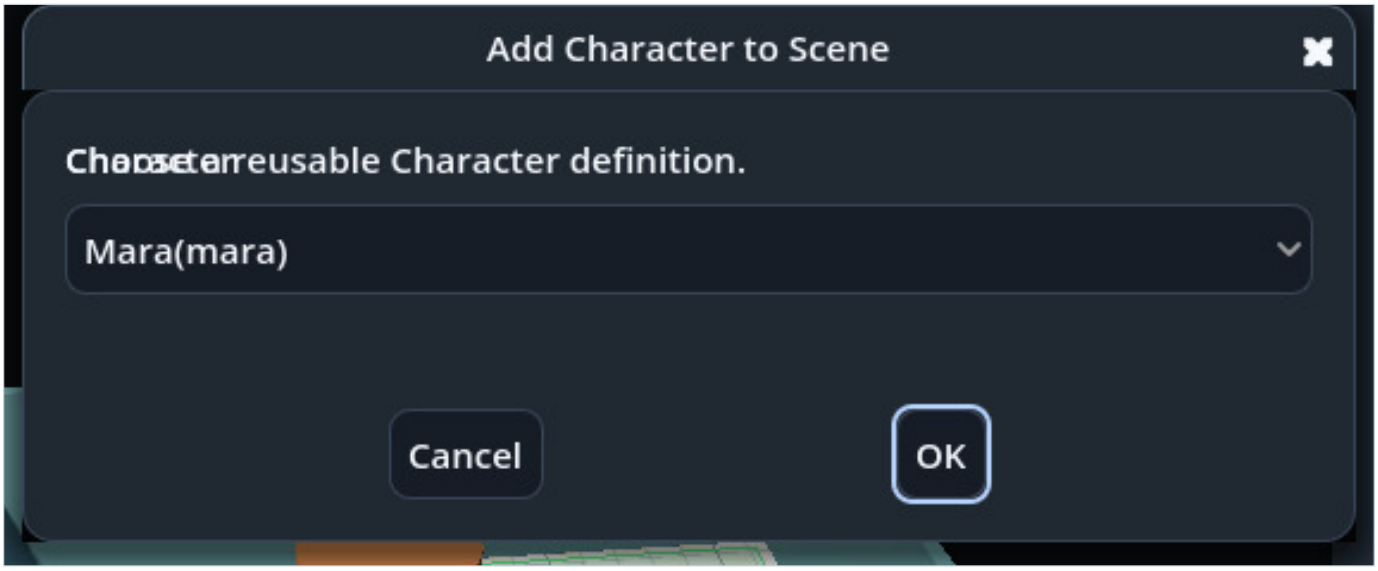


Figure 29. The chooser lists reusable characters that are available for placement in this room.

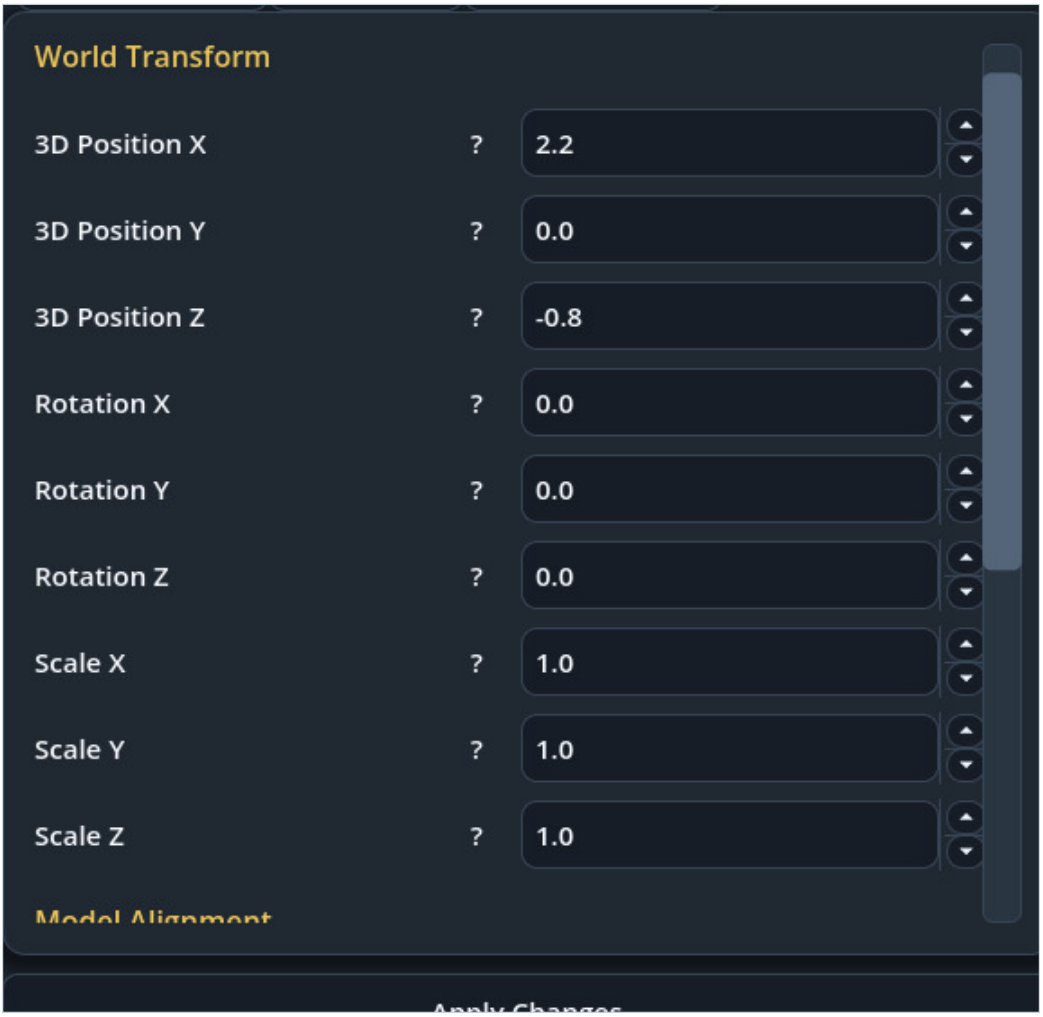


Figure 30. The room instance supplies Mara's workshop position. Her feet belong at Y 0 in this flat-room setup.

If Mara is missing from the chooser, check that her definition was applied and saved. The chooser excludes a reusable character already placed in the same room. Look under Characters in Scene Objects before creating another definition.

Scene character instances cannot simply be duplicated like boxes. To make another distinct person, create or clone a reusable Character definition and give it a different ID, then place that character.

Try a character tint

Select Mara's room instance and open **Visual Effects**. Set **Model Tint** to **#e8aa54**, then Apply and Save. Tint multiplies the model's existing colour. The built-in placeholder is blue, so this warm tint gives it a dark green appearance in the example; it does not paint the capsule a flat amber colour.



Figure 31. Model Tint changes the character's appearance in this scene. It is different from a primitive's Material Color field.

A white tint leaves the original model colours available. An imported textured character may respond differently because its source colours differ from the placeholder. Adjust the tint while checking the actual result, and use white if you want to keep the original artwork.

15. Place and identify the player

The starter player is already in the room. Select **Player** under Characters in Scene Objects and move that existing placement.

1. Open Visual Adjustments.
2. Set 3D Position to **-2, 0, 2**.
3. Leave Rotation at 0, 0, 0 and Scale at 1, 1, 1.
4. Apply Changes.
5. In Visual Effects, set Model Tint to **#79c8e8** for the cool blue result used in the screenshots. Leave it white to keep the original appearance.
6. Apply again and Save Project.

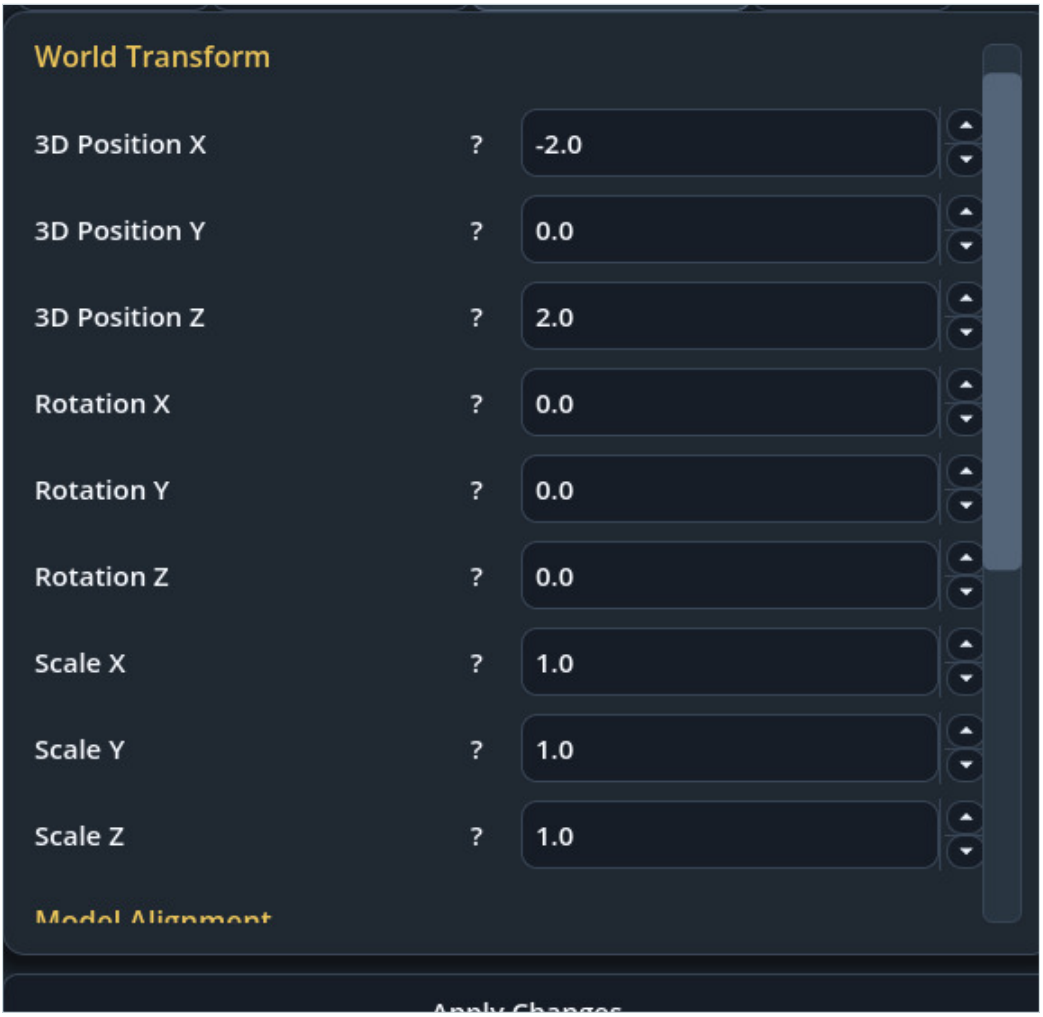


Figure 32. The player's scene position is its starting location for this simple setup. Move the existing starter player rather than making an extra copy.

If you removed the starter placement

Keep or restore the reusable **Player(player)** definition in the Character Designer. In the Scene Designer, use **Add to Scene > Character**, choose Player, place it, and enter the position above. A definition can exist in the project even when its placement was removed from a room.

Check the player flags

Open the player's **Player Settings**. **Playable Character** should be on. **Default Starting Character (Only One)** should also be on for this one-player lesson. Mara's equivalent starting flag should be off.

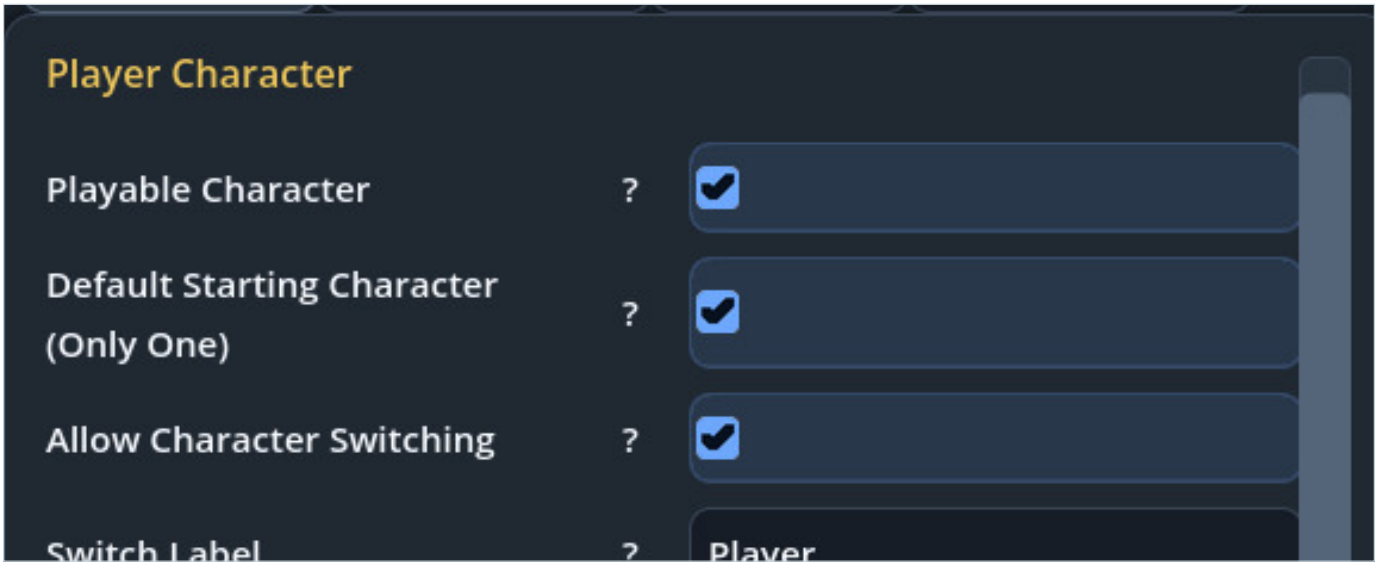


Figure 33. The player has a different job from the NPC even though both use reusable Character definitions.

Allow Character Switching is for games that change between playable actors. It is not needed to make Mara speak or stand in the room. Leave the project with one obvious starting player while checking this lesson.

16. Choose the project's starting player

Room placement and project startup should agree. The safest final check is in the project settings.

1. Open **Project** from the Adventure Database or project workspace.
2. In General, find **On Game Startup**.
3. Set **Startup Action** to **Start in Player Character Location**.
4. Set **Player Character** to **Player(player)**.
5. Apply Changes and Save Project.

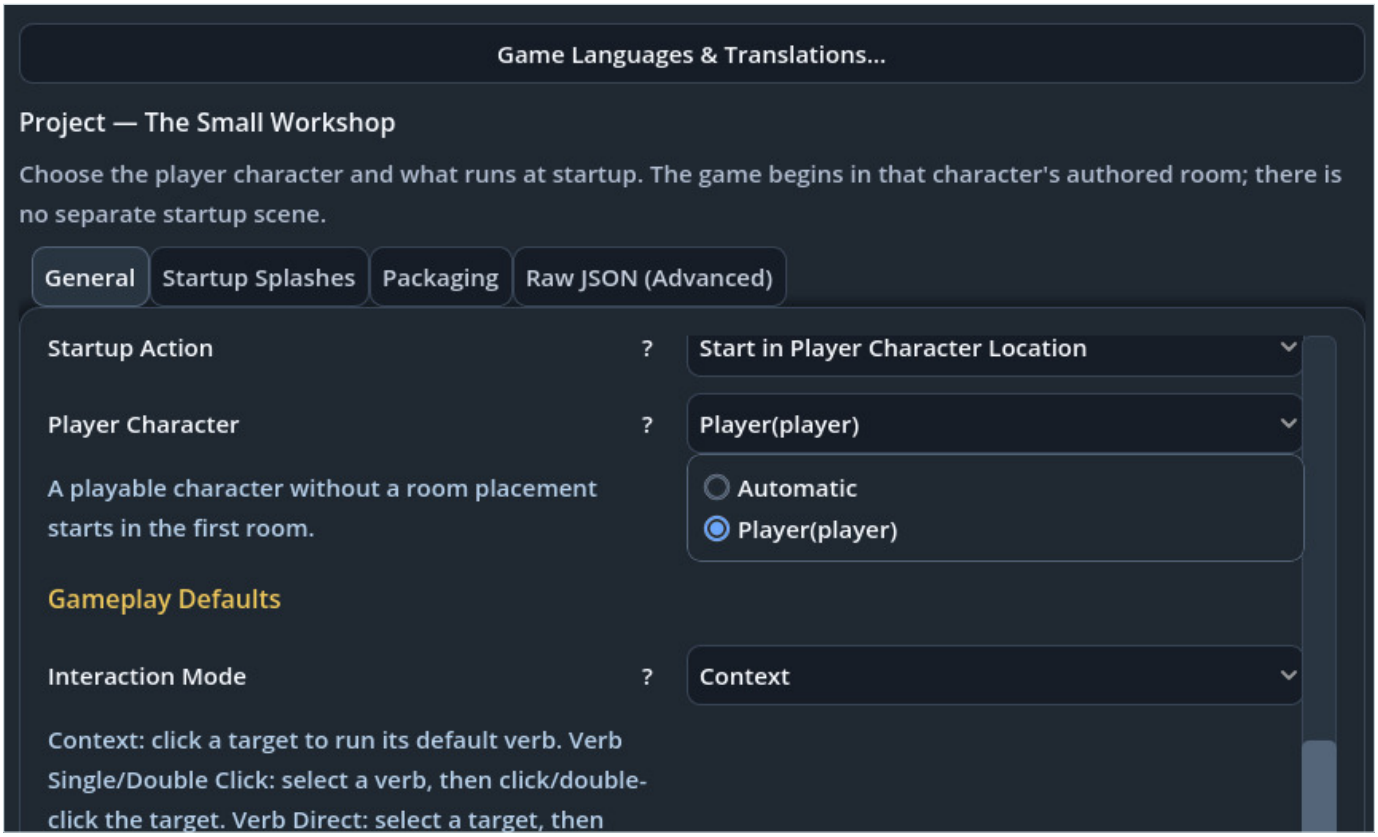


Figure 34. Project startup chooses the player whose room location should begin the game.

The normal Play / Restart button uses project startup. **Test Scene** runs the room currently open in the Scene Designer. Both are useful, but they answer different questions. A room can pass Test Scene while the project's startup action sends the full game somewhere else.

For this new project, the player is placed in The Small Workshop, so starting at the player location should open that room. A playable definition without a room placement can fall back to the first room; an explicit placement is clearer.

17. Set up the room camera

Use the existing starter camera for the first shot. There is no need to add several cameras just to see the room.

1. Expand **Cameras** in Scene Objects and select the starter camera.
2. In General, set Camera ID to `wide` and Camera Name to **Room View**.
3. Keep Projection as Perspective and Enabled at Runtime on.
4. Apply Changes.
5. Open **Position & View** and enter the values below.
6. Apply and save.

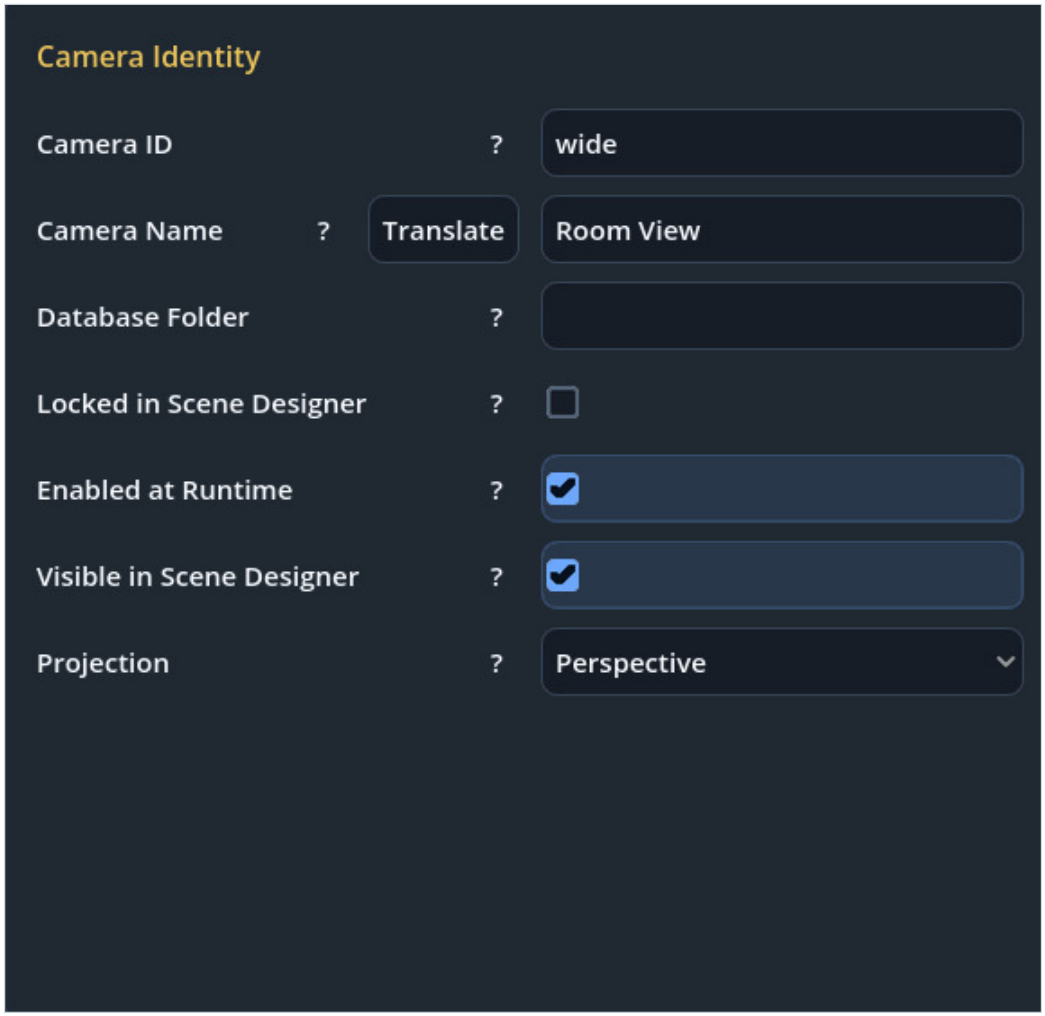


Figure 35. Give the camera a recognisable name and ID. This is the authored game camera, not the editor's orbiting inspection view.

Camera setting	X	Y	Z
Position	0	8.5	14
Look Target	0	0.8	-0.3

Set **Field of View** to **48**, **Near Clip** to about **0.05** (the field may show **0.051**) and **Far Clip** to **100**. Orthographic Size is not used by this perspective shot.



Figure 36. Position locates the camera. Look Target is a point in the room that it aims toward.

The camera sits outside the open front of the room and looks down toward its centre. There is no ceiling or front wall in its way. If you later add either one, inspect the shot again rather than assuming the camera can see through it.

What the camera values mean

Field of View changes the viewing angle. A wider angle can show more of the room but also changes the sense of perspective. Position changes the physical viewpoint. Look Target changes the aim. These are different ways of altering a shot.

Near and Far Clip determine the visible depth range. A wall too close to the camera can be clipped, and a Far Clip that is too short can hide distant objects. Neither setting decides where the player may walk.

Use Perspective for this lesson. An orthographic camera removes perspective size changes with distance and uses Orthographic Size to frame the view. It is a separate visual choice, not a fix for a poorly aimed perspective camera.

18. Make it the default camera

Creating or editing a camera does not always make it the room's selected opening shot. Set the default explicitly.

1. Select the **room name** in Scene Objects.
2. Open **Camera & View** in the Inspector.
3. Find **Default Camera**.
4. Choose **Room View(wide)**.
5. Apply Changes and Save Project.

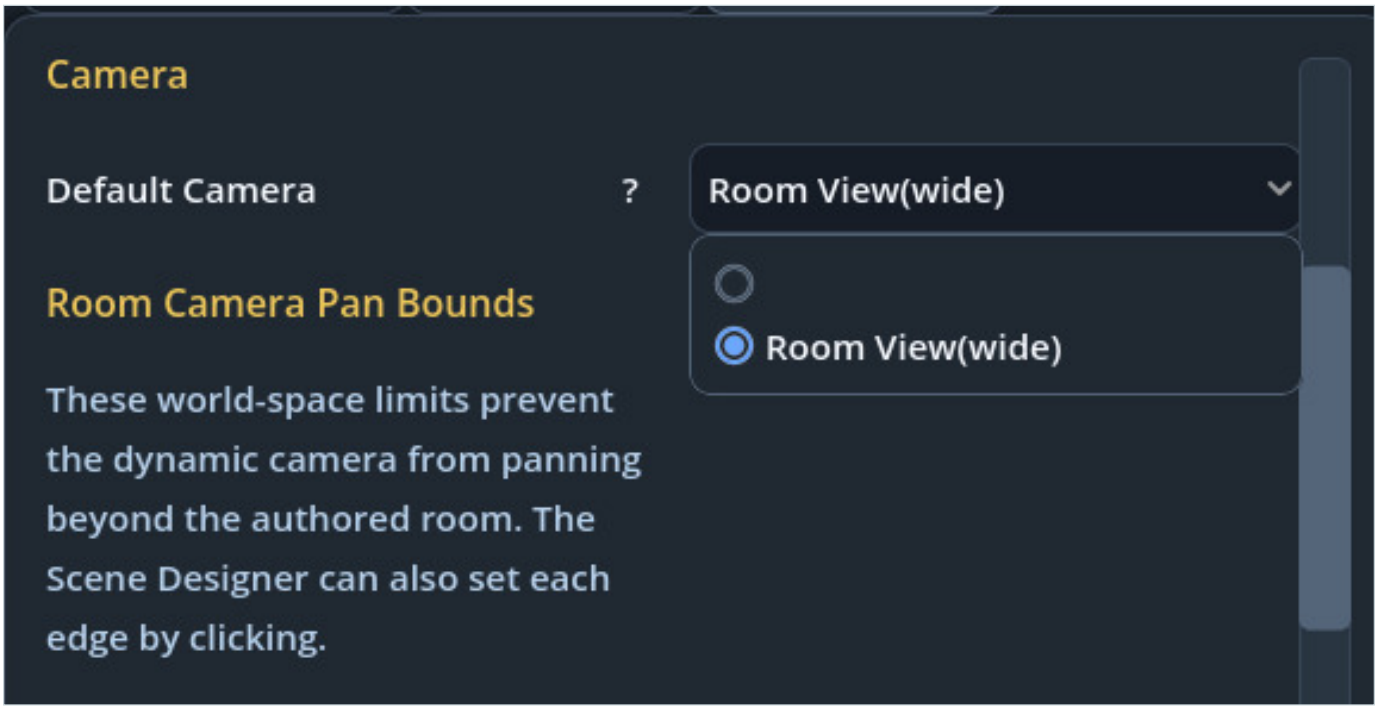


Figure 37. The room selects its Default Camera. The name and ID help distinguish it from other authored shots.

The project can start at the correct player and still use an unintended room camera. Check both settings: Project chooses how the game starts; the room chooses which camera opens that room.

For a fixed opening shot, leave special camera following and Advanced Camera behaviour at their ordinary inherited/disabled defaults unless you have deliberately configured them. A project or room follow rule can move a camera after it has been selected, so investigate those overrides if a saved fixed shot immediately drifts during play.

19. Copy an editor view into a camera

Typing camera coordinates is dependable. Sometimes it is easier to compose the shot by looking around first.

1. Move the editor view to an angle you like.
2. In the Scene Designer toolbar, click **Copy View to Camera**. Scroll the toolbar sideways if necessary.
3. Select the authored camera which should receive this shot.
4. Click **Apply View**.
5. Check its preview, confirm the room's Default Camera, and save.

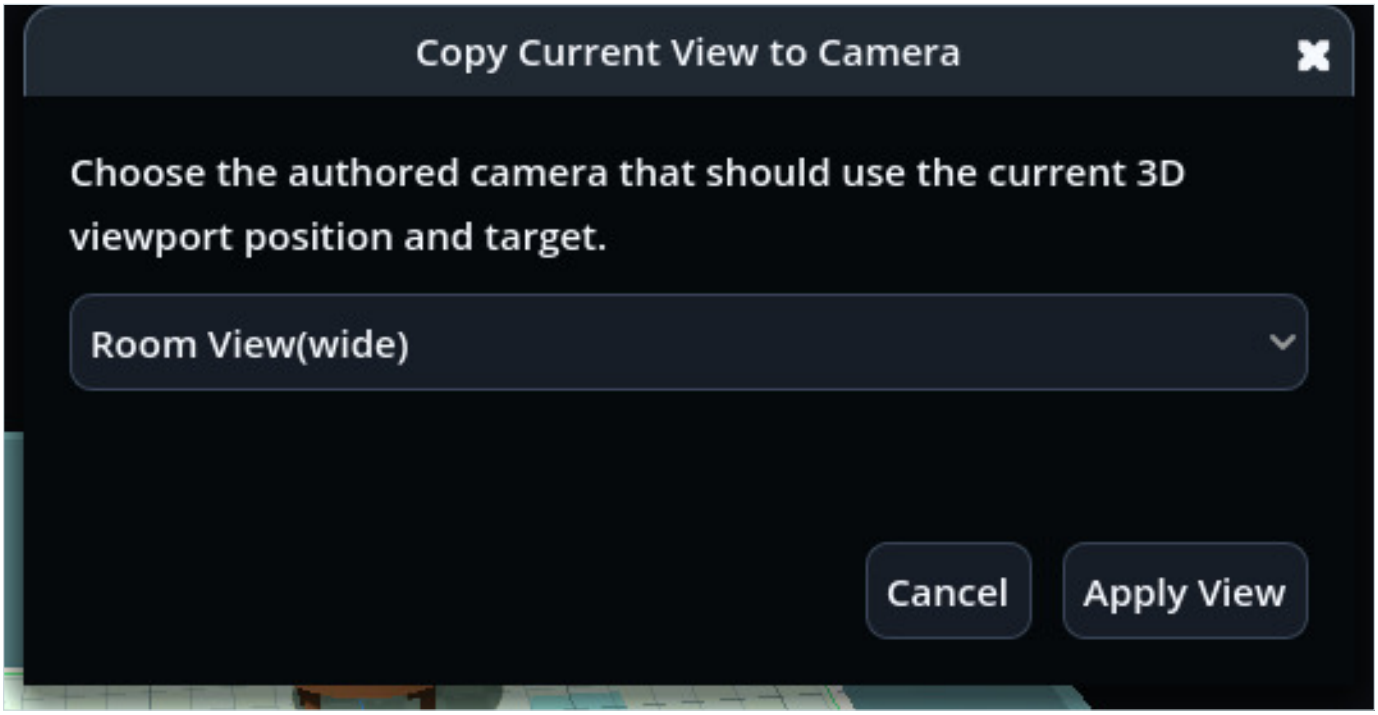


Figure 38. Copy View to Camera writes the editor's current position, aim and field of view to the chosen authored camera.

This changes the camera record. Merely orbiting the editor view does not. If you are following the numeric recipe exactly, keep the values in Chapter 17 instead of replacing them with an experimental view.

Check the authored camera preview

Select Room View in Scene Objects. The designer can show the selected camera's preview alongside its settings. This preview is useful for seeing an obstructing wall or an actor near the edge of the shot.

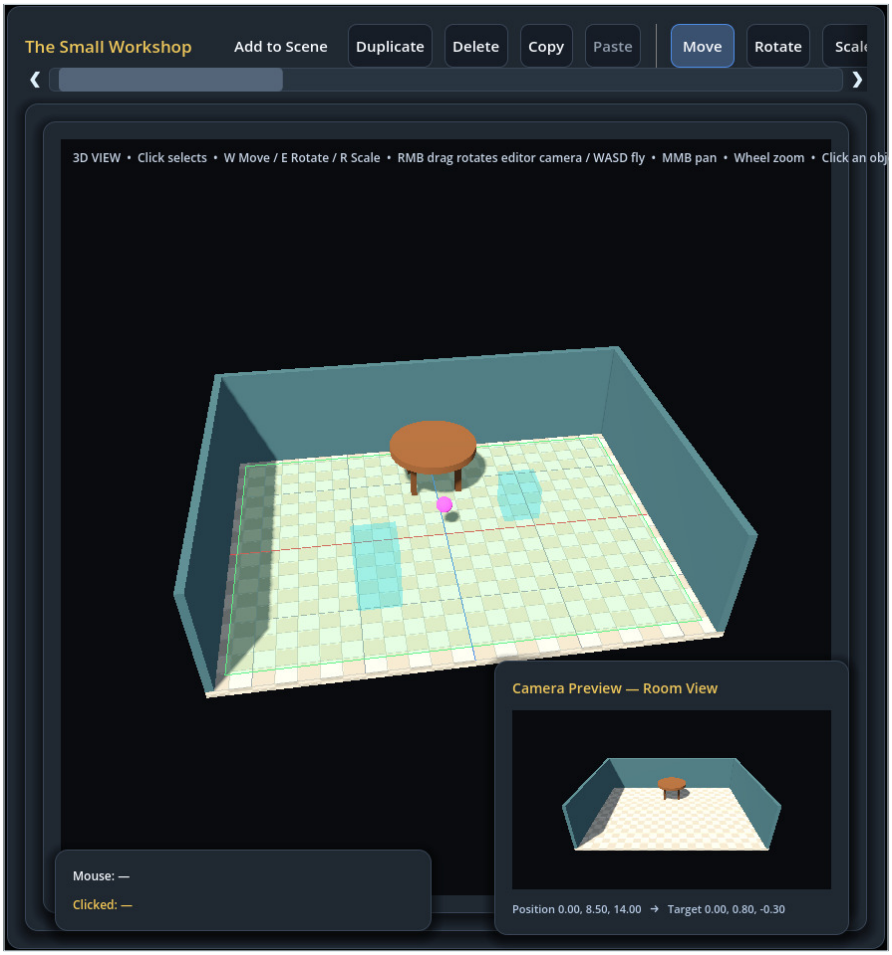


Figure 39. Selecting the authored camera exposes its settings and preview. Use a play test as the final framing check.

The camera's visible outline, target handle and preview are editing aids. They do not mean the camera body itself will appear as a prop in the game.

20. Give Mara one simple response

An NPC can stand in the room without having any interaction attached. Add one short response so that the play test checks more than appearance.

1. Select Mara in Scene Objects.
2. Open **Verb Actions**.
3. Set **Default Verb** to **Talk** for this context-sensitive example.
4. Find the Talk action group. Add or select a **Show Message** action in its action selector.
5. Enter a short line, such as **Mara: Welcome to the workshop. The floor and walls began as ordinary boxes.**
6. Set Duration to about **6 seconds** and leave Block Player Input off for this simple test.
7. Apply Changes and Save Project.

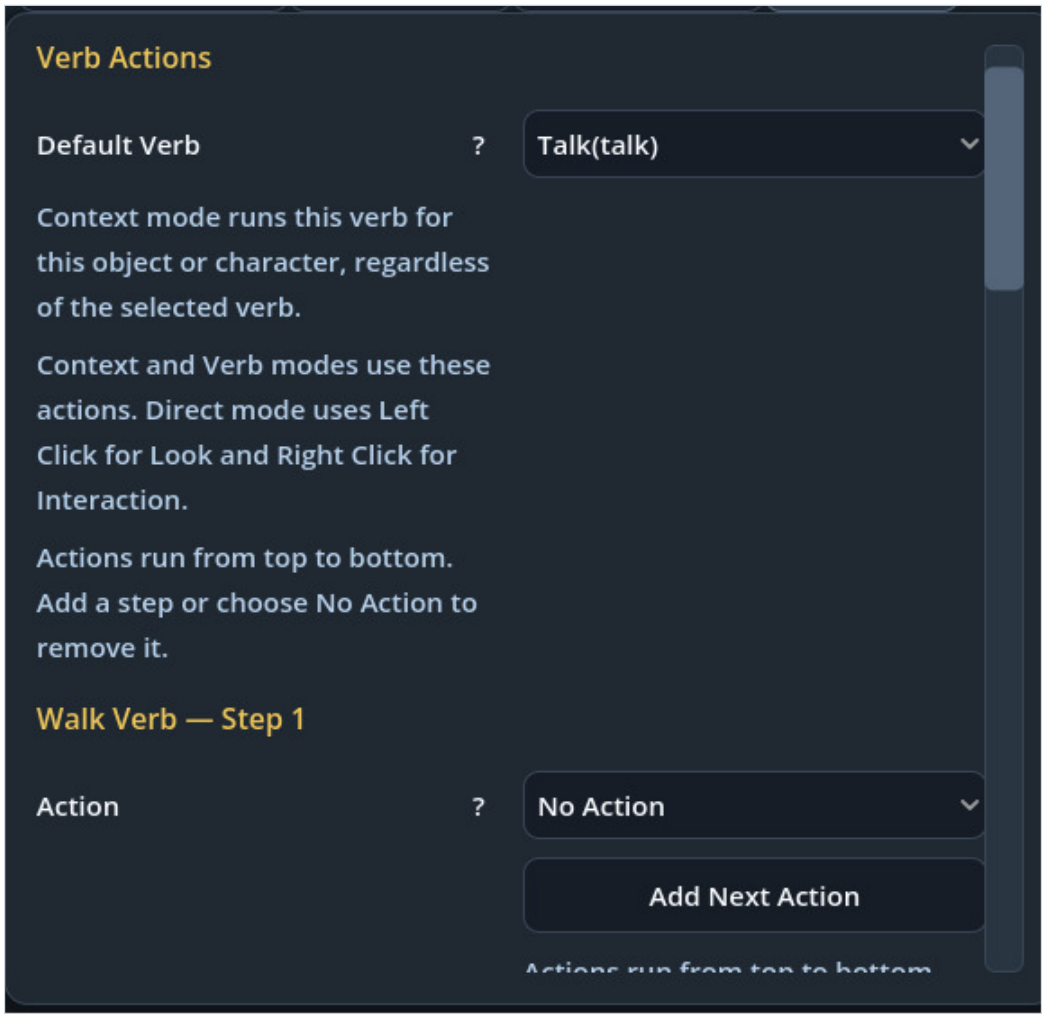


Figure 40. Verb Actions assigns behaviour to Mara. Default Verb lets a context-sensitive click use Talk.

Some action groups use **Add Next Action** to append another step. Actions run in order. Do not fill a Walk group when you intended to edit Talk; scroll to the correct labelled group.

Show Message is a straightforward text response. A full conversation can use Say, a speaker, recorded voice and a Dialogue record later. Those are separate story-authoring tasks. This small response is enough to check that the NPC can be selected and that its action runs.

In other interaction modes, choose Talk using that mode's normal verb controls before clicking Mara. The default-verb test here assumes the new project's context-sensitive setup.

21. Play-test the room

Save before testing. Then click **Test Scene** with The Small Workshop open. Use Play / Restart afterwards to check the complete startup route too.

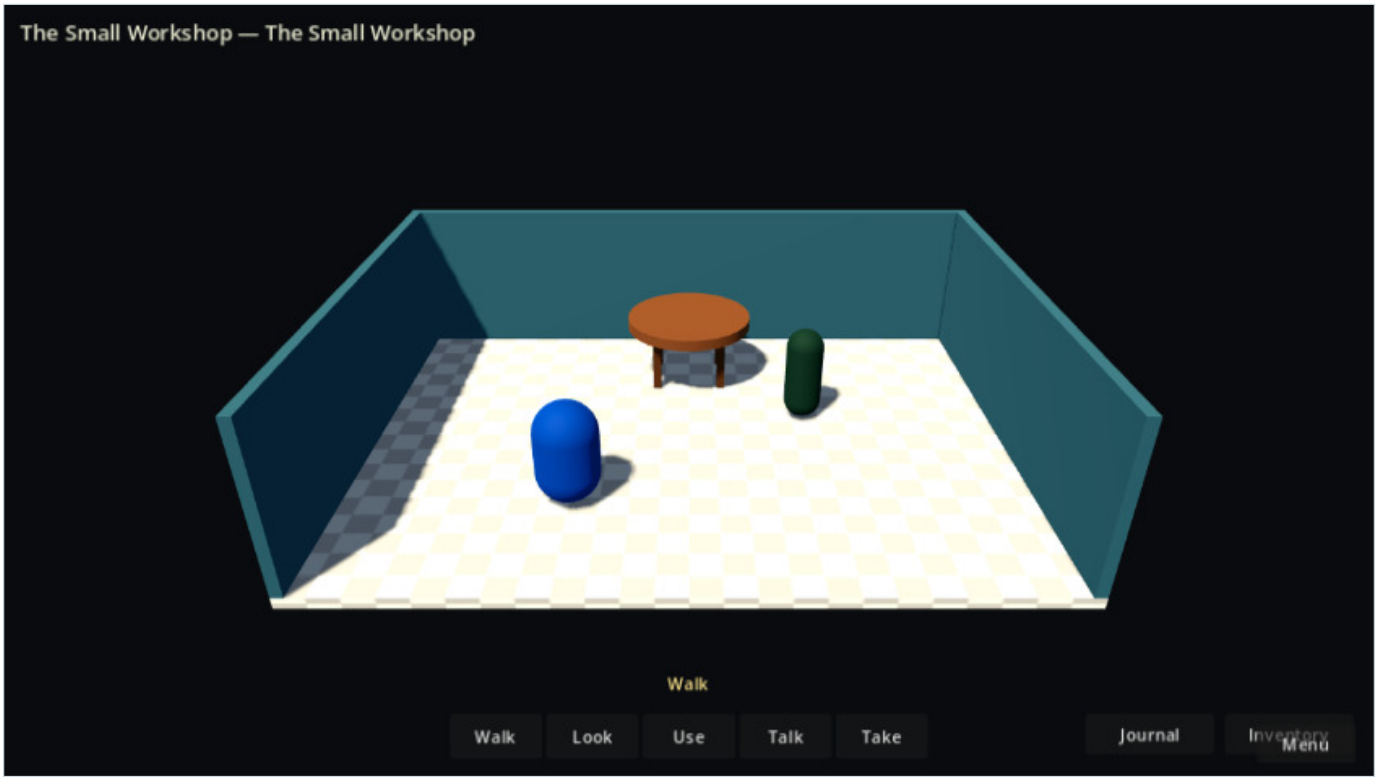


Figure 41. The room running with its authored camera, two character placeholders, floor, walls, workbench and light.

Walk the useful routes

Click an empty area of floor. Walk from the player's starting corner toward the centre, then toward Mara. Try another point near a side wall. Keep the destination on the floor rather than clicking the back wall and expecting it to become a walk target.

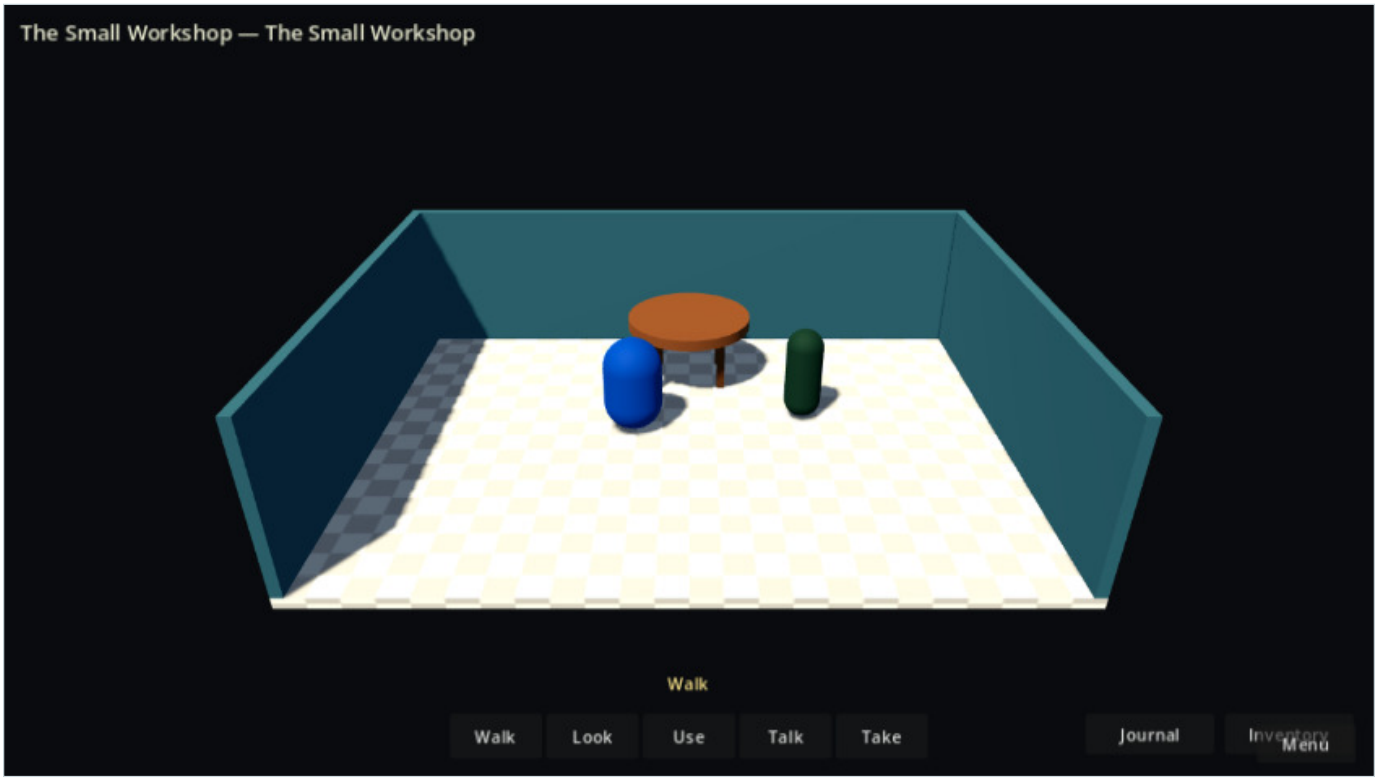


Figure 42. The player has moved from its starting position. Test actual movement rather than relying only on a green editor overlay.

Check that the actor stays on the floor, has a route past the workbench, and remains inside the intended walking area. A clear route should exist between the player and Mara. The room can look spacious while an oversized collider or blocker still obstructs that route.

Click Mara using the appropriate interaction mode. The message should appear and then clear after its duration.

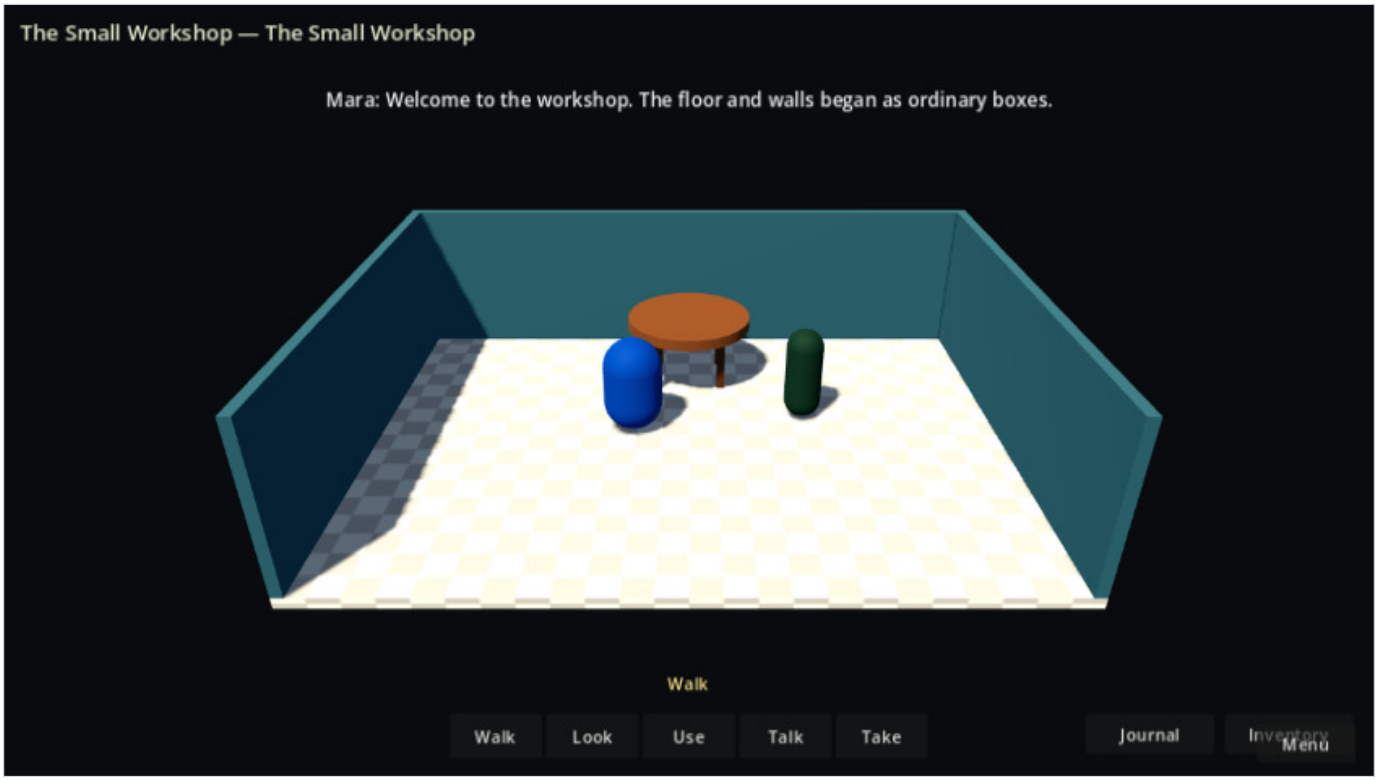


Figure 43. A simple NPC response confirms that the placed character's interaction is working.

Stop, change and restart

Use **Stop** to return to editing. Make the next change, Apply it, save, then restart the test. The running game is a runtime view of project data; restarting is a dependable way to check the latest complete state.

> **Before you continue**
> The player should begin in the workshop, the intended camera should open, the floor should be visible, the actor should walk, and Mara's response should run.

22. Keep the room manageable

Use meaningful IDs and names before the list becomes crowded. Floor, Back Wall and Room Key Light are much easier to find than several nearly identical New Object entries.

Lock, enable and show

Control	What it is for
Lock Selected or the lock column	Prevents accidental editing transforms. Unlock the item before trying to move it again.
Enabled at Runtime / enabled column	Controls whether the record participates in the game.
Visible / eye column	Controls the item's authored visibility where supported by that record.
Editor Visibility menu	Hides whole categories from the editing view only. It does not remove them from the game.

Locking the floor and finished walls can save a lot of accidental dragging. An item hidden by Editor Visibility can still exist in Scene Objects, so select it there or show the category again.

Use folders for architecture, props, characters and lights as the room grows. Database folders organise records; they are not a substitute for the object's actual 3D position.

Duplicate and copy carefully

Duplicate is useful for repeated primitive walls, posts and lights. A duplicate needs a clear name and a deliberate position. Copy/Paste is useful for reusing a compatible scene record, but its references and placement still need checking in the destination.

Characters have a special rule: create or clone the reusable Character definition for a distinct actor, then place it. Do not treat the scene-instance Duplicate restriction as a reason to turn an NPC into an ordinary Asset object.

Build in passes

First establish the room size and walking route. Then place the actors and camera. After that, improve materials, furniture and lighting. Keep the plain, functional version saved before adding complicated effects.

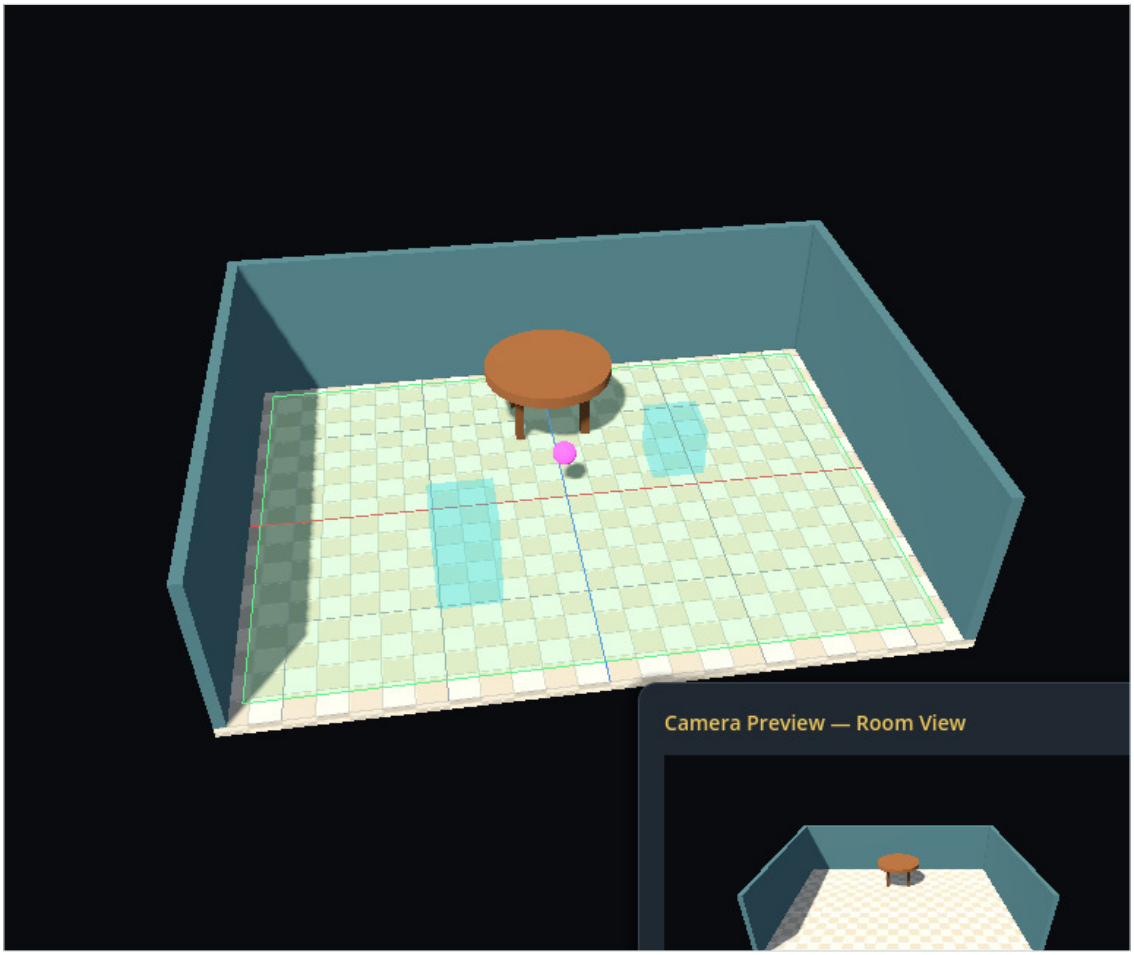


Figure 44. The finished editing scene. The solid-colour walls keep attention on the actors and the walking route.

23. The rest of the scene tools

You do not need every tool to build this workshop. This reference explains where the other controls fit when you are ready to extend it.

Add to Scene

Category or item	Typical job
Primitives	Simple Box, Sphere, Cylinder and Plane geometry.
Procedural	Generated furniture, architecture, vegetation and other supported props.
Character	Places an existing reusable Character definition.
Imported Asset / model actor	Places media from the asset database as an object.
Particle Emitter	Adds a configured visual effect to the world.
Point, Spot or Directional Light	Adds authored room lighting.
RGB Lighting	Creates or configures decorative light strings.
Trigger	Runs actions for supported entry, exit, click or other events.
Region	Defines an area for supported interactions or local effects.
Ambient Sound	Places a sound source with an audible area.
Spawn	Defines a named arrival position for scene changes or another spawn-aware action.
Navigation Blocker	Marks an area that should be excluded from walking.
Camera	Adds another authored shot. It still needs selection as default or activation by game behaviour.
2D / 2.5D	Adds the appropriate stage/background/object records for those room workflows.

Room Inspector tabs

Tab	Main purpose
General	Room identity, audio, background and broad visibility/view settings.
2D / 2.5D Room	The initial room-family choice and stage-specific setup.
Camera & View	Default Camera, room viewport choices and camera bounds/follow overrides.
Lighting & Fog	The room's environment and supported lighting/fog settings.
Post Process	Screen presentation and supported room effects.
Room Events	Actions at room entry, exit and other supported room events.
Transitions	How this room appears or disappears during scene changes.
Screen Overlay	The room's overlay presentation.
2D Camera / 2D Navigation / 2D Layers & Objects	Settings and guidance for the stage workflows. These do not replace a 3D navigation area's Shape controls.
Custom Properties / Raw JSON	Extra data or advanced inspection. They are not required for the room built here.

Object and camera tabs

A primitive's General tab contains identity, blocking, colliders, event-volume information and geometry. Visual Adjustments holds the world transform. Visual Effects holds textures, material response, colour and effects. Object Events, Verb Actions and Inventory Actions are different routes for behaviour; choosing a texture does not create those actions.

A camera's General tab identifies it. Position & View frames the shot. Dynamic Follow and Advanced Camera control more elaborate behaviour. Depth of Field is a rendering effect, with renderer-specific requirements; it does not focus or resize the walking area.

Camera bounds are not navigation bounds

Camera Bounds limits what a moving/panning camera can reveal. Navigation Areas limits the actor's walkable surface. Use Navigation Bounds in the camera-bounds tool can provide a convenient starting rectangle, but the two systems still have different jobs.

Similarly, changing the room's viewport resolution changes its logical image size. It does not make the floor wider or give the player a larger walkable map.

24. Fix the common problems

Read the symptom first, then change one relevant setting. Do not rebuild the whole room because one field was wrong.

Problem	Check first	Fix to try
Inspector changes the wrong thing	Its heading and selected object.	Select Floor, the wall, the room or the camera explicitly in Scene Objects.
New box is difficult to see	Its position, current view and selection.	Select it in Scene Objects and use Frame Selected.
Wall floats above the floor	Size Y and Position Y.	A 3-high centred box needs Position Y 1.5 to begin at Y 0.
Floor is above or below the actor	The floor's top surface and walking height.	For thickness 0.2, use centre Y -0.1 with walking at Y 0.
Object is much larger than its Size suggests	Transform Scale.	Restore Scale 1, 1, 1 while following this recipe.
Rotation looks wrong after entering 90	The field's angular unit.	These ordinary 3D Rotation fields use radians; use the gizmo or the correct radian value.
Texture is missing from the list	Image Kind and applied asset record.	Apply the Image asset in Asset Designer, then return to the primitive.
Texture is stretched or faces differ	Mapping and transform scale.	Compare Model UV with Triplanar and adjust Texture Scale.
Image colours look unexpectedly tinted	Tint Texture with Material Color.	Turn it off for the image's original colours, then Apply.
Scene looks lit in the editor but dark in play	Authored light and room environment.	Add/check the room light; editor inspection lighting is not the complete game setup.
Player cannot walk anywhere	Navigation enabled, floor blocking and starting position.	Keep the flat floor from acting as a room-wide obstacle and put the player inside a connected area.
Player cannot reach the enlarged part of a floor	Navigation Area bounds.	Resize the walkable area too; changing Grid spacing does not do it.
Player stops too far from a prop	Collider size/offset, body radius and interaction approach.	Check physical clearance before reducing Stop Distance.
Player walks through a wall	Blocks Other Characters and collider.	Enable blocking and check the physical box covers the visible wall.

Problem	Check first	Fix to try
NPC is absent from the placement chooser	Applied reusable definition or an existing room placement.	Save the definition; check Characters in Scene Objects for an existing instance.
NPC looks like a placeholder	Empty Model / Asset.	This is expected in the basic lesson. Assign a model and real animation states when ready.
Game controls the wrong character	Playable/default flags and project Player Character.	Leave Mara as an NPC and select Player(player) for startup.
Wrong room opens from Play	Startup Action and player placement.	Use Start in Player Character Location and verify the player is in this room.
Wrong camera opens	Room's Default Camera.	Select Room View(wide), Apply and Save.
Camera immediately moves away	Inherited room/project follow or advanced camera behaviour.	Check those overrides, rather than repeatedly changing the fixed position.
Wall blocks the entire view	Camera position and intervening geometry.	Use the open front or move/aim the camera deliberately.
NPC click does nothing	Correct verb group and interaction mode.	Put the response in Talk and use the intended Default Verb or verb UI.
Changes disappear after reopening	Apply and Save.	Apply the edited record, then Save Project.

R1. The workshop measurements

These values describe the room built in the lesson. They are reference tables inside the book, not files to import. All floor and wall Rotation values are 0, 0, 0 and Scale values are 1, 1, 1.

Object	ID	Size X, Y, Z	Position X, Y, Z
Floor	floor	12, 0.2, 9	0, -0.1, 0
Back Wall	wall_back	12, 3, 0.2	0, 1.5, -4.4
Left Wall	wall_left	0.2, 3, 9	-5.9, 1.5, 0
Right Wall	wall_right	0.2, 3, 9	5.9, 1.5, 0
Workbench	workbench	Table generator's normal geometry	0, 0, -2.5
Player	player	Character height about 1.8	-2, 0, 2
Mara	mara	Character height about 1.8	2.2, 0, -0.8

Setting	Value
Floor blocking	Off for this flat navigation setup.
Walls and workbench blocking	On.
Floor solid colour	#918b7c
Wall solid colour	#486d71
Workbench colour	#8b5832
Primitive material Metallic / Roughness	0 / about 0.85
Optional floor Image asset	Floor Tiles(floor_tiles), or your own applied Image asset.
Optional floor texture settings	Triplanar; start at Scale 1, compare 0.5 and 2; tint off.
Walkable rectangle	Min X -5.6, Max X 5.6, Min Z -4.1, Max Z 4.1.
Walking height	Y 0 in the starter's flat navigation.
Room camera	Room View(wide).
Camera position / target	0, 8.5, 14 / 0, 0.8, -0.3.

Setting	Value
Camera FOV / Near / Far	48 / 0.05 / 100.
Room Key Light	Directional; #fff1d8; intensity 0.7; Cast Shadows on.
Light rotation, radians	-0.96, -0.59, 0.
Project startup	Start in Player Character Location; Player(player).

R2. A final room check

Work through this once before moving on to a more elaborate room.

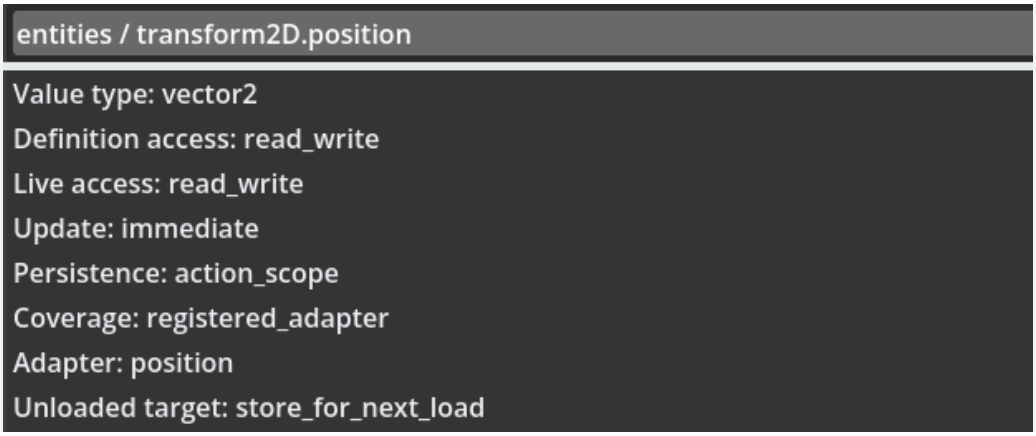
1. The room family is 3D and the working project has been saved.
2. Floor is a thin box whose top meets the walking height.
3. All three walls meet the floor and leave the camera's front view open.
4. Each surface has the intended colour or applied Image texture.
5. The room has an authored light.
6. A connected navigation area covers the usable interior.
7. Floor does not obstruct the whole walking area; solid walls and furniture block appropriately.
8. Mara is a reusable NPC and has one placement in this room.
9. Player is playable, is the intended starting character, and has a placement inside the walking area.
10. Room View has the intended aim and is selected as Default Camera.
11. Test Scene allows walking and runs Mara's simple response.
12. Play / Restart begins the complete game in the expected room and shot.

The screenshots show the editor and a small working scene made for this guide. The camera, geometry and interaction examples were checked in the running engine. They illustrate the version named on the cover; use later release notes if a field moves or changes. No separate practice project, ZIP or PDF attachment is included.

RUNTIME WALKTHROUGH

Move a native 2D object

A 2D entity has a position on the room canvas measured in pixels. A painted background layer has its own scenery settings. Choose the entity when you want to move one sign, light or prop.



0.86.4.0 reference • entities / transform2D.position expects two numbers. The capital D is part of the path.

1 Find the placed object

In Scene Designer, open a 2D room and select a native 2D entity on its canvas. Note the room ID, entity ID and starting X and Y. Use an object whose initial position is easy to see.

2 Give it a destination

On a switch or interaction add Change Object Property. Choose the room and that entity. Set Property to transform2D.position. For a sign starting at [100, 80], enter [124, 80] to move it 24 pixels right. Use a destination relative to your own starting coordinates.

3 Check rotation or scale separately

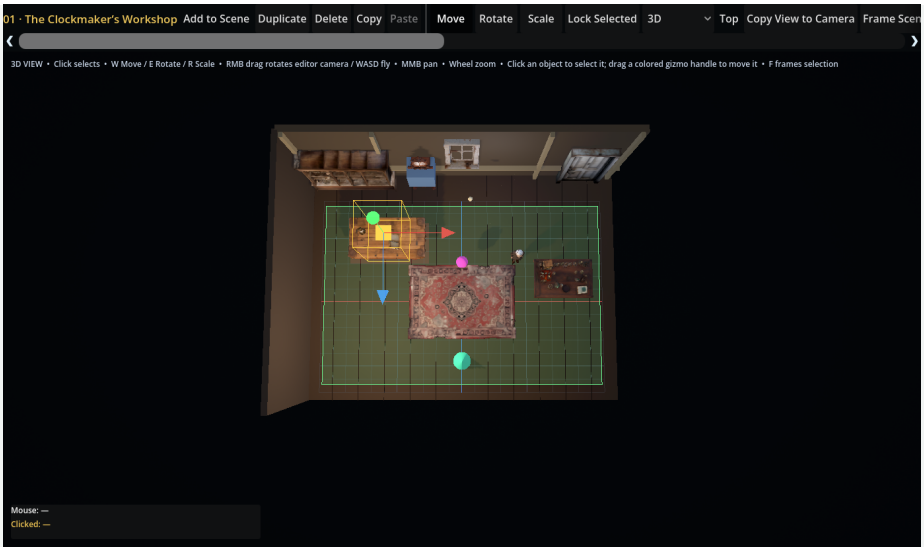
Run the event and look for the rightward step. In a second action try transform2D.scale with [1.25, 1.25]. To turn the object, use transform2D.rotation with one angle. The default unit is degrees; a record explicitly configured for radians uses radians. Change one field at a time so you can see its effect.

> **When the target is a character**
>
> character.walkSpeed, character.turnSpeed, character.animationState and character.animationSpeed affect supported actor controllers. A movement rate changes how quickly a character travels. An animation rate changes how quickly its clip plays.

RUNTIME WALKTHROUGH

Give a placed prop a new model

Keep the workbench as the target of your interaction. Change only the visual asset when the player repairs it. Existing events can still point to the same workbench.



0.86.4.0 Scene Designer • The highlighted workbench sits left of the rug in the Clockmaker workshop. The move handles belong to the placed object. Use that object as the repair target.

1 Identify the two records

Select the workbench in the room. Write down its object ID. In the asset list, find a second model already imported into the project. Open Asset Preview and check that the replacement faces and sits the way you expect. Note its asset ID.

2 Wire the repair

Open the repair interaction and add Change Object Property. In Object choose the original workbench. Set Property to assetId and To to the replacement asset ID. Choose room to see it for this visit, or persistent if the repaired workbench must appear in a later checkpoint.

3 Run it twice

Trigger the interaction. A model-based asset object may report component_rebuilt because its visual was exchanged. Use the workbench again and check its interaction still runs. An unavailable or incompatible resource should leave the previous visual installed.

> **Check the collision separately**
> Stand on each side of the new model. A wider or taller visual can extend beyond the object's authored collision shape. Compare the physical shape and any attach points with the replacement art.