Test and Experiment

Testing Your AR Experience in the Preview Panel

Learning Objectives

1. Understand what to look for as you are testing your experience
2. Describe the Parent-Child Entity relationship
3. Explain how to hide objects without deleting them

Test Your Lens

Testing your tap action in the Preview Panel is an important step as we progress. It transforms your project from a static design into a functional, interactive AR experience. However, a professional developer doesn't just watch the screen—they also "listen" to the software.

How to Run Your Test

Once you have attached your Behavior Script to your image, you need to verify that the math and logic are working correctly.

1. *Locate the Preview Panel:* This is the window on the far right of your screen.
2. *The Interaction:* Move your mouse over the phone screen in the Preview.
3. *The Trigger:* Click directly on the image you have set up.
4. *The Observation:* Watch the image to see if it grows or shrinks based on your X, Y, Z settings.
5. *The Data Check (The Logger)*: Look at the Logger at the bottom of your screen.
    - If you see Red Text, your “rubber duck” partner needs to help you investigate a logic error!

What to Look For During the Test

When testing a scale action, pay attention to the "feel" of the movement and the feedback from the software:

- *Response Time*: Does the image scale the instant you tap? If there is a delay, check your Trigger settings in the Inspector.
- *The Logger’s Story*: Is the Logger silent, or is it barking errors? A successful test usually means a clean, quiet Logger
- *Anchor Points*: Does the image grow from the center, or does it "slide" to one side? This relates back to the Pivot Position. If your pivot is in the center, it will scale evenly in all directions.
- *Resetting*: Click the Reset icon (the circular arrow) in the Preview Panel to return the image to its original size and start the test fresh.

Experimenting!

Now that you've set up your first interaction, it's time to move from "following instructions" to Experimenting.

This is where you truly learn how the "logic" of your Lens works by seeing what happens when you push the buttons!

Laboratory: The Hierarchy Experiment

Take a look at your Objects Panel on the left. You'll notice that your Behavior Script isn't just floating around—it is nested directly underneath your Image Object. This is called a Parent-Child Relationship.

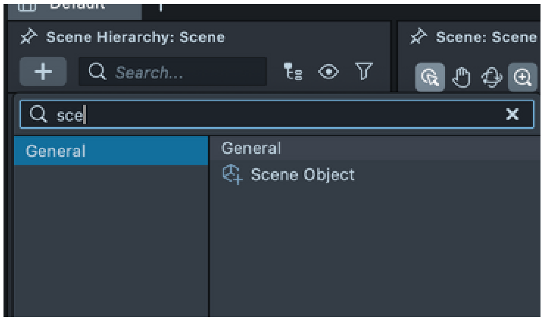
Who is the Boss?

In computer science and 3D design, the object on top is the Parent, and everything tucked underneath it is the Child.

- *The Rule*: Whatever happens to the Parent must happen to the Child.
- *Scaling Results*: If you use the Scale Tool (R) to make the Parent (the Image) twice as big, the Child (the Behavior Script) stays attached and grows with it

Experiment: The "Parenting" Test

To truly see the difference between Local and World scale, we need to create a "Family Tree" in your Objects Panel. This will show you how a "Child" object reacts when its "Parent" changes.

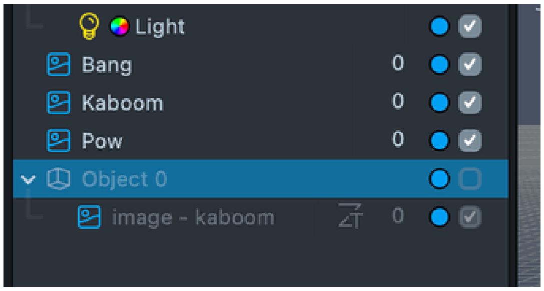


The Set-Up

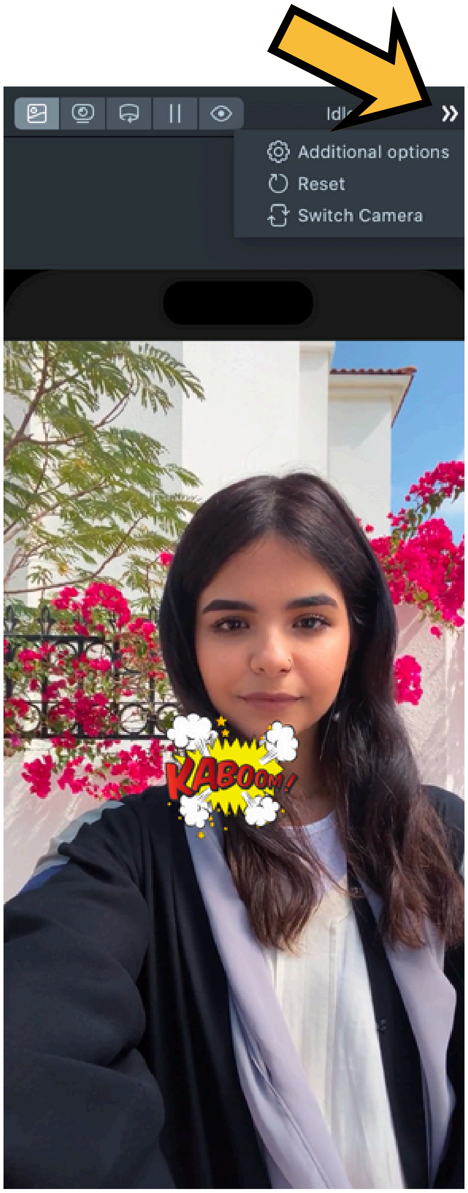
1. *Create an Object:* Click + Add New in the Objects Panel and select Scene Object. (Think of this as an empty, invisible box).
2. *Make a Child:* In the Objects Panel, click and drag your Image Object directly on top of the new Scene Object.
    - You'll know it worked if the image is now indented underneath the Scene Object.

Tip:

If you created something cool and don't want to delete it but it's in the way right now, uncheck the box to make it invisible and inactive until you're ready to use it.



When experimenting with triggers and scaling, the Preview Panel can quickly become a mess of giant or tiny images. Fortunately, Lens Studio allows developers to quickly reset the preview.



Don’t Fret - Reset!

As you experiment with Behavior Scripts, you might accidentally make your image so big it covers the whole screen, or so small it disappears! Instead of trying to "undo" every tap, you can simply refresh the simulation.

How to Reset the Preview

Look at the Preview Panel (the window that looks like a phone screen). In the top corner, you will see a >> icon. Click on it and select Reset. Every object will snap back to its starting position and size, and every Behavior Script "forgets" the taps you just made.

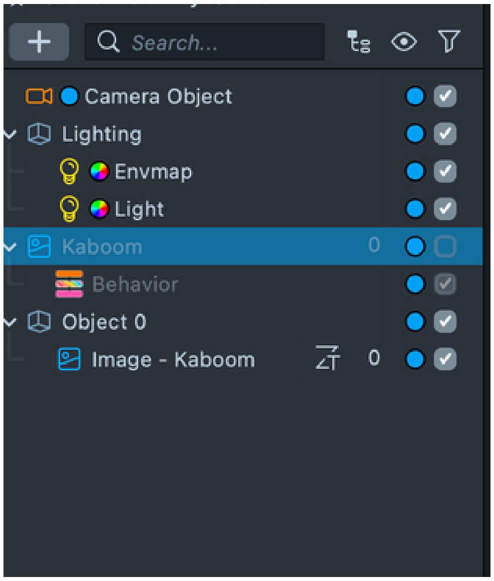
Hiding Objects

Sometimes, you need to get an object out of the way so you can focus on something else. You don't have to delete it! You can simply "turn it off" temporarily.

How to Toggle Visibility

In your Objects Panel (on the far left), look at the list of your images. Next to each name, there is a small checkbox.

- Uncheck the box: The object becomes invisible in both the Scene Grid and the Preview Panel. It's still there, but it's "sleeping."
- Check the box: The object "wakes up" and reappears exactly where you left it.



Logic Alert: The "Parent" Rule (Again!)

This is where the Parent-Child Relationship gets really interesting.

- If you uncheck the Parent (the Scene Object), every Child inside it disappears too!
- It’s like turning off the lights in a room; even if the furniture is still there, you can’t see any of it.

Knowledge Check

Relationships and Refinement

Instructions: Answer the following questions to demonstrate your understanding of object organization and the testing process.

1. The Observation Deck

When testing your "Tap to Scale" interaction in the Preview Panel, which of the following is a sign that your Pivot Point might be off-center?

A. The image changes color when you tap it.
B. The image grows perfectly in all directions from the middle.
C. The image "slides" or shifts to one side as it gets larger.
D. The Logger shows a red "Reference Error."

2. The "Sticker Sandwich": Parent-Child Logic

In Lens Studio, you can drag one object on top of another in the Objects Panel to create a Parent-Child relationship. If the "POW" sticker is the Parent and the "Bang" sticker is the Child, what happens to the "Bang" sticker if you move the "POW" sticker across the screen?

A. It stays exactly where it was, floating in space.
B. It follows the "POW" sticker and moves with it automatically.
C. It is deleted from the project.
D. It turns invisible.

3. Managing Visibility

You have an object in your scene that you want to hide for a moment, but you don't want to delete it because you might need it later. What is the best way to "hide" it?

A. Drag the object into the Trash in the Asset Browser.
B. Right-click the object and select "Delete."
C. Uncheck the Enabled box at the very top of the Inspector Panel for that object.
D. Move the object behind your computer monitor so you can't see it.

4. Short Answer: The Logger's Evidence

You tap your object in the Preview Panel and nothing happens. You look at the Logger and see a red error message. Why is this message more valuable than just seeing that the tap didn't work?

Answer: ____

(Hint: Think about your "Darth Vader" debugging partner and how they use data!)

Going Global

Using Behavior Scripts Globally

Learning Objectives

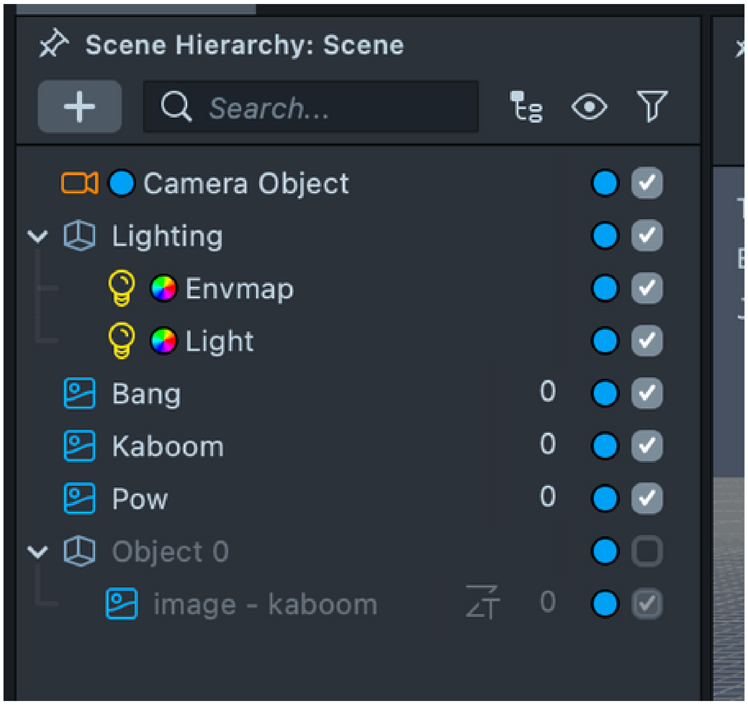
1. Understand what is meant by “global behavior” in AR experience
2. Describe how to create a global script
3. Explain the back camera logic
4. Describe “state management”

Behavior Scripts - Globally

Up until now, you've been giving instructions to specific objects (Local). But what if you want a behavior that affects the entire experience of the Lens—like background music that starts as soon as the Lens opens, or a countdown timer that affects every object at once? For this, we use a Global Behavior Script.

What is a Global Behavior?
In computer science, "Global" means it is not tied to just one small part of the system; it's available to the whole project.
Instead of hiding the "brain" inside a specific image folder, we create a dedicated Controller object. This is like the "Command Center" of your Lens. It doesn't have a physical body or an image; it only exists to hold instructions that rule over the entire experience.

Pro-Tip: Clean Hierarchy
By keeping your Global scripts in their own object and not nested in another object you won't have to hunt through dozens of images to find the script that controls your background music. It's all about staying organized as a developer!



Scene Hierarchy: Scene

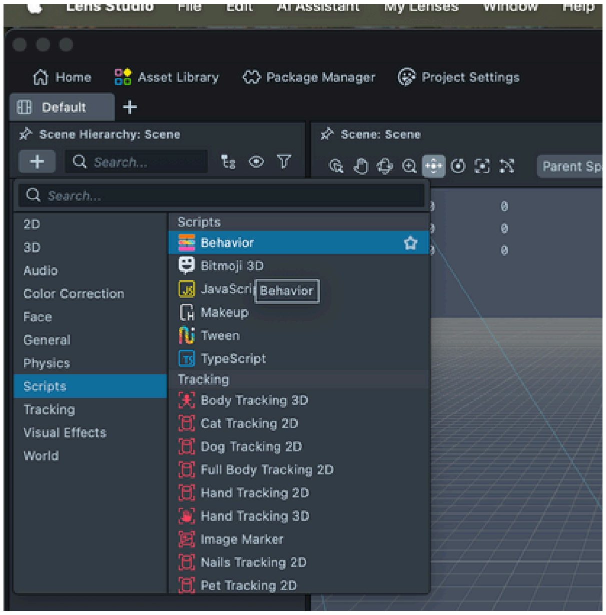
+ Search...

- Camera Object
- Lighting
  - Envmap
  - Light
- Bang 0
- Kaboom 0
- Pow 0
- Object 0
  - image - kaboom 0

Global Behavior is like the School Bell.

- A Local behavior is a teacher talking to one student.
- The Global behavior is the bell that rings for the entire building. It doesn't belong to any specific classroom; it sits in its own "Control Box" and tells everyone in the school that it's time to change classes!*

Adding a Global Script



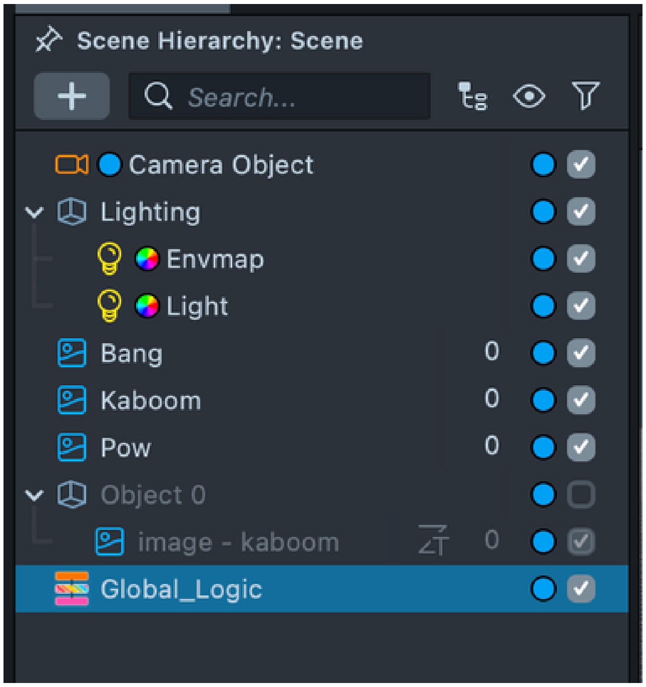
The Path to Global Power:

1. *Top Left Corner:* Look at the top of your Objects Panel and click the + Add New button.
2. *Navigate the Menu*: Click on Scripts to open that category.
3. *Select Behavior:* From the pop-out menu, click Behavior.

What Just Happened?

Look at your list of objects. You'll see a new item named Behavior.

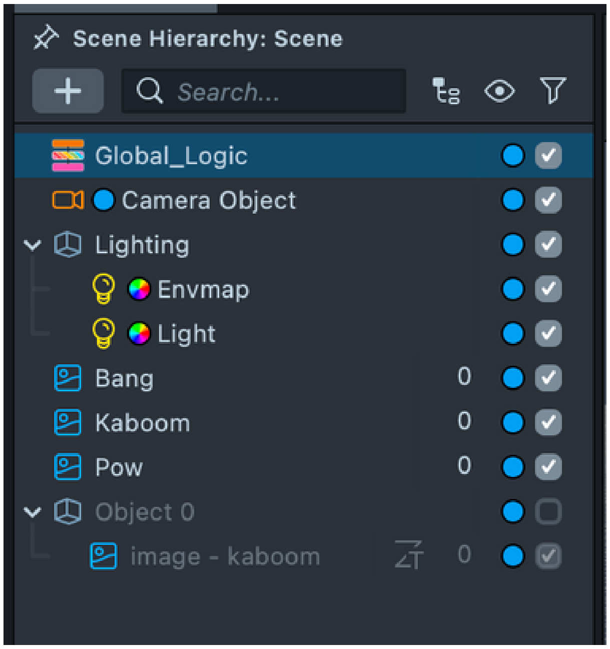
- The Invisible Box: Lens Studio automatically created a Scene Object for you and put the Behavior Script inside it.
- The Auto-Naming: Notice that it named the object "Behavior" for you. While this is helpful, it's still a good idea to rename it to something like Global_Logic so you don't get it confused with other scripts later!



Why this is great for "Global" logic:

When you use the Add Component method, you have to have an object selected first. When you use the + Add New method, you are creating something brand new and independent. This is perfect for Global scripts because they aren't "tied down" to your images from the start.

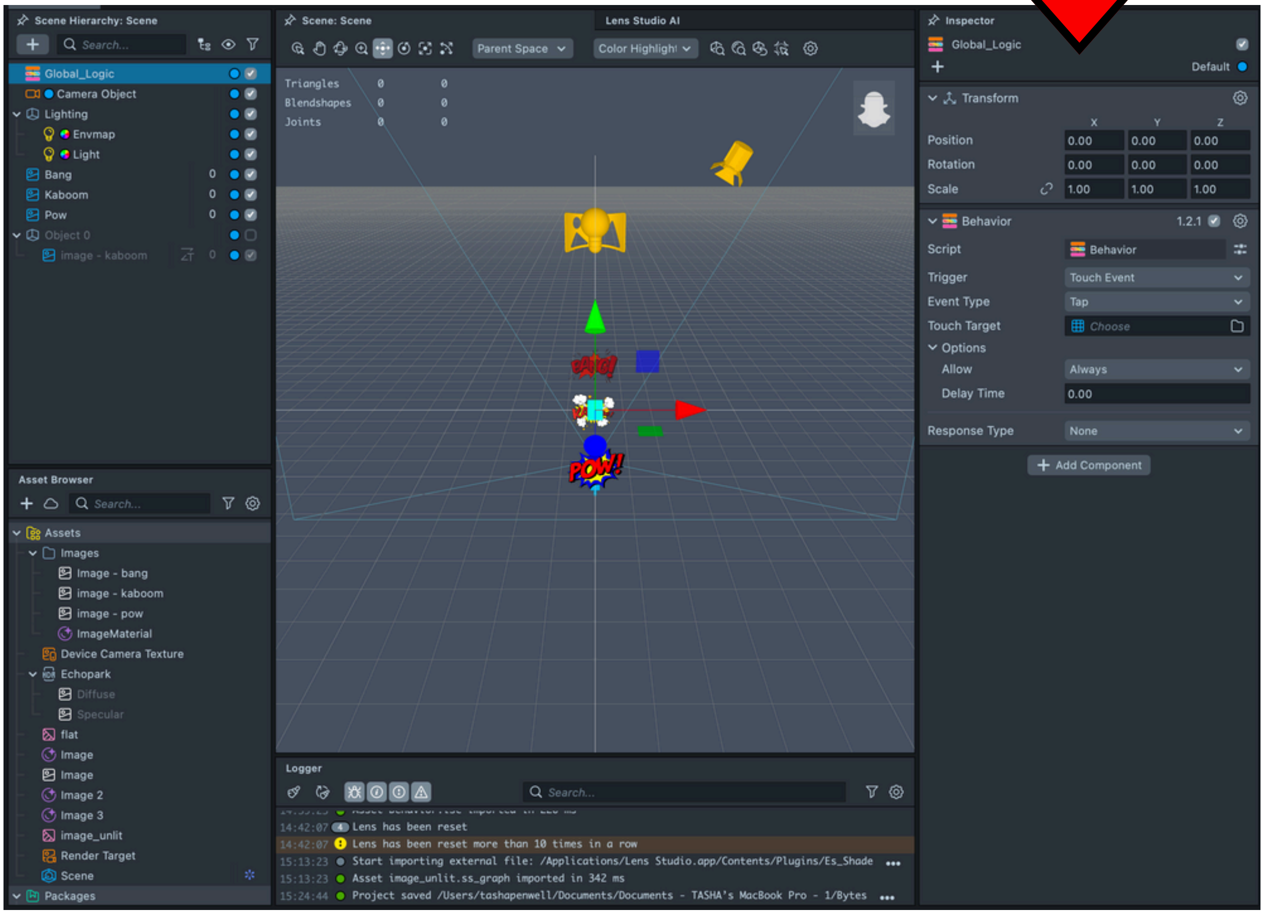
Pro-Tip: Keeping it Clean
Drag this new object to the very top of your list. It’s a professional habit that makes troubleshooting your "Global" logic much easier as your project grows.



Scene Hierarchy: Scene

- Global_Logic
- Camera Object
- Lighting
  - Envmap
  - Light
- Bang 0
- Kaboom 0
- Pow 0
- Object 0
  - image - kaboom 0

With the Behavior script selected, you can see the Inspector Panel is ready for any behaviors we want to add to the lens experience.



Scene Hierarchy: Scene

- Global_Logic
- Camera Object
- Lighting
  - Envmap
  - Light
- Bang 0
- Kaboom 0
- Pow 0
- Object 0
  - image - kaboom 0

Asset Browser

- Assets
  - Images
    - Image - bang
    - image - kaboom
    - image - pow
    - ImageMaterial
  - Device Camera Texture
  - Echopark
    - Diffuse
    - Specular
  - flat
  - Image
  - Image
  - Image 2
  - Image 3
  - image_unlit
  - Render Target
  - Scene
- Packages

Scene: Scene | Lens Studio AI

Parent Space | Color Highlight

```
Triangles     0     0
Blendshapes   0     0
Joints        0     0
```

Inspector

Global_Logic ☑

Default ●

Transform

	X	Y	Z
Position	0.00	0.00	0.00
Rotation	0.00	0.00	0.00
Scale	1.00	1.00	1.00

Behavior 1.2.1 ☑

Script	Behavior
Trigger	Touch Event
Event Type	Tap
Touch Target	Choose
Options	
Allow	Always
Delay Time	0.00
Response Type	None

+ Add Component

Logger

```
14:42:07  4  Lens has been reset
14:42:07  !  Lens has been reset more than 10 times in a row
15:13:23  ●  Start importing external file: /Applications/Lens Studio.app/Contents/Plugins/Es_Shade ...
15:13:23  ●  Asset image_unlit.ss_graph imported in 342 ms
15:24:44  ●  Project saved /Users/tashapenwell/Documents/Documents - TASHA's MacBook Pro - 1/Bytes ...
```

In AR development, we often want certain objects to appear only when the user is looking at the world (the Back Camera) and others to appear only when they are taking a selfie (the Front Camera). We can use a Global Behavior script to "enable" or "disable" these objects automatically.

The "Back Camera" Logic

We are going to tell our Global Controller to listen for a Camera Event. This ensures that your graphics only pop up when the user is pointed at the right thing.

Step 1: Setting the Global Trigger

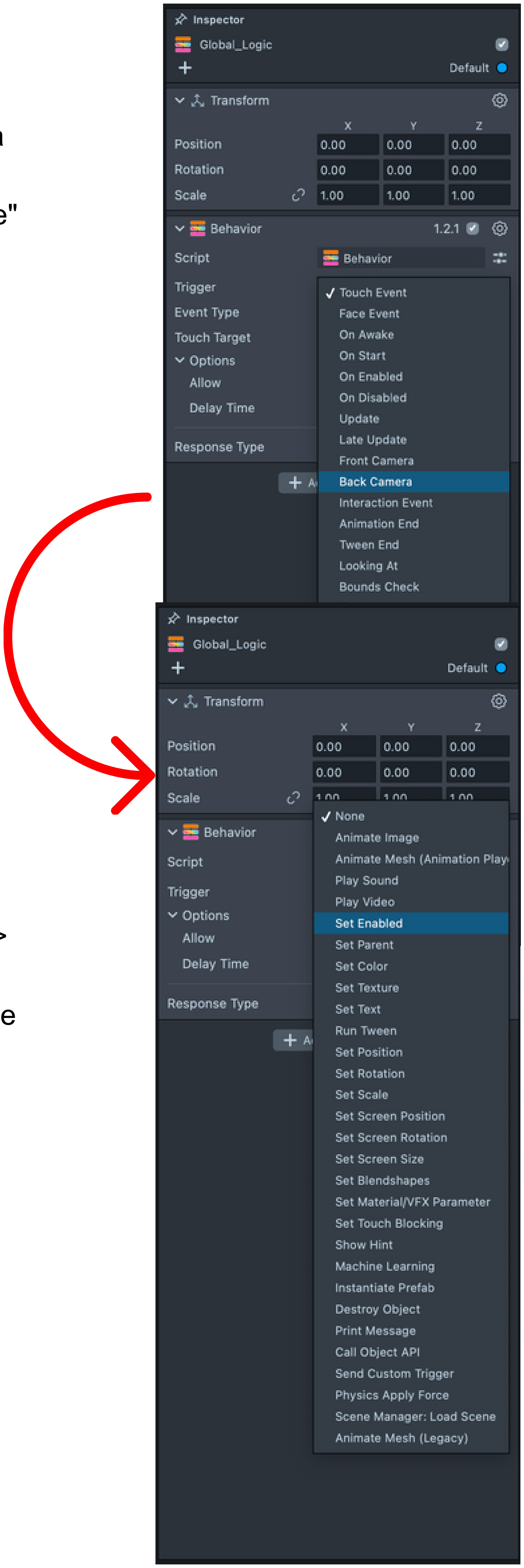
In the Inspector for your Global Behavior script:

1. *Trigger*: Click the dropdown and select “Back Camera”

Step 2: Setting the Response (Set Enabled)

Now we tell the Lens what to do when that back camera is active:

1. Response Type: Choose Scene Object > Set Enabled.
2. Target Entity: Click the box and select the Image Object (or the Parent folder) you want to show.



Inspector

Global_Logic

Default

Transform

	X	Y	Z
Position	0.00	0.00	0.00
Rotation	0.00	0.00	0.00
Scale	1.00	1.00	1.00

Behavior 1.2.1

Script: Behavior

In this demo, we want to have the “Pow” sticker disappear when the back camera is enabled.

To accomplish this, click on “Choose Target Object” and click on the appropriate sticker and then Ok

The Power of "Set Enabled" & Toggling

Think of Set Enabled like a master power switch for your AR objects. When you combine it with a Toggle command, you create a smart Lens that knows exactly when to show up.

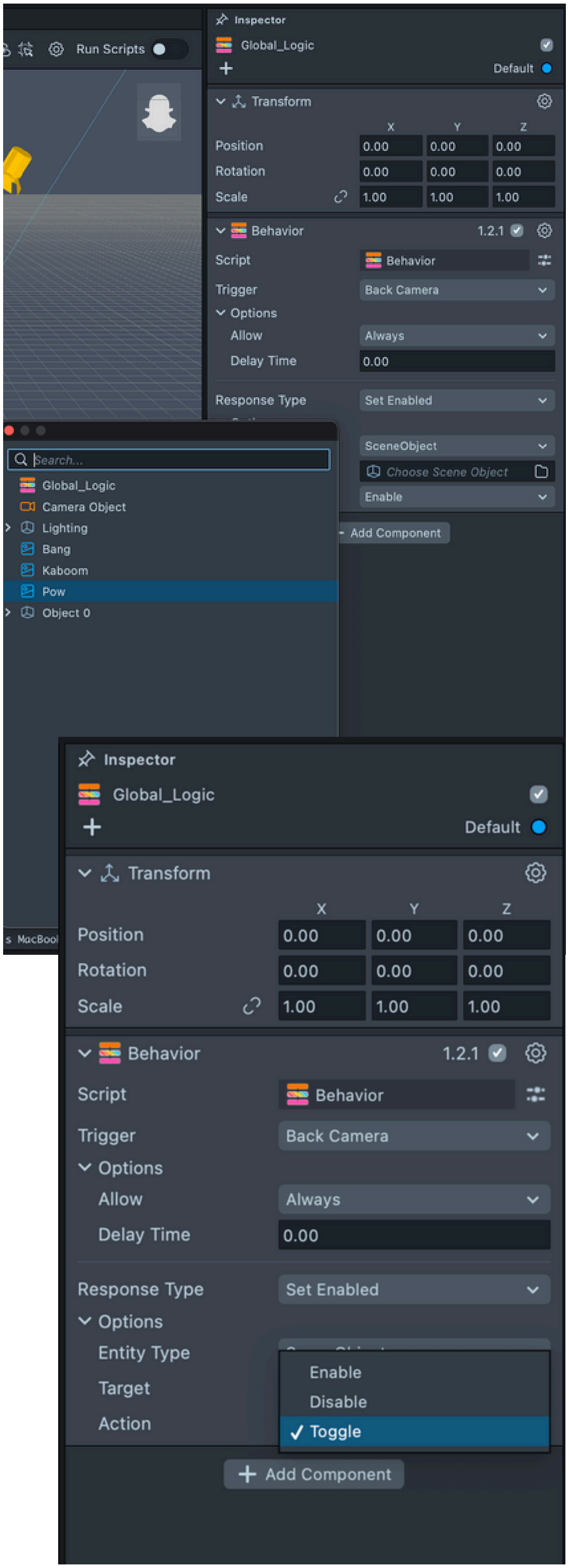
- Enabled (Checked): The object is "alive." It is visible in the scene, and all its scripts and animations are running.
- Disabled (Unchecked): The object is “dead”. It is completely invisible and takes up zero processing power—it’s like it doesn't even exist until you flip the switch.

The "Toggle" Logic

Instead of just turning something "On" or "Off," a Toggle tells Lens Studio: "Check the current state. If it’s on, turn it off. If it’s off, turn it on."

By using this globally, you can create a Lens with two distinct personalities:

- *Selfie Mode*: Your face filters are Enabled, and your world graphics are Disabled.
- *World Mode*: Your face filters shut off, and your 3D world graphics instantly Toggle on!



Inspector
Global_Logic
Default

Transform

	X	Y	Z
Position	0.00	0.00	0.00
Rotation	0.00	0.00	0.00
Scale	1.00	1.00	1.00

Behavior 1.2.1
Script: Behavior
Trigger: Back Camera
Options
Allow: Always
Delay Time: 0.00
Response Type: Set Enabled
Options
Entity Type: SceneObject
Target: Choose Scene Object
Action: Enable

Search...
Global_Logic
Camera Object
Lighting
Bang
Kaboom
Pow
Object 0

Global Variables vs. Local States

When you use the Global Behavior Script to toggle an object, you are practicing State Management.

- *State*: The current "truth" of your Lens (e.g., "The Back Camera is active").
- *The Global Listener:* Your script acts as a "Global Listener" that waits for the state to change.
- *The Execution*: Once the state changes (the flip), the script executes the toggle command on the Target Entity.

Knowledge Check

Global Logic and State Management

Instructions: Answer the following questions to demonstrate your understanding of how to control the "Big Picture" of your AR experience.

1. Defining "Global Behavior"

In an AR experience, what does it mean if a script or setting is considered Global?

A. It only works when the user is traveling to a different country.

B. It affects the entire Lens experience regardless of which specific object is selected.

C. It only controls the color of the ground.

D. It means the script was written by someone in a different part of the world.

2. Creating the "Brain": Global Scripts

To create a script that manages the whole scene, where is the best place to attach that script in your Objects Panel?

A. To a tiny sticker hidden in the corner.

B. To a "Script Holder" or "Controller" object that sits at the top level of your hierarchy.

C. Directly to the user's face.

D. You don't attach it; you just leave it in the Asset Browser.

3. The Back Camera Logic

When you switch your Lens logic to the Back Camera (World Tracking), how does the computer's "thinking" change?

A. It stops looking for a human face and starts looking for flat surfaces (like the floor or a table).

B. It turns the screen black to save battery.

C. It reverses all your stickers so they appear backward.

D. It only allows you to use the color green.

4. Understanding "State Management"

In programming, what is the goal of State Management?

A. To keep track of which U.S. state the user is currently in.

B. To remember "what is happening right now" (e.g., Is the game started? Is the secret sticker visible? Has the user clicked the button yet?).

C. To delete all assets after the Lens is closed.

D. To make sure the camera stays perfectly still.