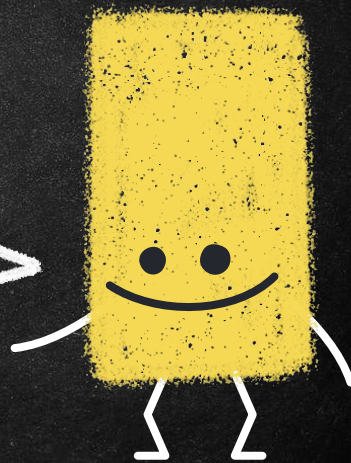
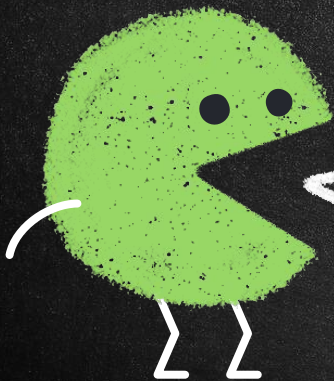




3D GRAPHICS

Lesson

by Aasine Cassara, Alana Robinson, Kate Maschmeyer,
Nicole Cojuangco, Patti Elfers, and Yeidy Levels



LEARNING TARGETS

- ❑ I can describe types of transformations (dilation, translation, rotation, reflection)
- ❑ I can explain the connection between transformations and 3D graphics.

OPTIONAL PRACTICE

- ❑ I can try out AFrame by remixing a template on replit to show various transformations

HW

DELIVERABLES

- Review reference slides to create a scene that uses AFrame that demonstrates transformations
 - use checklist criteria
 - can use entity from class

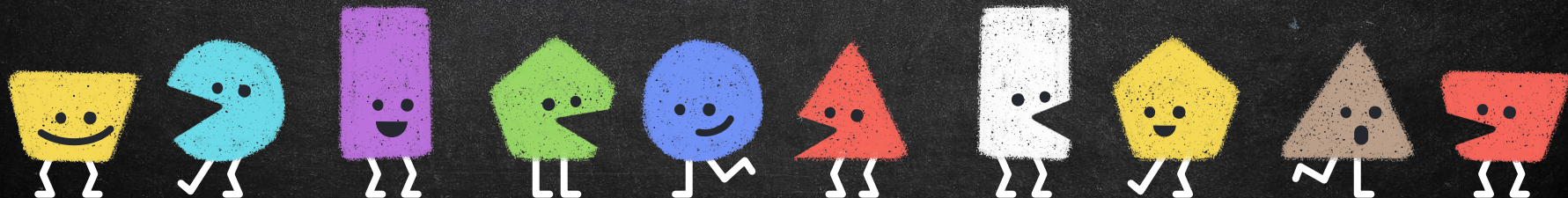
3

1

2

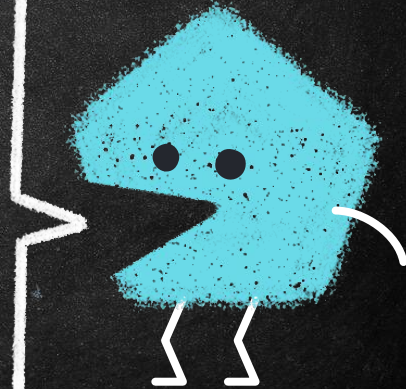
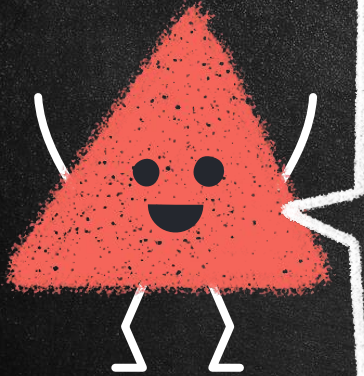
LESSON**LEARNING TARGETS**

- ❑ I can describe 3D transformations and where they might be used.
- ❑ I can decide which type of transformations will work to move one figure to another.
- ❑ I can create an entity using several shapes using Aframe



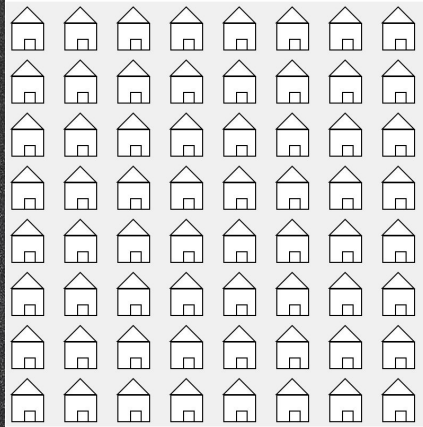
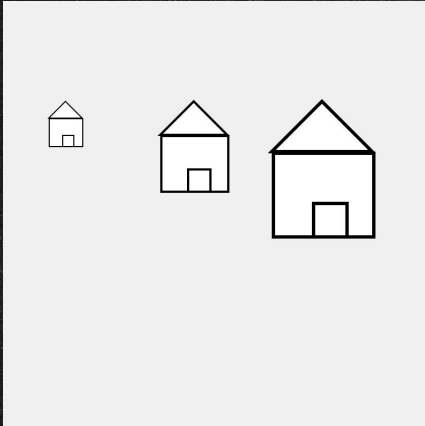
AGENDA

- Quick reminder: 2D Transformations
- Composition of 2D Transformations
- Going from 2D \rightarrow 3D: Understanding Projection and Perspective
- Why Use AFrame (WebVR)?
- 3D Coordinate System
- 3D Transformations
- 3D Graphics with AFrame
- Resources to keep learning
- Homework

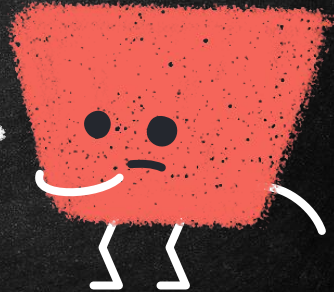


REMINDER: 2D TRANSFORMATIONS

We can use 2D transformations to update existing designs or create, place, and modify copies of 2D designs, like so:

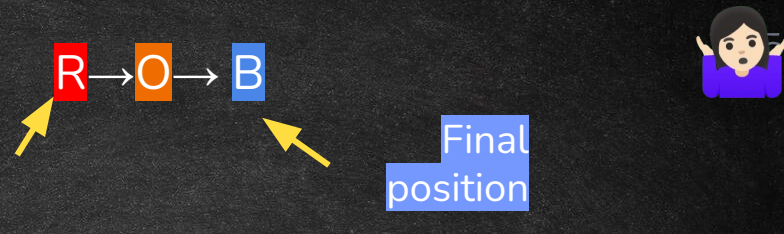


Source for images and notes for more: <https://genekogan.com/code/p5js-transformations/>



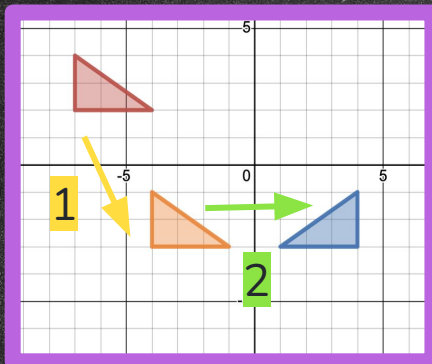
COMPOSITION OF 2D TRANSFORMATIONS

"SEQUENCE MATTERS!"



STEP 1: TRANSLATE

STEP 2: REFLECT



Take gray line 1 stop west

STEP 1:

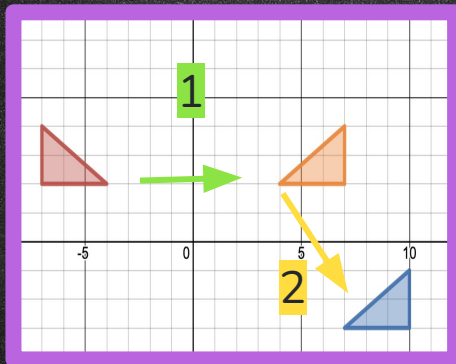
STEP 2:

Go uptown 4 stops



STEP 1: REFLECT

STEP 2: TRANSLATE



Go uptown 4 stops

STEP 1:

STEP 2:

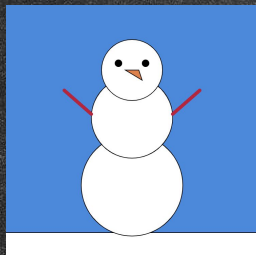
Take gray line 1 stop west



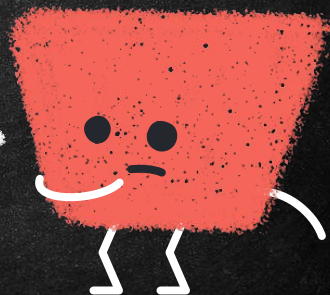
GOING FROM 2D \rightarrow 3D:

UNDERSTANDING PROJECTION AND PERSPECTIVE

Making 2D graphics is great! After all, our screens are effectively 2D.



However, to make something look truly 3D on our 2D screens, we need to understand projection and perspective.

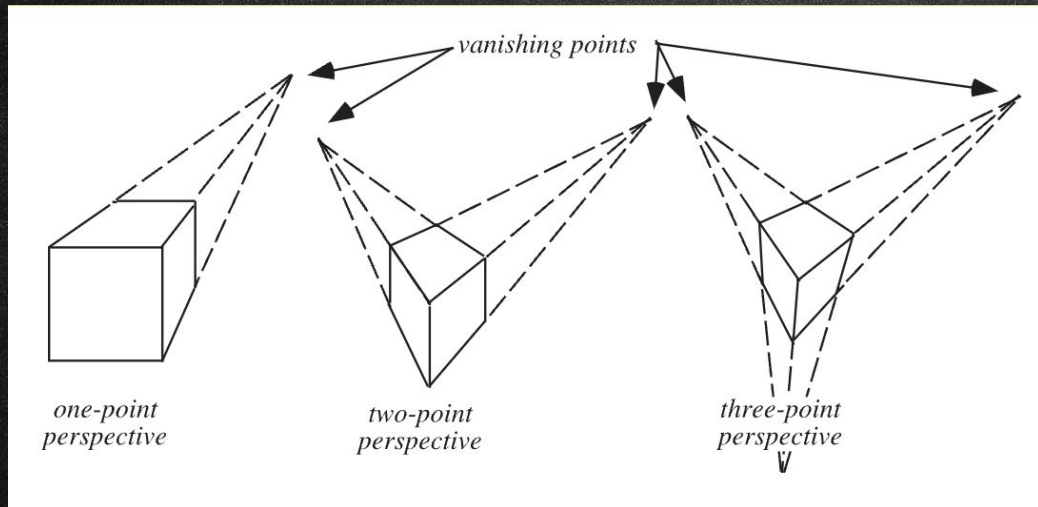


GOING FROM 2D \rightarrow 3D:

UNDERSTANDING PROJECTION AND PERSPECTIVE

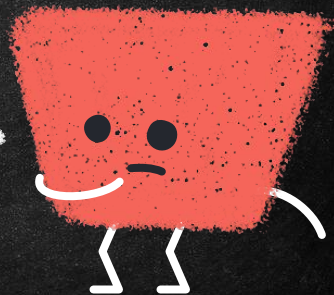
Projection is the method of converting a 3D object into a 2D object.

Perspective is a type of projection!

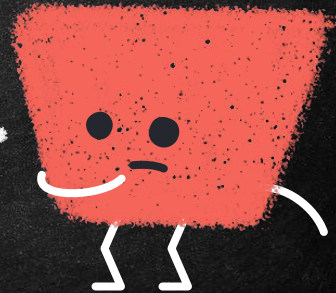
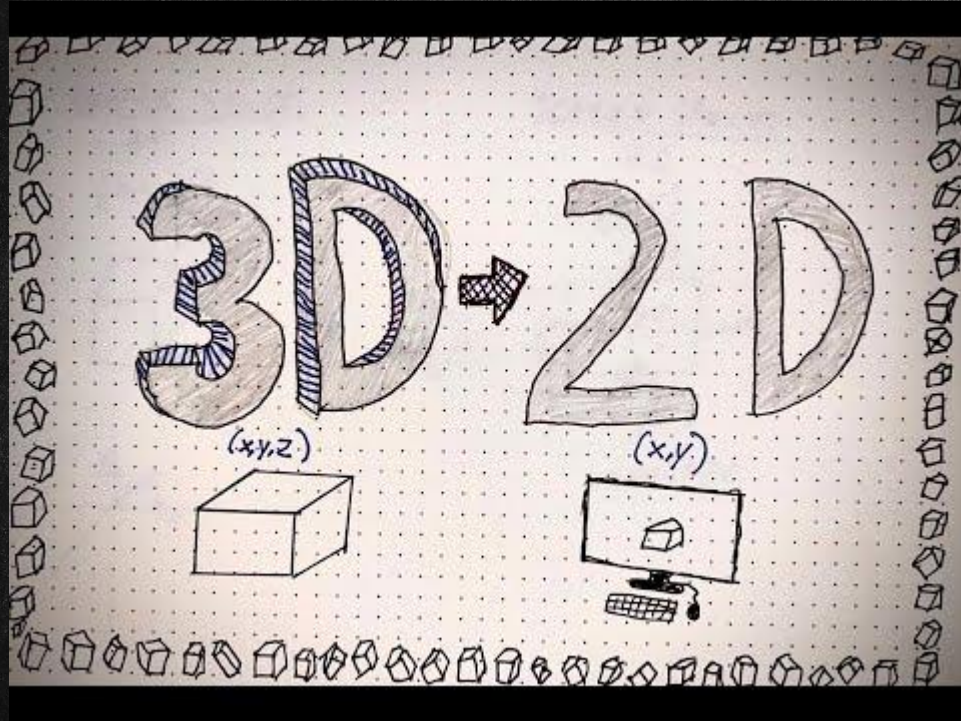


[Image source](#)

You likely learned about perspective in art class.

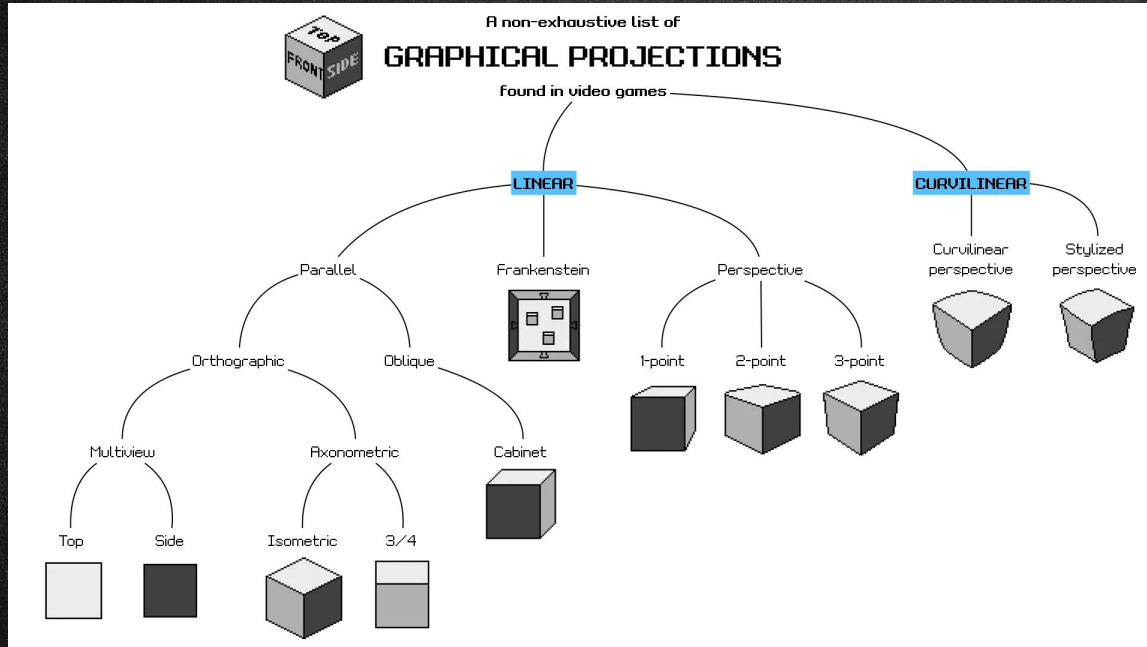


PERSPECTIVE PROJECTION

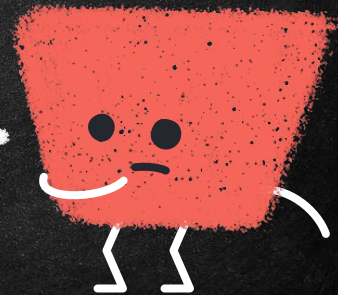


GOING FROM 2D → 3D:

UNDERSTANDING PROJECTION AND PERSPECTIVE



GAME DEVELOPERS GUIDE TO GRAPHICAL PROJECTIONS!



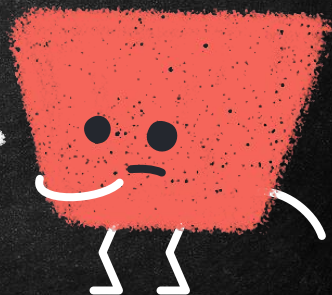
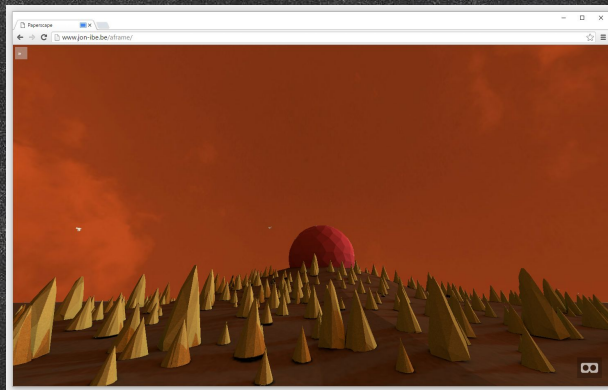
WHY USE A VR LIKE AFRAME ?

Two types of Virtual Reality (VR):

Native

vs.

Web



WHY AFRAME ?

Two types of Virtual Reality (VR):

NATIVE - needs a special device/hardware like OCCULUS
@\$399

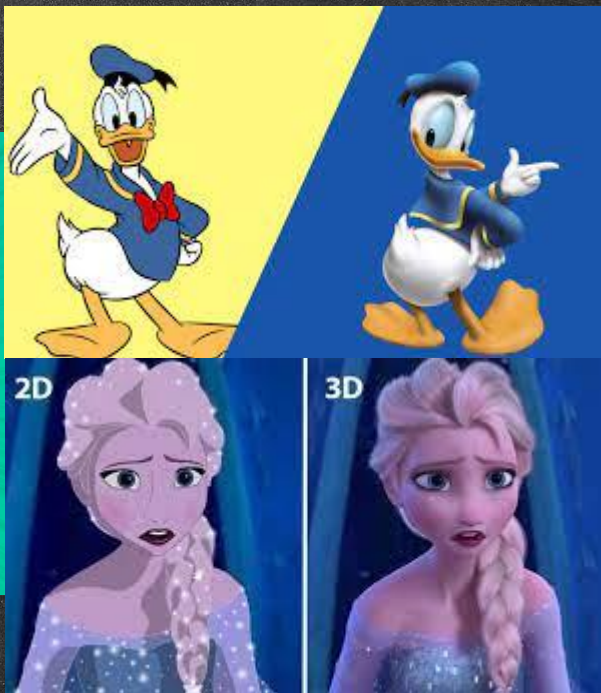
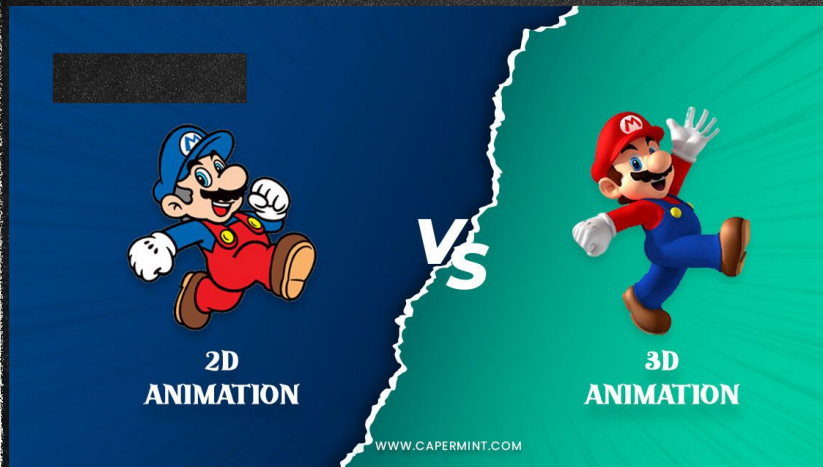
WEB - are applications that run in the browser like AFrame

- Why use WebVR like Aframe ? 00-1:16 [Aframe Webvr - VR Development Part 1 - Setting Up](#)



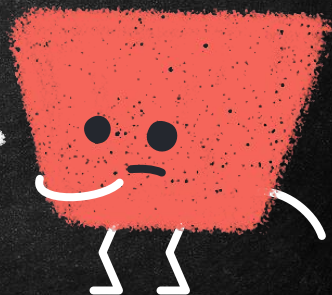
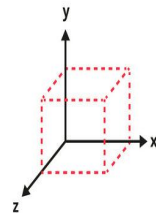
2D TO 3D COORDINATE SYSTEMS

To communicate to the computer what and how we want to draw objects, we need a coordinate system.

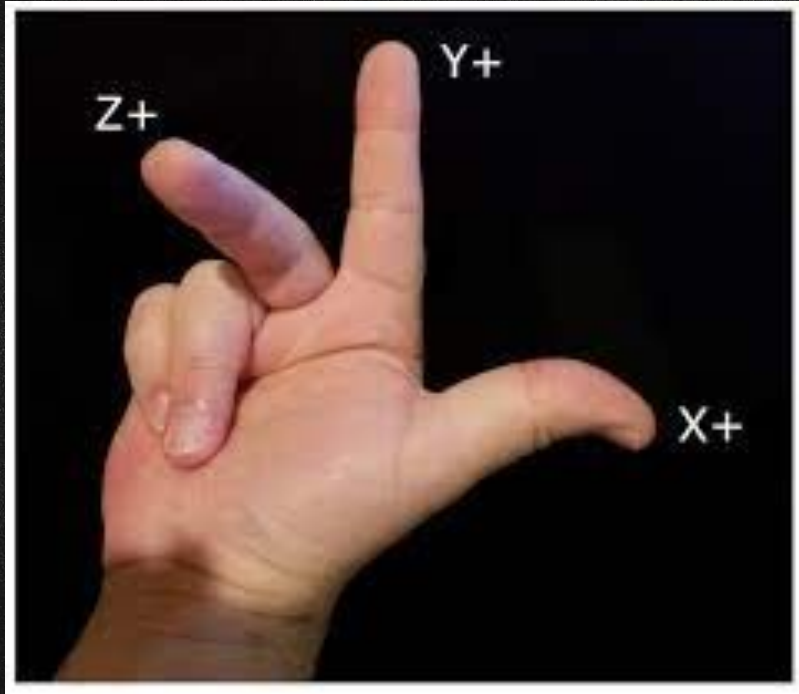


2D Coordinate System

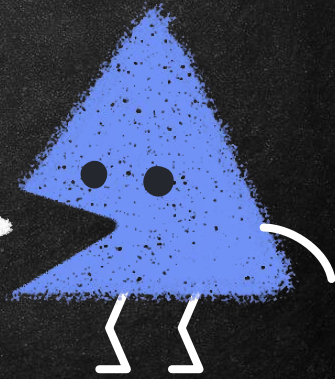
3D Coordinate System



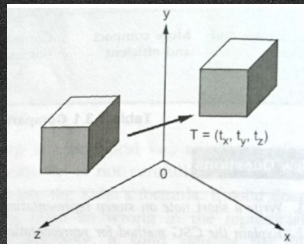
3D COORDINATES: RIGHT-HANDED SYSTEM



[Image source](#)



TRANSLATION IN 3D



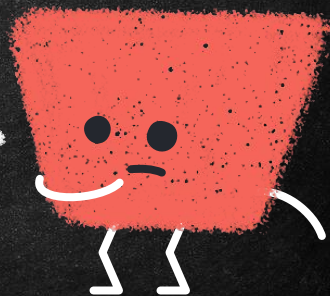
We're still "sliding" just like we did in 2D, but now we can slide in the x, y, and z directions.

We'll use position to translate objects in AFrame.

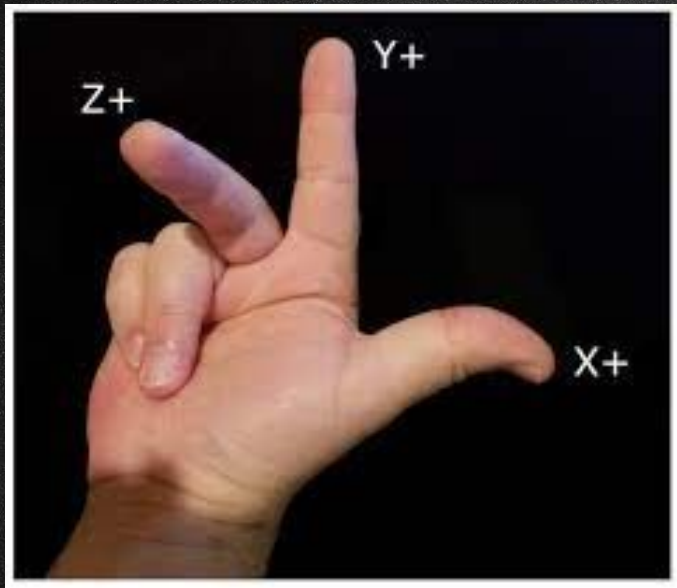
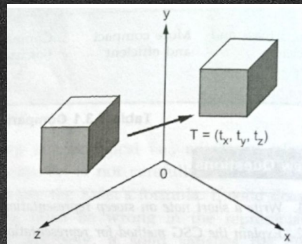
`position = "x y z"`

Default position is 0 0 0, right where we are!

HS math teachers: transformations for graphics are represented with matrices and use homogeneous coordinates.



TRANSLATION IN 3D

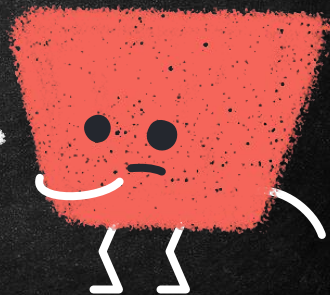


[Image source](#)

Try acting this out....

- Translate your hand in the positive x-direction.
- Now translate your pen in the negative x direction.
- Try for positive y, then negative y.
- Finally try for positive z and then negative z.
- Challenge: Translate your hand to (-5, 5, -5)

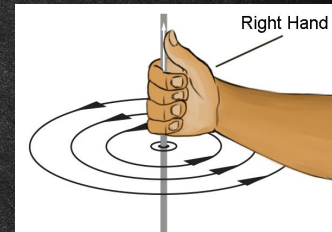
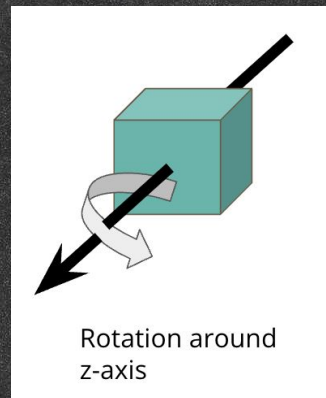
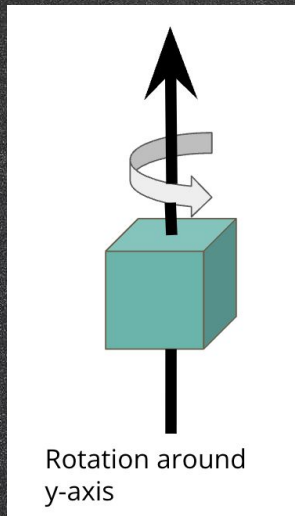
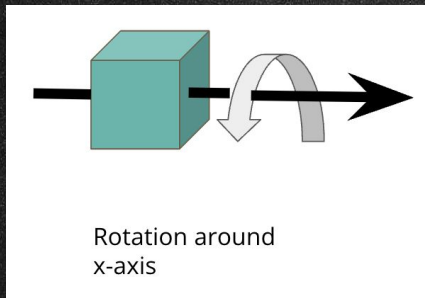
Hint: Use this picture to help you!



ROTATION IN 3D

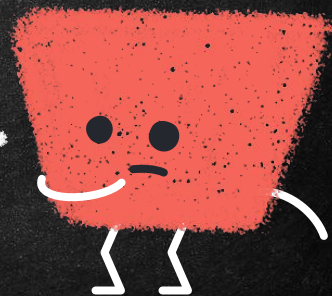
rotation = "x y z"

We can rotate components around the x, y, and z axes. It can help to visualize the axes going through your object.

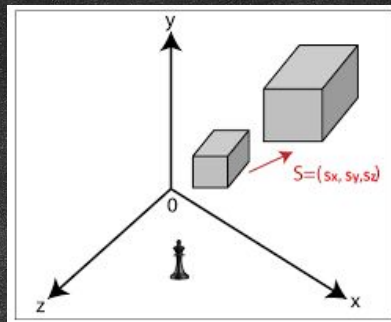


[Image source](#)

You can still use the right hand rule: if you point your thumb along the axis you wish to rotate around, your fingers will curl in the direction of positive rotation.



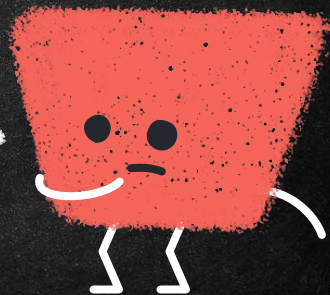
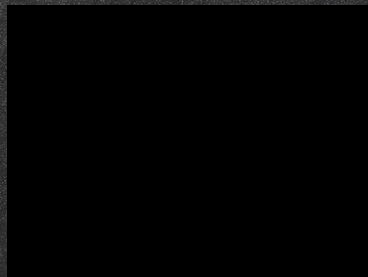
DILATION/SCALING IN 3D



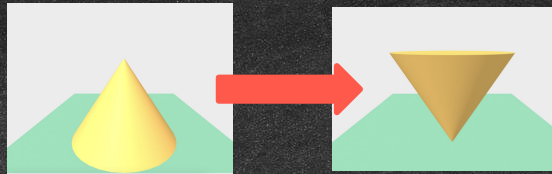
We're still stretching or compressing just like we did in 2D, but now we can stretch or compress in the x, y, and z directions.

We'll use scale to dilate objects in AFrame.

```
scale = "x y z"
```



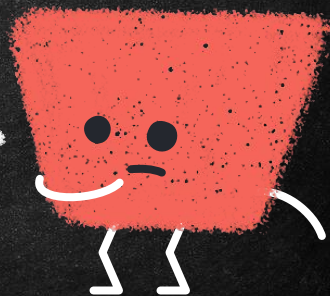
REFLECTION IN 3D



We're still "flipping" just like we did in 2D, but now we can flip in the x, y, or z directions.

We can use scaling to reflect:
set the scale to -1 in the direction you'd like to reflect.

| | <u>x</u> | <u>y</u> | <u>z</u> | |
|---------|----------|----------|----------|--------------------------|
| scale = | "-1 | 1 | 1" | reflects over the x-axis |
| scale = | "1 | -1 | 1" | reflects over the y-axis |
| scale = | "1 | 1 | -1" | reflects over the z-axis |



3D GRAPHICS WITH AFRAME

Starter Code: Code along!

Breakout
Room #1

Reteach of AFrame 1.4



Breakout
Room #2

Independent Work with
Assistance

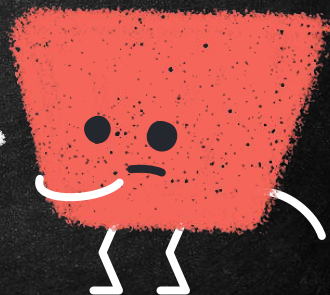
(Create a design that you'll transform with entities)



Breakout
Room #3

Independent Work without
Assistance

(Create your own design that you'll transform with entities)



AFRAME REFERENCE SLIDE

20

If you want to learn more: [PlayList on AFrame](#) (there's so much more to learn!)

