PROJECT ASTRA

(Expect mistakes and whatever in this document, it is not supposed to be public)

(CONFIDENTIAL)

GRAYWATER STUDIOS

AUTHOR:

oshawottice

SECTION 1
INFORMATION

What is Project Astra?

Project Astra is essentially a combination of the best ideas from previous platformer projects abandoned by Graywater adjacent studios. These include, “**Pip Pip Hurray**”, “**Bootleg sonic game**”, and “**Paperventure**”.

Why?

Even though we have attempted to create many platformers in the past, nothing has ever worked out due to a multiplude of factors. Here are some examples from each game.

- **Pip Pip Hurray:** The story was there but there was no gameplay and integrating the gameplay was too hard or not worth it at that point. It had cool cinematics, story, and lots of potential. Because of these reasons, I believe the story was limited because of poor planning gameplay wise so I’d like to reuse or use a remastered version of the story in the new game.

- **Bootleg Sonic Game:** The inverse of the first example, there was gameplay (mostly because of a template but whatever) but no story at all! This just caused the game to fall apart once the first dialogue box started.

- **Paperventure:** Paperventure had it all actually. It had gameplay and a story. It had a map and obstacles and most of the systems were already in place for a fully functional game to work properly. This should’ve worked! BUT IT DIDN’T.

Paperventure had the same flaw as the other games but probably to a higher degree only because its development had lasted longer, fixing the mistakes of its predecessors but not the newly found glaring issue. This being a lack of structure in executing the plans for each game. You see, whenever we started a new game, a waterfall of new ideas

would come down and we would just write everything without taking a while to think about how it will all be executed as a collective. Sure, for making a solo game or a regular game without a story such as Pet's Go, not having this structure will be easier to manage due to the more straightforwardness of each role. However, this is different for a platformer game with a story.

It's easy for amateur developers to come into a game thinking that all they would have to do is make a small change occasionally and a game will make itself eventually, but this has clearly been not working. It takes focus and thought to make something. Examples being "**Bloxenheimer**", "**Bloxenheimer 2 Chapter 1**", and "**Hyper Karts**". What do they have in common? They were completed to a state releasable to the public and were developed with intense focus during a select period.

What about burnout? What about burnout??? This may be an obvious flaw with what I'm saying but the problem I've noticed with my own observations about burnout with teams like this is that people with burnout often have no clue that they themselves have burnout or are just ignorant of it. Paperventure is the craziest example of this as the developers would just "drop out" of the project silently without explicitly telling the other members so. This causes the remaining members to take on a larger load of the remaining work because of the unstructured nature of how work is being done. If you guessed correctly, this sort of creates a death loop where nobody wants to start work on the game again.

I think we were all aware of these issues but did not want to address them directly for one reason or another. Something needs to change, or this game will end up like the others. There could be more reasons, but I believe that, again, structure or lack thereof is the main one.

potential solutions

Of course a 100% sure solution is impossible to find, but I believe there could be ways to mitigate this issue!

1. Knowing exactly who does what from the beginning
2. Updating on everything that happens in the game
3. Equal or close to equal involvement from everyone
4. Dedicated work sessions

5. Regular meetings
6. Define Clear Milestones
7. Be accountable for pitfalls in skills.
8. Have a clear idea backlog, develop for compatibility
9. Prioritize features that are seen as important, not just do the easiest first.
10. Don’t cut corners, it costs the whole team in the long run.
11. Create organized documentation somewhere
12. Get external feedback regularly
13. Always reflect on why previous games did not work.

TLDR: Don’t be fucking lazy alone, plan ur stupid ass so we can take breaks at the same time. (THIS IS TO EVERYBODY)

Section 2

Tools

Isn't Roblox enough?

You think you are able to do everything within Roblox Studio and you are correct for the most part!

The problem arises when you streamline the process of making a game with other people. In not using the correct tools, you are intentionally holding back the entire game. Believe it or not, working with Unions (Glorified unoptimized boolean operations) and many different parts is hell to work with when you're the one making it all work together.

Of course there is a steep mastering curve for most of these tools but honestly the skill level you need to be at to get started only takes a weekend usually. Mastering comes with the job and lots of practice but I am only asking for the minimum for now. Making a game is a huge commitment and if you decide to ever get involved with this you better make sure what you use will help everybody long-term.

List of Tools

- **Roblox Studio:** Map Layouts, general prototyping, scripting, UI Layouts.
- **Adobe Illustrator:** Reusuable vector elements to be able to be scaled up.
- **CSP / Photoshop:** Game art and textures.
- **Blender:** Low-Poly Tri modeling for the game, anything else is extra.
- **Fl Studio:** Music.
- **Chippng:** For transparent images. https://rbxxaxa.github.io/chipng/

Alternative Tools

- **Paint.net:** If you're too lazy to "buy" creative cloud.
- **Krita:** Open source art program.
- **Lmms:** Music but free.

SECTION 3

TUTORIALS

Intro

These will just be basic tutorials, just to get you started in each program (except for Roblox, I'll assume you already know how to use that one, if that's not the case then watch this youtube video I guess:
https://youtu.be/PM6LNc8k7bE?si=2Kkte_gR6MFrycGL)

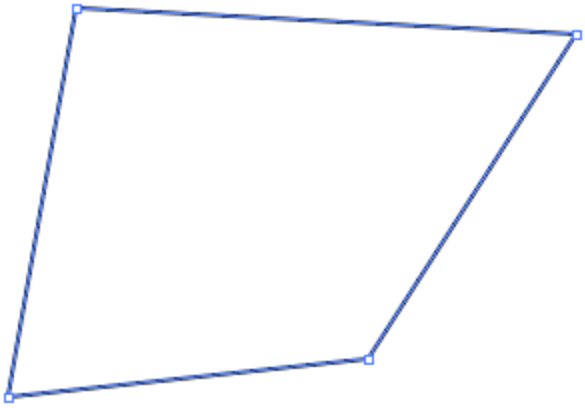
adobe illustrator

This is a vector based program, meaning when you create objects, as opposed to rasters (photoshop), they will not have a set pixel resolution. I don't have the exact definition with me at the moment but that is what it basically is.

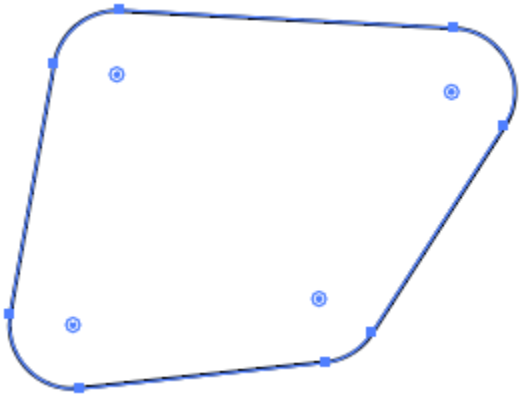
To get started press P or go to Pen Tool:

Connect the dots and and you can form a shape!!

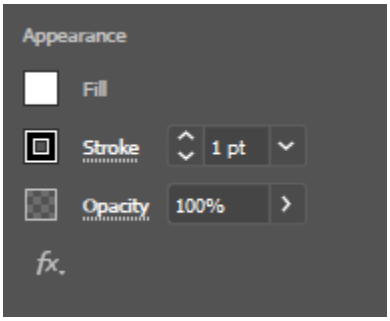
While holding your mouse down you can do simple curves!



Now when you select the newly connected object, you can click and hold down the circles on the edge to make them have a border-radius effect.

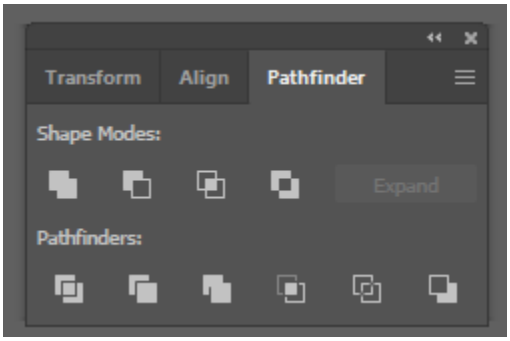


In Properties you could change the fill and stroke



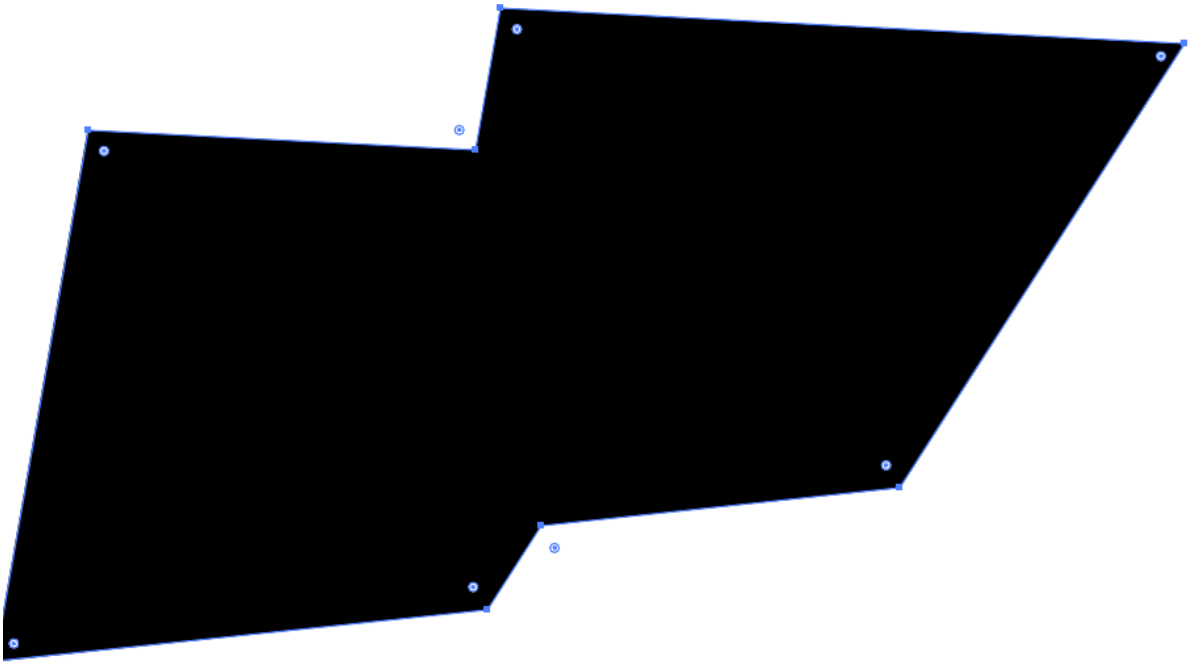
Now it's time for the pathfinder, basically 2D booleans.

If the menu isn't visible, you can find it by going to Window > Pathfinder to open it up.

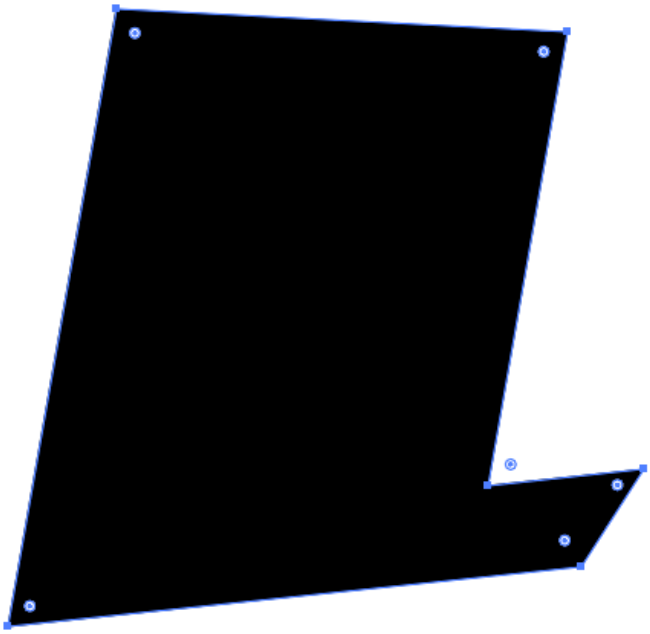


Basically once you have this opened, you can start playing around with it, to select multiple objects at once, you can select while holding LEFT-SHIFT or do a drag select.

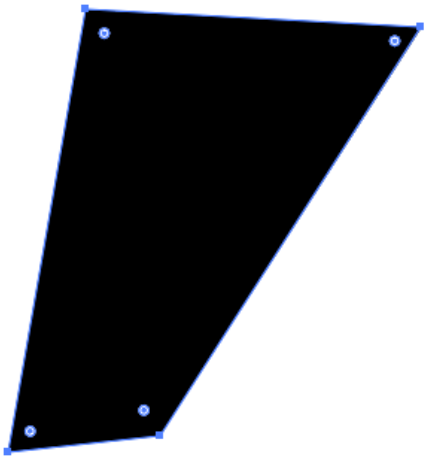
UNITE:



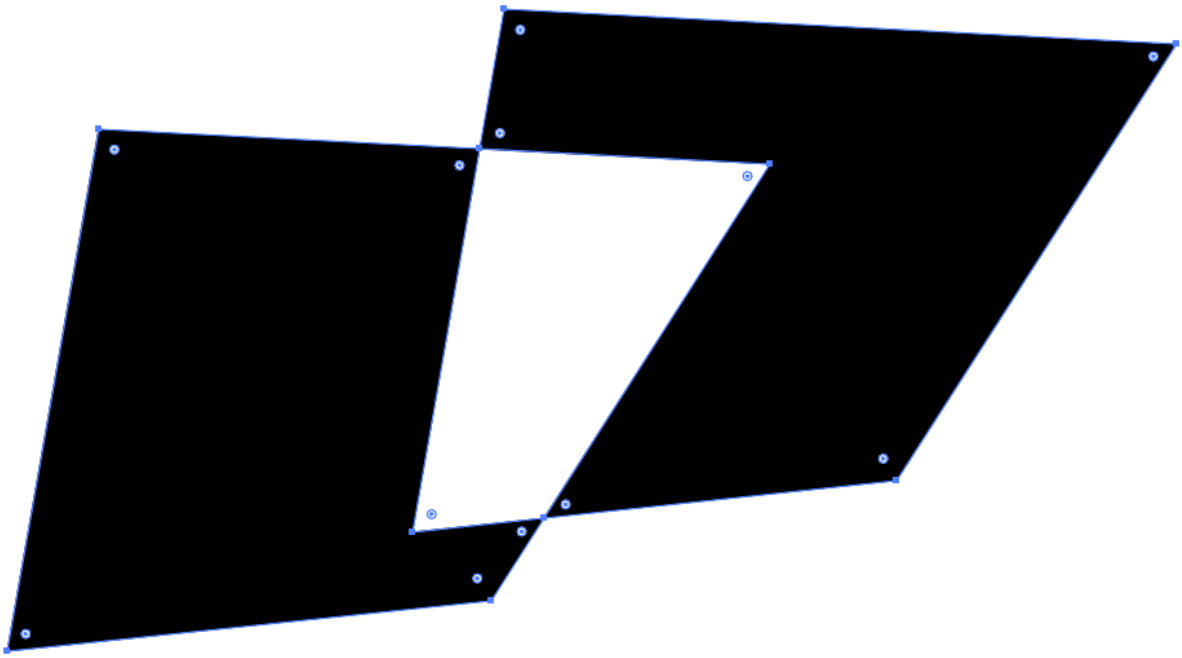
MINUS:



INTERSECT:



EXCLUDE:

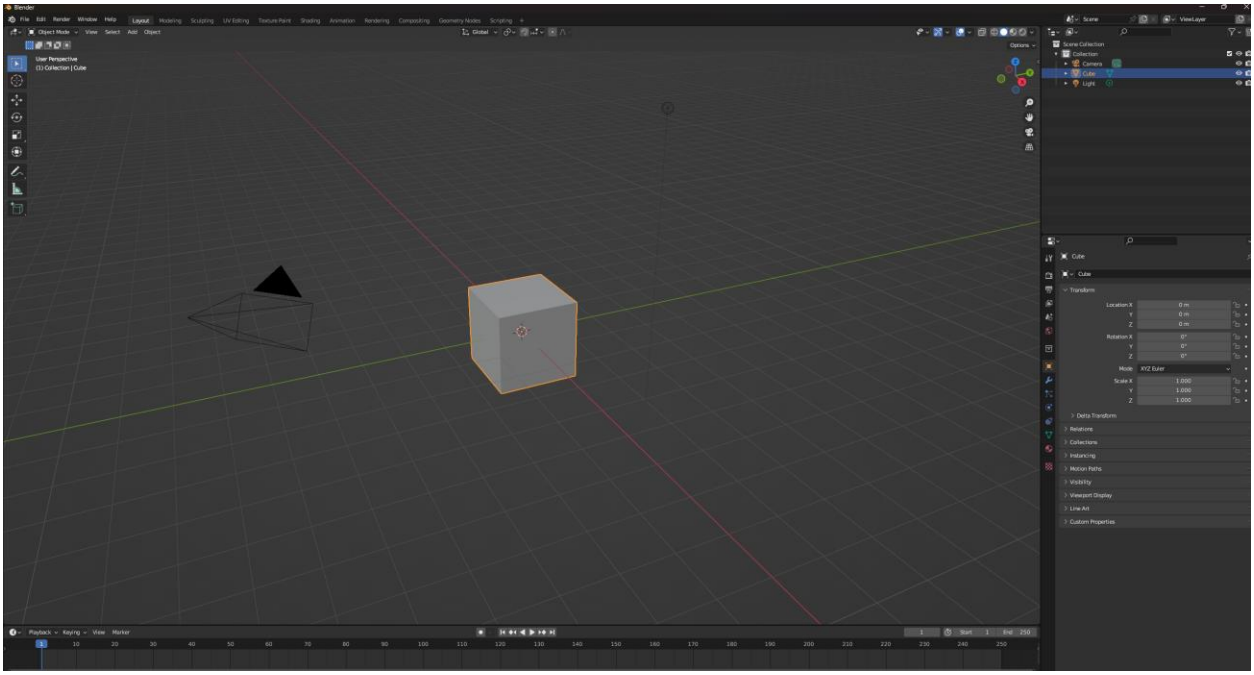


blender

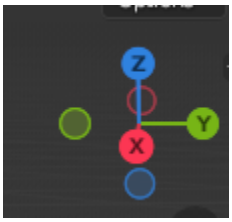
Blender is a program that can do many things. Not just 3D Modeling, it can do Animation, VFX, art, sculpting, and much more, but for this tutorial we're just going to make simple objects and shapes in 3D. This is a heavily simplified tutorial.

When you open up Blender, make sure you have a mouse with a working scroll wheel with you.

This is what it should look like for Blender versions above 2.8.

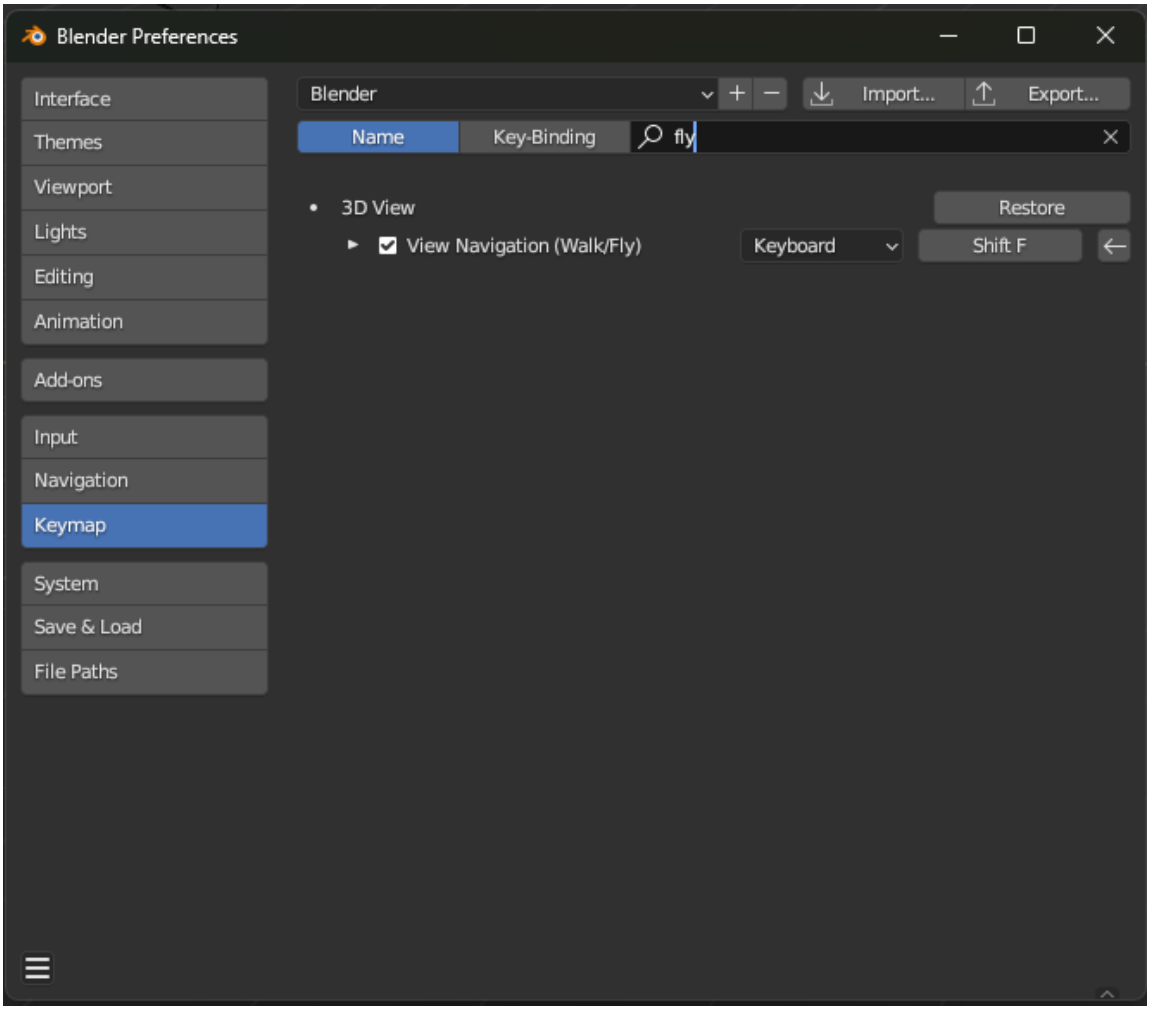


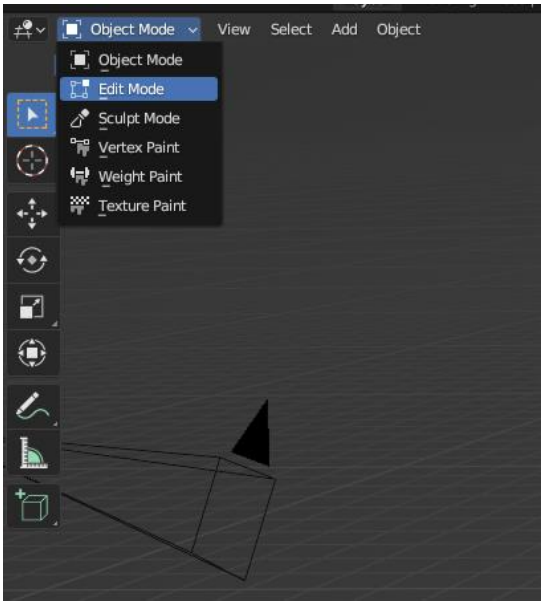
You can hold your middle mouse button to rotate your camera around.



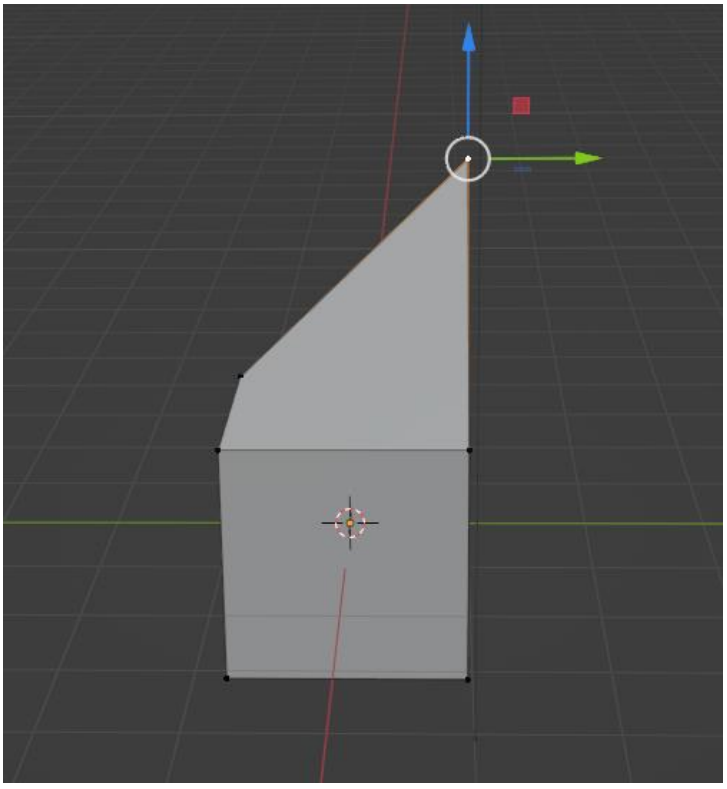
You can also click on the Axis for a straight orientation and stuff.

If you want to fly around, you should check your assigned key combination at Edit > Preferences > Keymap > “Search fly in the search bar”, you could change if you want, I set mine to Shift + F





Now switch from object to edit mode while selecting and object.



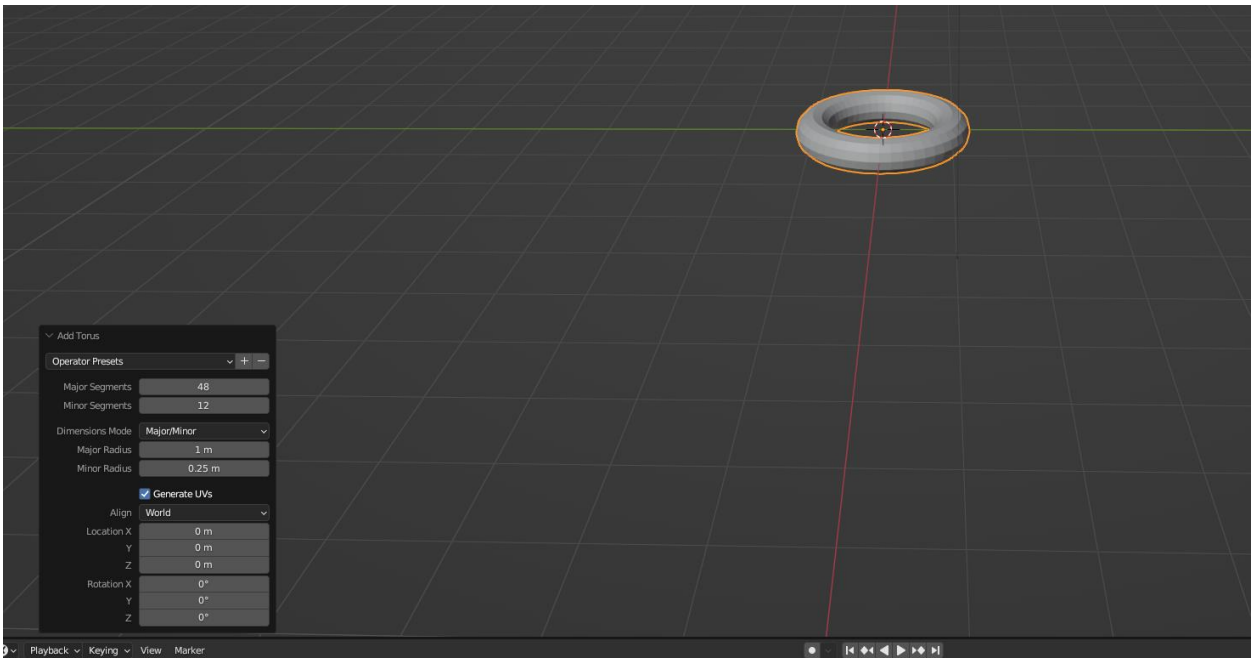
You should automatically be in Vertice selection mode and press A to select all.

The three modes are Vertice, Edge, and Face.

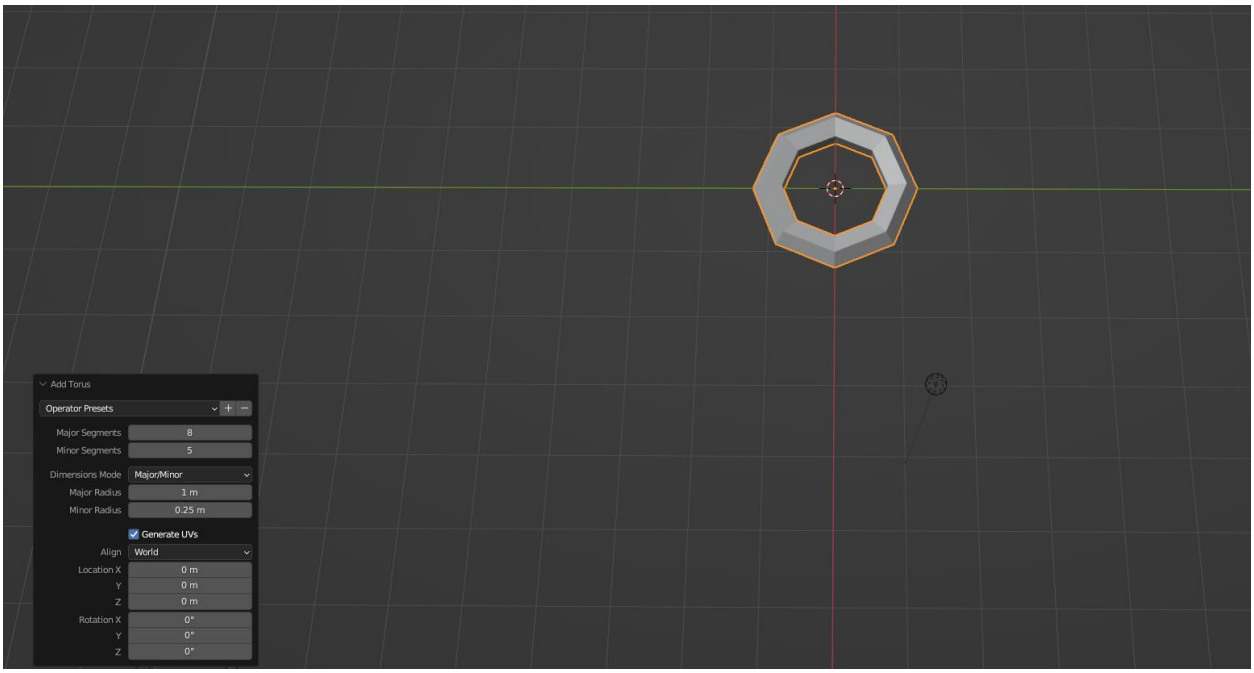
Now while selecting anything, in edit mode, if it's a face, edge or vertice it will give you options on what to do? I suggest you mess around with that first.

In object mode it will just prompt you to delete the object.

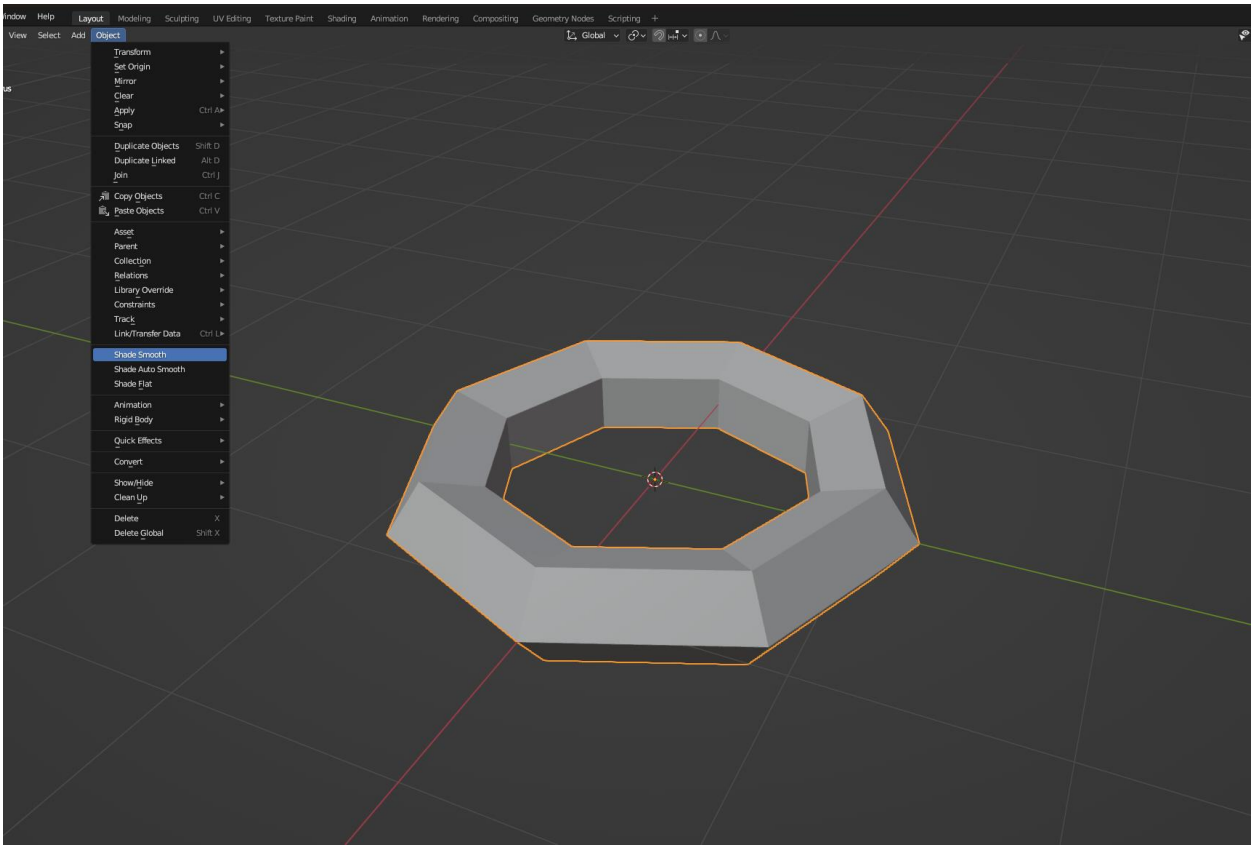
Once deleted press Shift+A to create a new object. You will have many options but for now you should just do mesh and pick something.



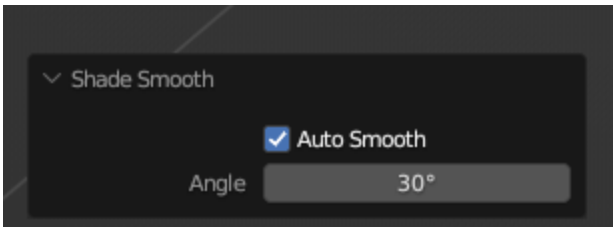
I generated a torus but it has too many triangles, we can change the segments by opening up the menu that pops up on creation on the bottom left.



Much better!



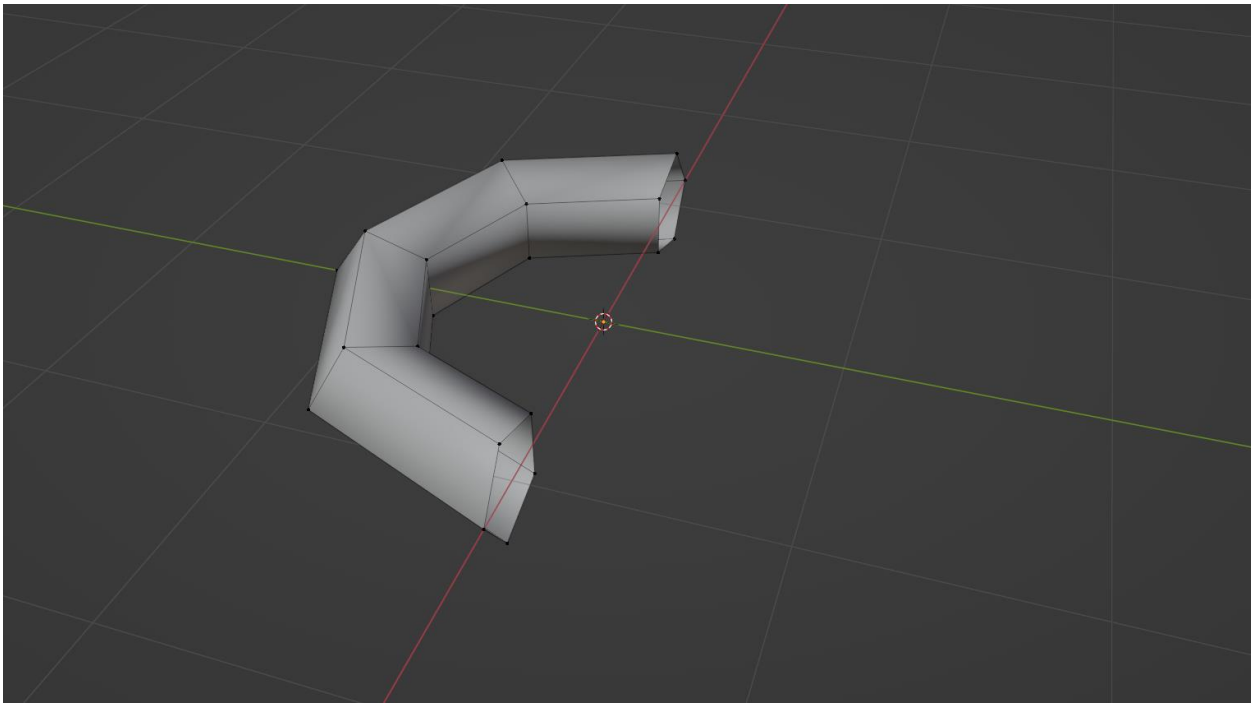
Now to make it smooth!:



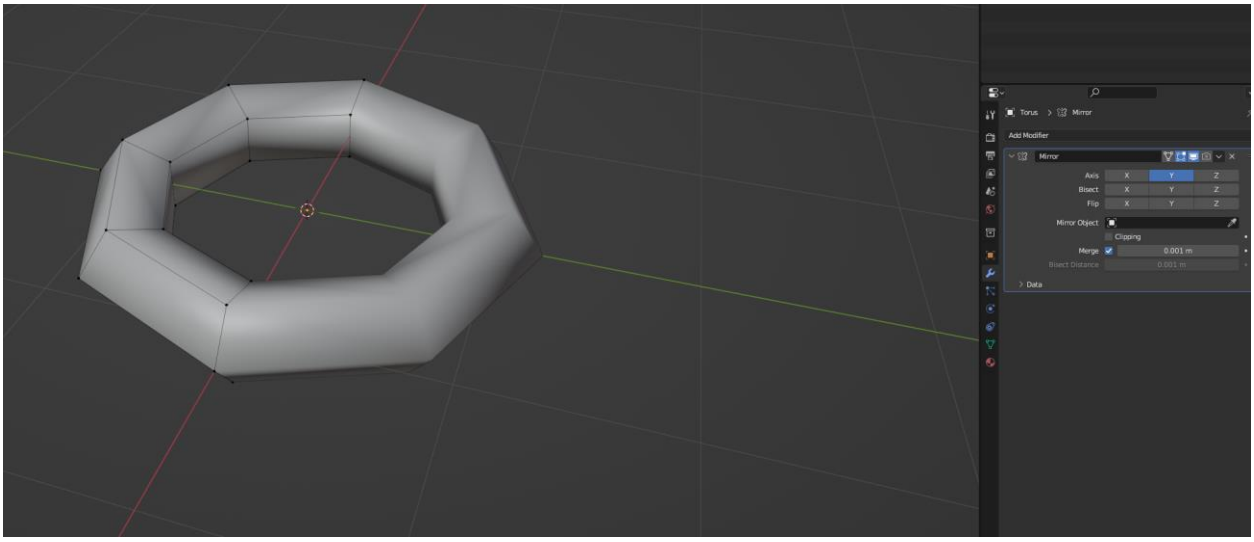
Auto Smooth is an amazing option if you're feeling lazy.

Now it's time for modifiers. In layman's terms, they are a set of operations that will help you do things quicker.

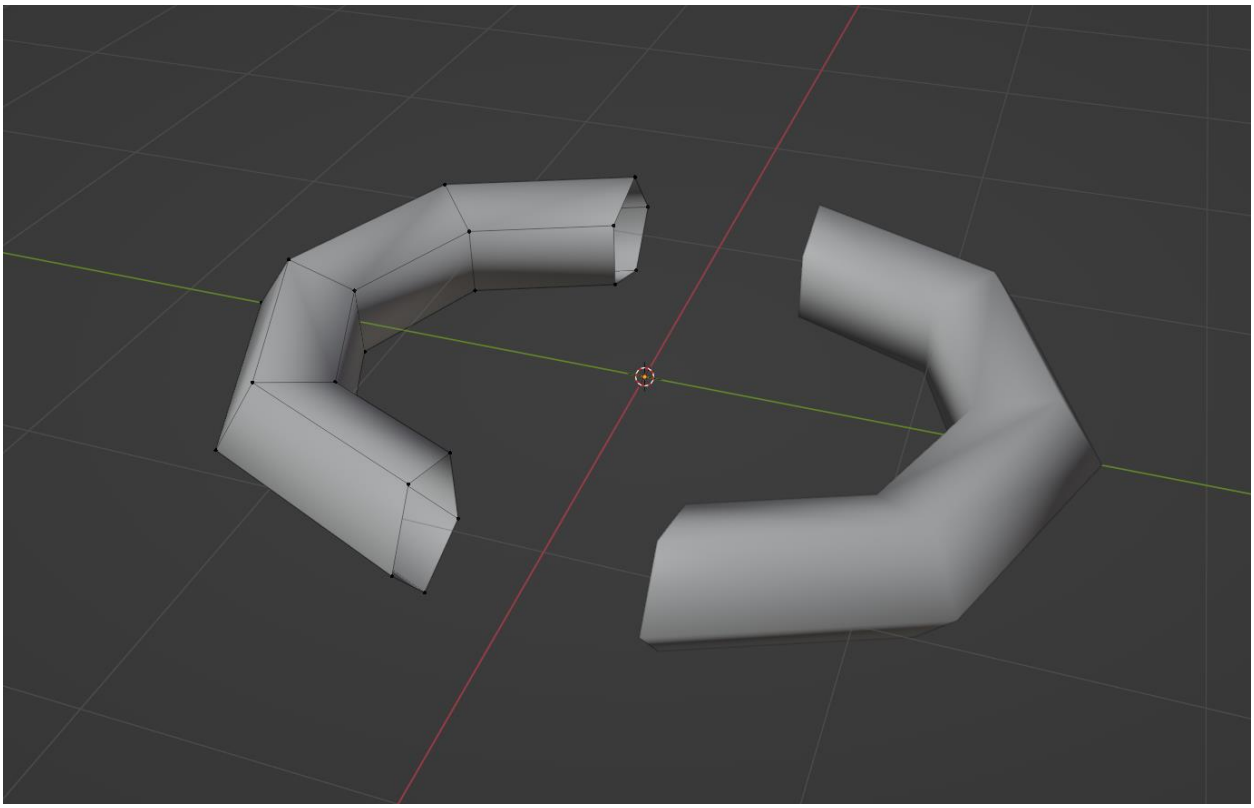
The modifier you'll probably use the most would be the mirror modifier:
It just mirrors from a center point.



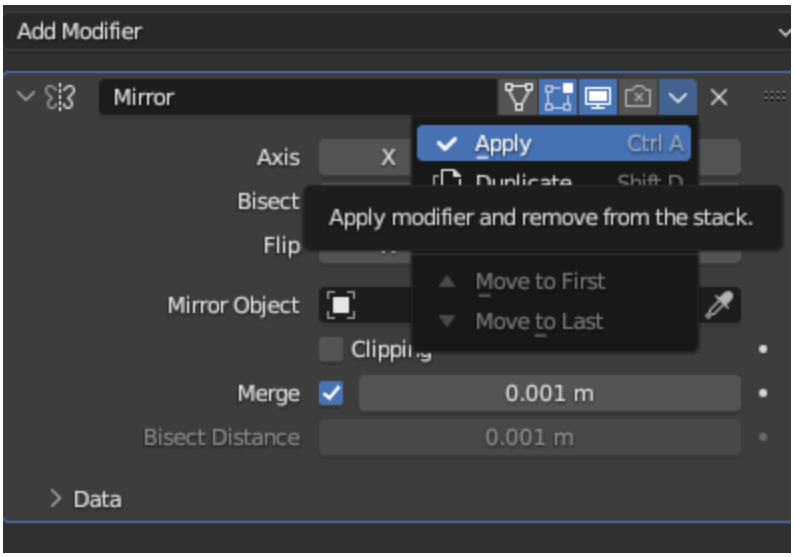
Let's say I cut my torus in half.



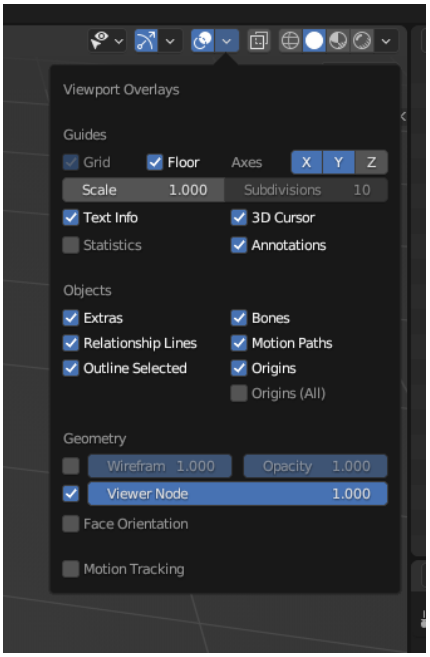
On the side menu I go to the wrench icon and add a Mirror Modifier.
I select the right Axis and it becomes mirrored!



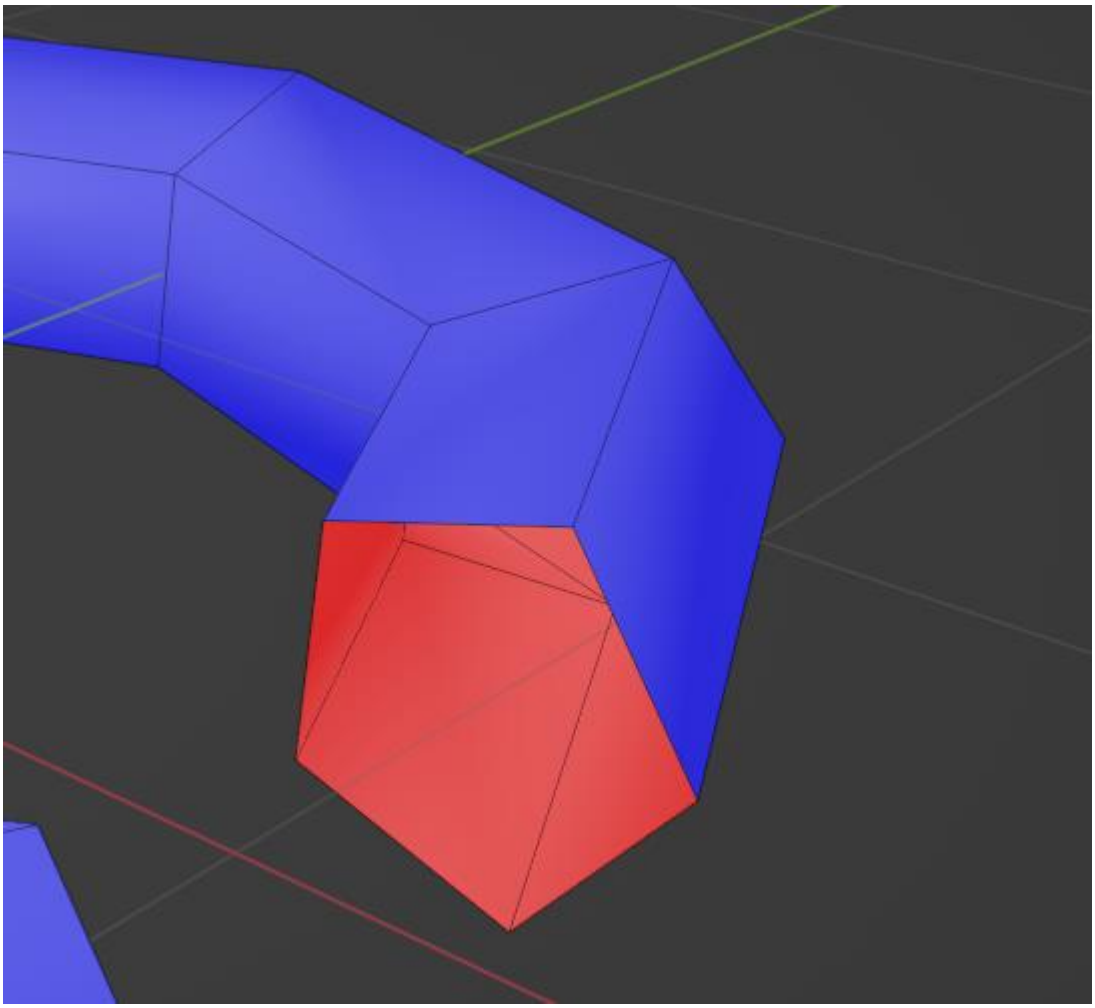
You should be careful however because if the vertices are not perfectly centered to the middle of the object, it will not be connected.



To apply the modifier, go back to Object mode and press Apply.

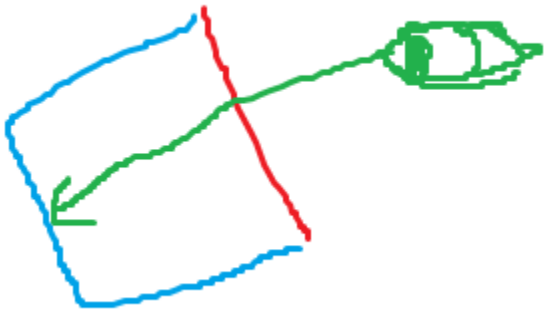


Whenever needed check on Statistics to see how many triangles you got and Face Orientation to see if your faces are the right way.

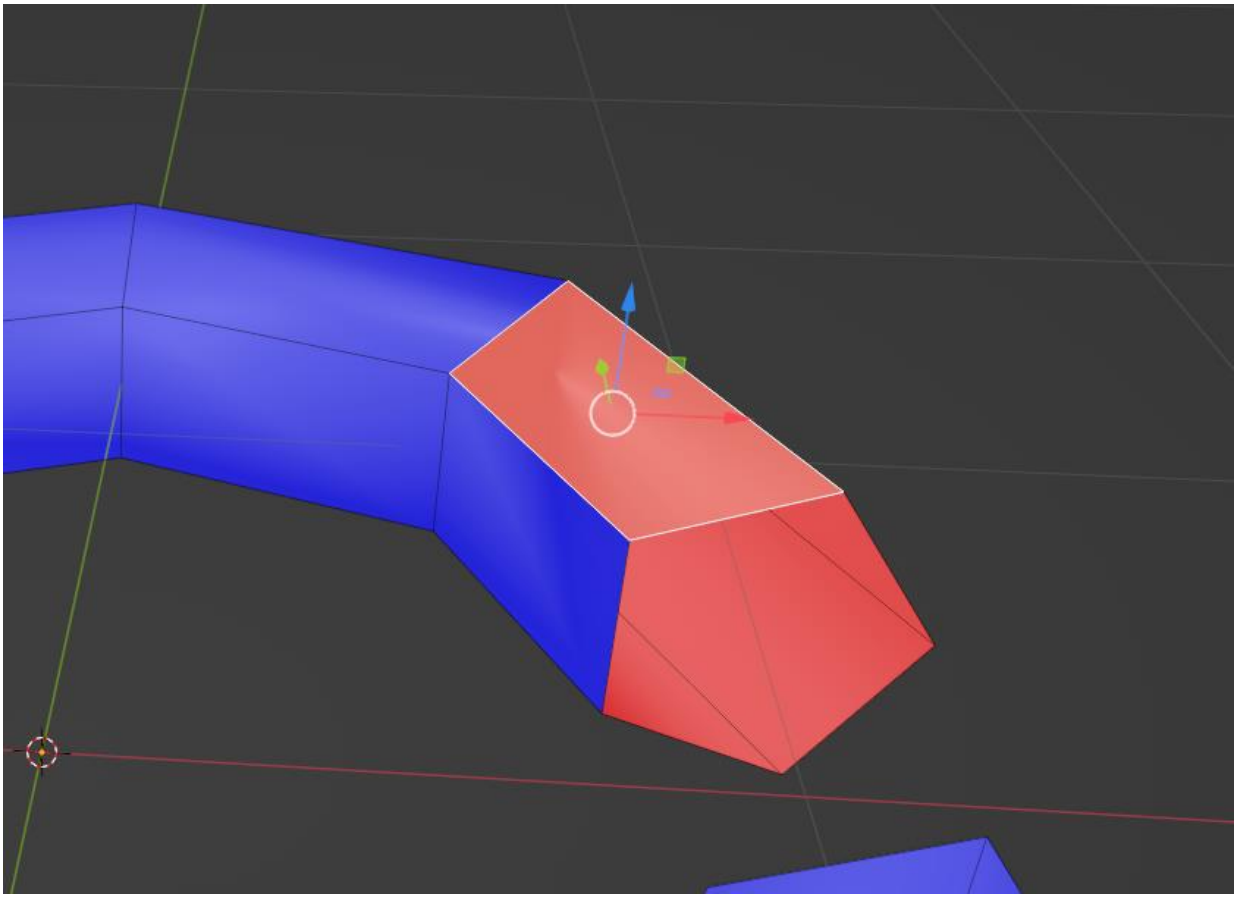


Each face has a outward and inward appearance by default

As you can see from the torus example, Blue = outward, red = inward, if the outside was red, the inside would be blue and it would have the effect of looking through a mesh



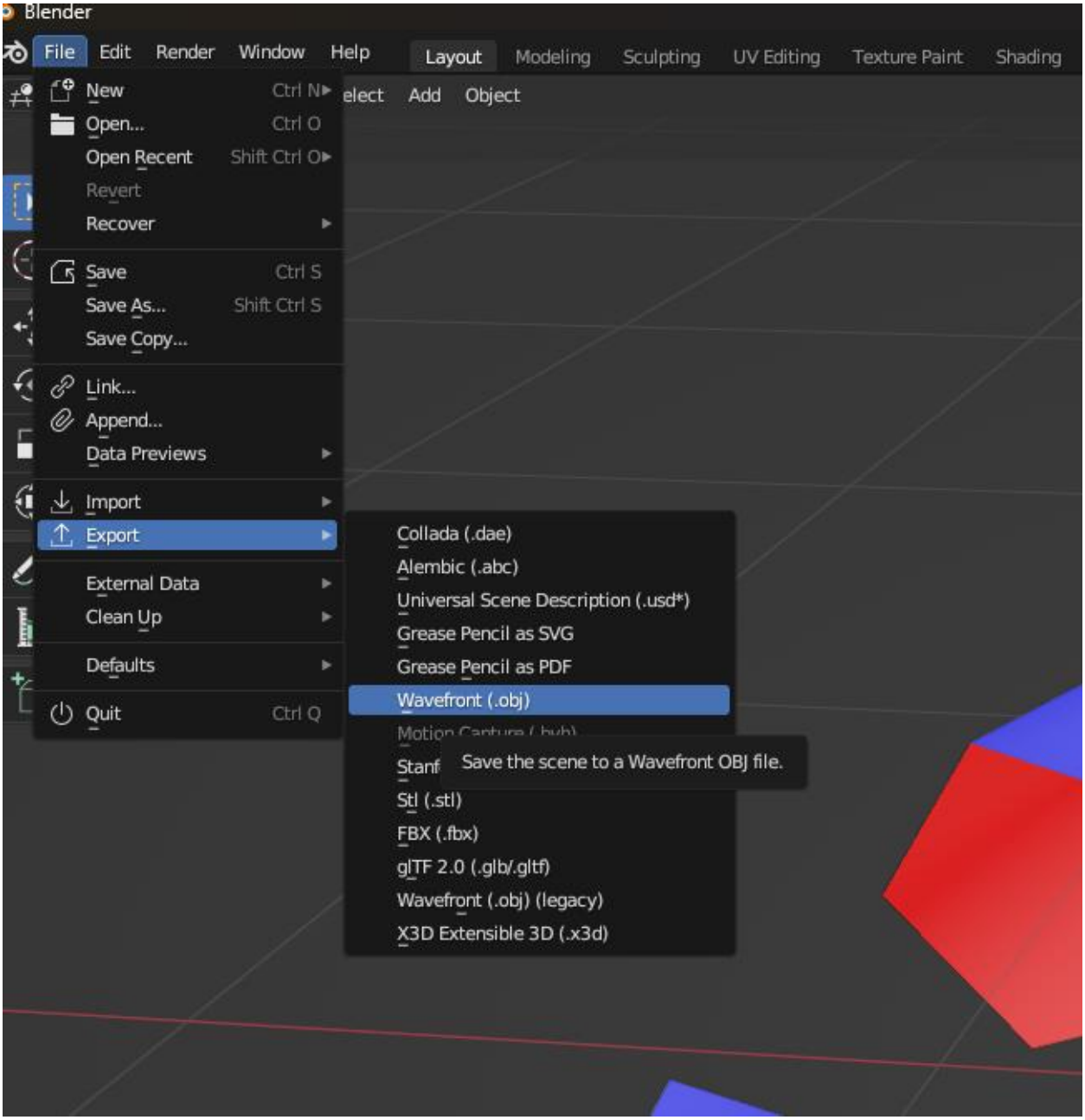
Here is a crude visualization, however you'll understand if you ever make the mistake.



Let’s say you have what is supposed to be an outwardly facing face in the wrong direction.

While selecting that face, you could do ALT+N and click on Flip.

If you’re lazy you could try selecting all faces and click on Recalculate Outside instead of Flip or inside but I find that less accurate as a computer never has a 100% chance of knowing your intent.



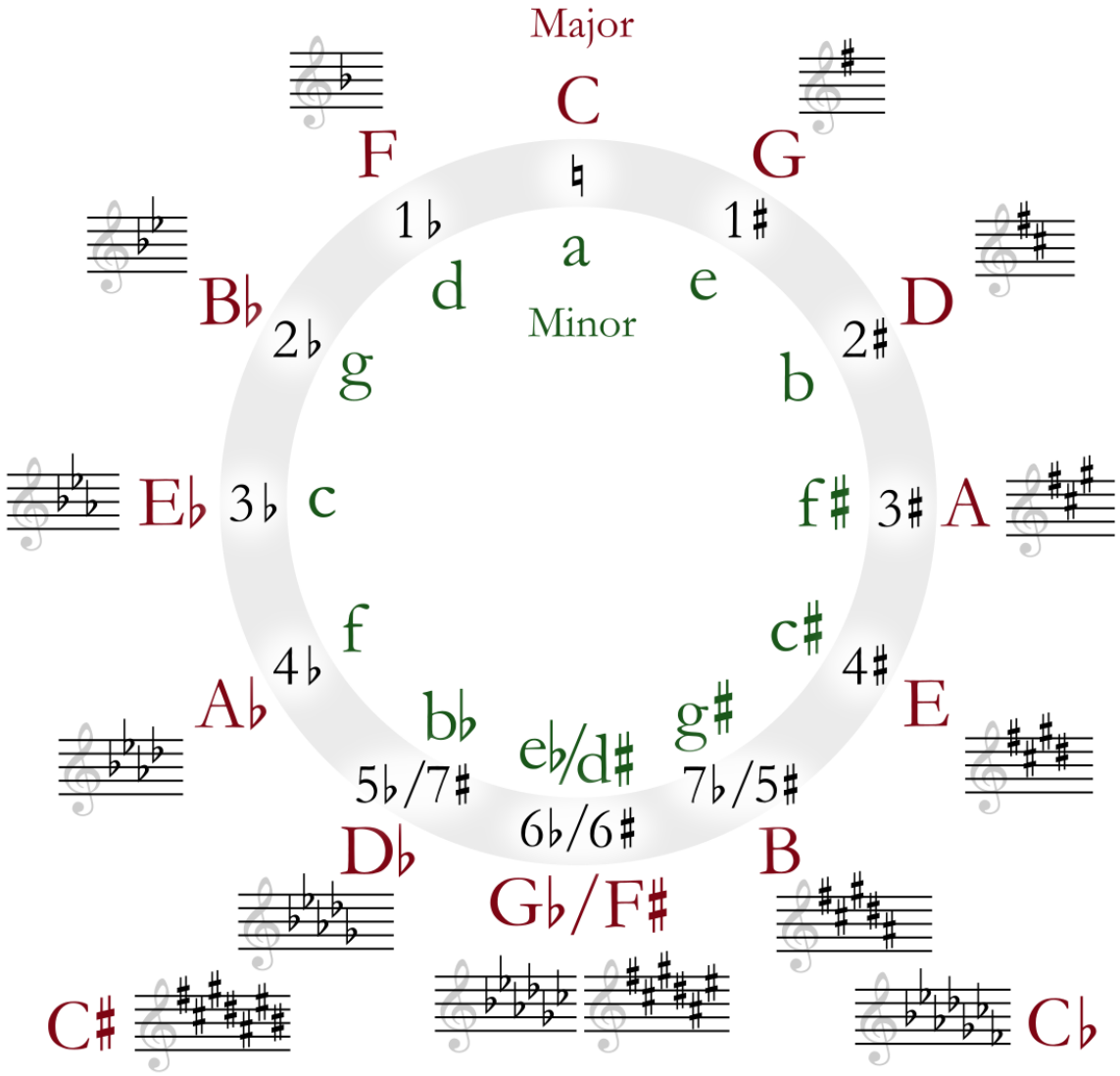
If you want to export, just export as obj or fbx.

For static objects without vertex paint or special things, just use obj.

Other Tools

For art and music tools, I believe that I am not qualified enough to teach you. If you're getting hired for these skills, I believe that you would already know more about your specific program than me.

If you're doing music heres a reminder.



SECTION 4

GAMEPLAY

warning

The gameplay is very work in progress, anything can be changed at any time.

SECTIONS

STORY

warning

The story is very work in progress, anything can be changed at any time.