

Computational Fluid Dynamics Simulation of a 2D Airfoil (NACA0012) in ANSYS Fluent

Description: A Structured Manual for Beginners on 2D CFD Simulation

Keywords: *Aerodynamics, Airfoil, CFD, Fluent, ANSYS, NACA0012*

This is written as a log for my future self, and for any poor soul who decided to walk in the less trodden path of ANSYS Fluent. This is a beginner friendly module that will be helpful for students, instructors and of course, my future self who is prone to forgetting heh!

The module will walk through setting up your first 2D airfoil simulation in ANSYS Fluent. If you don't know anything about ANSYS, then maybe you should read some other...

Just kidding. Stick around. When I started, I didn't even have ANSYS installed (obvious lie, but works for the dramatic effect, no?). This module assumes you have absolutely no prior knowledge about Fluent, or ANSYS in general.

ANSYS Fluent* for Dummies, if you will.

(*Conditions apply – We are just learning a 2D simulation for an airfoil. I know you want to design your own flying object or land a job at F1 as an aerodynamicist. May be someday I will write about those, given I land a job at SpaceX or Mercedes AMG Petronus F1 team – which seems bleaker and bleaker the more day passes to be honest!)

Regardless, the basic principles that you will learn will translate to a better understanding of Fluent, ANSYS and CFD in general. Who knows, may be one of you will land a job in those organizations, while I'll be drinking my tea and contemplating on which color of Lego set I want to buy – red or blue?

The module will be divided into 2 parts –

- Theory – What Exactly is Computational Fluid Dynamics
- Sessional – Decoding the Clunky UI of ANSYS to Set Up the Airfoil

Let's get started, shall we?

Theory: What is Computational Fluid Dynamics

So, you want to be (and think like) an aerodynamicists? You want to wow yourself (and later, anyone who dares to be drawn into your madness) with real world physics of how fluid (we

will only discuss about air for now) acts around an object? You want to plot fluid mechanical coefficient like it's nobody's business?

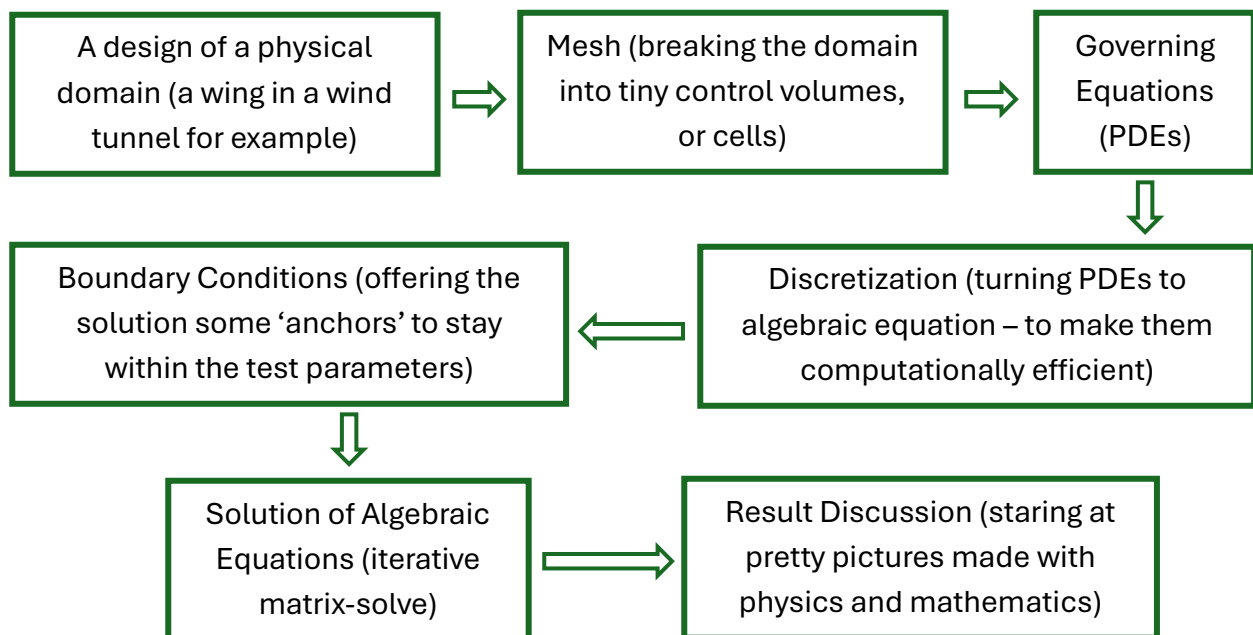
Welcome to the sacred world of Computational Fluid Dynamics – where objects show you how good (or bad, no judgement here) they are in fluid loads, and where velocity vectors paint Bernoulli's equations and Navier-Stokes equations like Vincent van Gough.

Imagine an airfoil cutting through air. There will be various forces in and around the airfoil acting based on the classical mechanics principles (Sir Issac Newton). This, in turn, will result in forces like drag and lift – the bread and butter for any (and all) fluid dynamic analysis. But since there are a lot of component involved in such process, figuring out these key values (that will help you improve the aerodynamic performance) with formulas is a daunting task. Who has the time to do a few million calculations every time your object makes a slight adjustment in its position?

Introduce Computational Fluid Dynamics, the prodigal son of Finite Volume Method (or Finite Difference Method), a numerical tool that you've probably learned in college and wondered why am I even learning this?

CFD is a numerical method designed to solve physical problems by dividing a complex domain in smaller chunks (mesh). Under the hood, it is basically a differential equation solver that tackles the Navier-Stokes equations by discretizing (turning PDEs into algebra) and solving with iteration.

Here is a 'hack' version of what CFD is –



Congratulations. Now you know what CFD is better than your classmate who got a 4 in her fluid mechanics course but has a visible twitch the moment she sets foot in the uncharted domain of ANSYS.

Which is a completely useless skill to have, but you still have it.

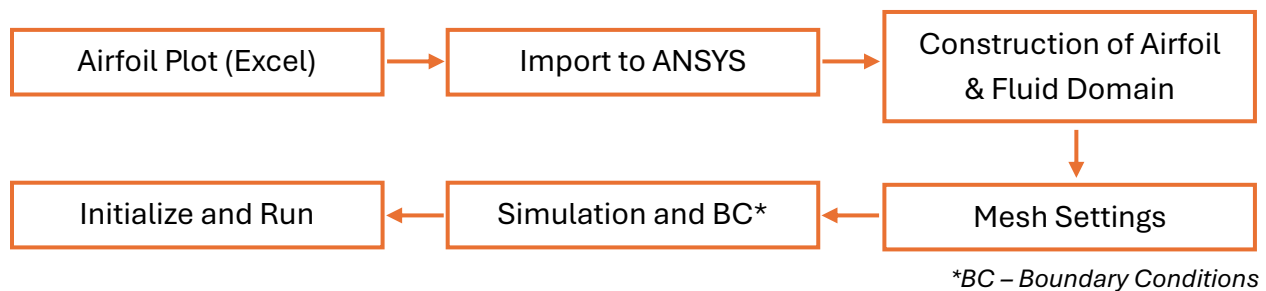
And I am going to stop right here before it turns into a lecture. Let us see how exactly we can set up our own model for a complete simulation run.

We will be using the NACA0012 airfoil in this example. But anything else works. The module, as stated already, will be focused on 2D only, with varying AoA.

(I made a [Python script](#) for AoA manipulation in the ANSYS fluent simulation domain)

Sessional: Let the Fun Begin

Before we start with the nitty-gritties, let's take a look at the overview of the process, and then we can start building our own simulation shenanigans from the ground up.




Please note, the result discussion is a separate entity and will not be discussed in this module. There is a separate section for results in my website, you may find it [here](#).

Let us start, shall we?

Excel (Airfoil Plot)

There is a website (search 'airfoil tools'), where you can get a plot (in a cartesian co-ordinate, X, Y values) that represents all the standard NACA airfoils. The following screenshots are helpful while setting up for a particular airfoil profile –

NACA 4 digit airfoil generator (NACA 0012 AIRFOIL)



Max Camber (%) First digit: 0 to 9.5%

Max camber position (%) Second digit: 0 to 90%

Thickness (%) Third & fourth digit: 1 to 40%

Number of points 20 to 200

Cosine spacing ☒ Cosine or linear spacing

Close Trailing edge ☒ Open or closed TE

[Send to airfoil plotter](#)
[Add to comparison](#)
[Add to My airfoils](#)

Dat file

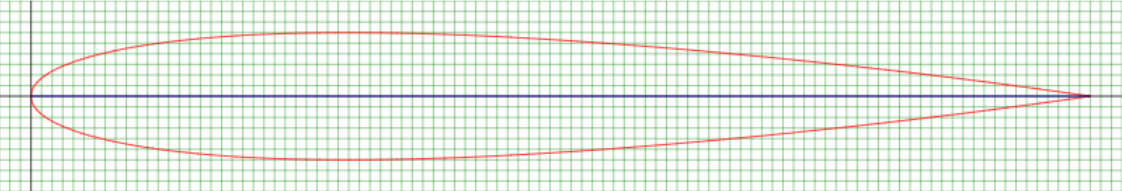
```

NACA 0012 Airfoil M=0.0% P=0.0% T=12.
1.000000 -0.000000
0.999753 0.000036
0.999013 0.000143
0.997781 0.000322
0.996057 0.000573
0.993844 0.000891
0.991144 0.001280
0.987958 0.001737
          
```

These values will depend on the 'xxxx' number of the NACA profile. Please note: max camber position should be 40% for the NACA2412 (as an example) profile, not 4%.

Open trailing edge will cause issues in mesh and results

Once the 'plot' is complete, click 'Send to airfoil plotter'.



Name = NACA 0012 Airfoil M=0.0% P=0.0% T=12.0%
Chord = 1000mm Radius = 0mm Thickness = 100% Origin = 0% Pitch = 0°

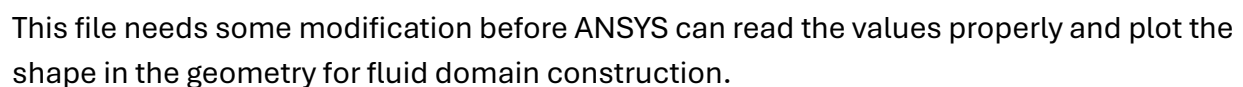
[Open full size plan in new window](#) [Open paginated plan in new window for multi page printing](#)
[3 CSV file of coordinates](#) [SVG image as text file](#)
[Restore defaults](#) [Download PDF file](#)

Airfoil	user-000 - NACA 0012 Airfoil M:-	Choose from database list or add you own airfoils here .
Chord (mm)	1000	Chord width in millimetres. (1 inch = 25.40mm)
Radius (mm)	0	Radius of camber in millimetres, Zero for no curve
Thickness (%)	100	Thickness adjustment. 100% is normal thickness. 50% is half. 200% is double
Origin (%)	0	Adjust the position of the origin e.g. 50% is mid chord
Pitch (degrees)	0	Pitch or angle of attack. 180 flips the plot
Halo (mm)	0	Line parallel to airfoil for wing covering or jig. Negative values are external, positive internal.
Halo (mm)	0	Second line parallel to airfoil as above
Colour	Colour	Colour palette or black & white
Line thickness (%)	100	Scale the line thickness (10% to 500%)
Reverse	<input type="checkbox"/>	Plot a mirror image
Data box	<input checked="" type="checkbox"/>	Print the airfoil data on the image
Camber line	<input type="checkbox"/>	Show camber line on image
X grid (mm)	10	X grid size in millimetres
Y grid (mm)	10	Y grid size in millimetres
Paper width (mm)	280	Used for printing plan. A4 landscape approx 280mm
Paper height (mm)	180	Used for printing plan. A4 landscape approx 180mm
Plot		

Change the chord length to your specification value (1), plot (2), and click 'CSV file of coordinates' (3).

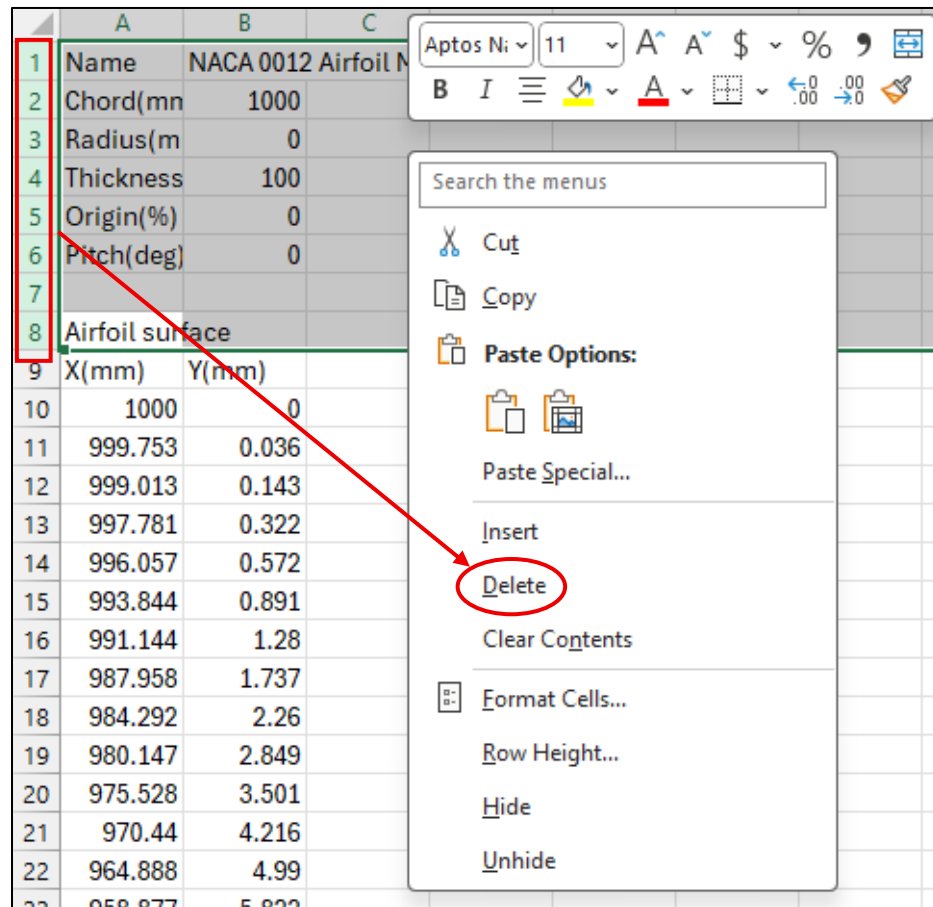
Your plot file (Excel) is ready. Let us jump into the Excel file for some modification before we can bring that plot file in ANSYS.

If you open the Excel file now, here is what you will see –



Here is a step-by-step guide to prepare the plot file for ANSYS –

- Delete the following rows



- Delete everything under this row

197	993.844	-0.891
198	996.057	-0.572
199	997.781	-0.322
200	999.013	-0.143
201	999.753	-0.036
202	1000	0
203		
204	Camber line	
205	X(mm)	Y(mm)
206	0	0
207	0.247	0
208	0.987	0
209	2.219	0
210	3.943	0
211	6.156	0
212	8.856	0
213	12.042	0
214	15.708	0

- Create three new columns and name all the columns accordingly. Then, fill up the values as shown in the picture.

	A	B	C	D	E	F
1	#group	#point	#x	#y	#z	
2	1	1	1000	0	0	
3	1	2	999.753	0.036	0	
4	1	3	999.013	0.143	0	
5	1	4	997.781	0.322	0	
6	1	5	996.057	0.572	0	
7	1	6	993.844	0.891	0	
8	1	7	991.144	1.28	0	
9	1	8	987.958	1.737	0	
10	1	9	984.292	2.26	0	
11	1	10	980.147	2.849	0	
12	1	11	975.528	3.501	0	
13	1	12	970.44	4.216	0	
14	1	13	964.888	4.99	0	
15	1	14	958.877	5.822	0	
16	1	15	952.414	6.71	0	
17	1	16	945.503	7.651	0	
18	1	17	938.153	8.643	0	
19	1	18	930.371	9.684	0	

- Finally, change the values in the last rows accordingly, this will 'close' the curve.

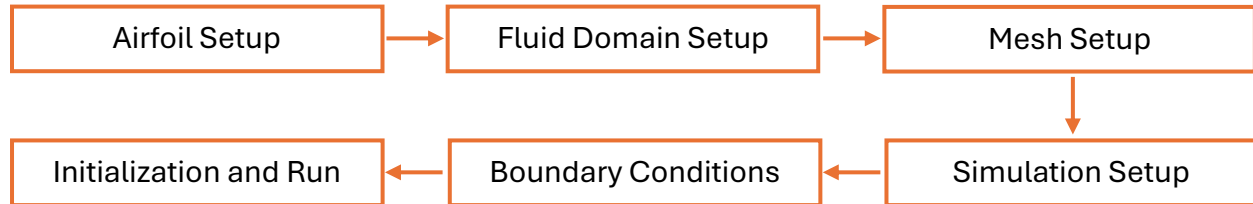
194	1	193	984.292	-2.26	0	
195	1	194	987.958	-1.737	0	
196	1	195	991.144	-1.28	0	
197	1	196	993.844	-0.891	0	
198	1	197	996.057	-0.572	0	
199	1	198	997.781	-0.322	0	
200	1	199	999.013	-0.143	0	
201	1	200	999.753	-0.036	0	
202	1	0				
203						
204						
205						
206						
207						

And we are done. 'Save As' text file (tab delimited) and the plot is ready to be imported in ANSYS. Pats on the back. Go get some snacks, we've done something here!

ANSYS (Computational Fluid Dynamics)

Now that our plot file is ready to be imported, it's time to get into ANSYS (How was the snack? Tasty?).

Before we dive in, here is a small overview of ANSYS workflow –



Airfoil Setup

This is one of the most important (if not the most important) part of the whole workflow. This is where you mathematically define the model and fluid domain, with 'backbones' for mesh settings and boundary conditions. Pay attention to every single detail here. I know you are not hungry since you just had a snack so don't give me any excuses.

Good?

Good.

The way we tackle this section will be short description of what you need to do, associated with screenshots from the ANSYS window. DO NOT skim any of it. One tiny mistake here and you'd pull your hair off later down the workflow. Trust me, speaking from experience.

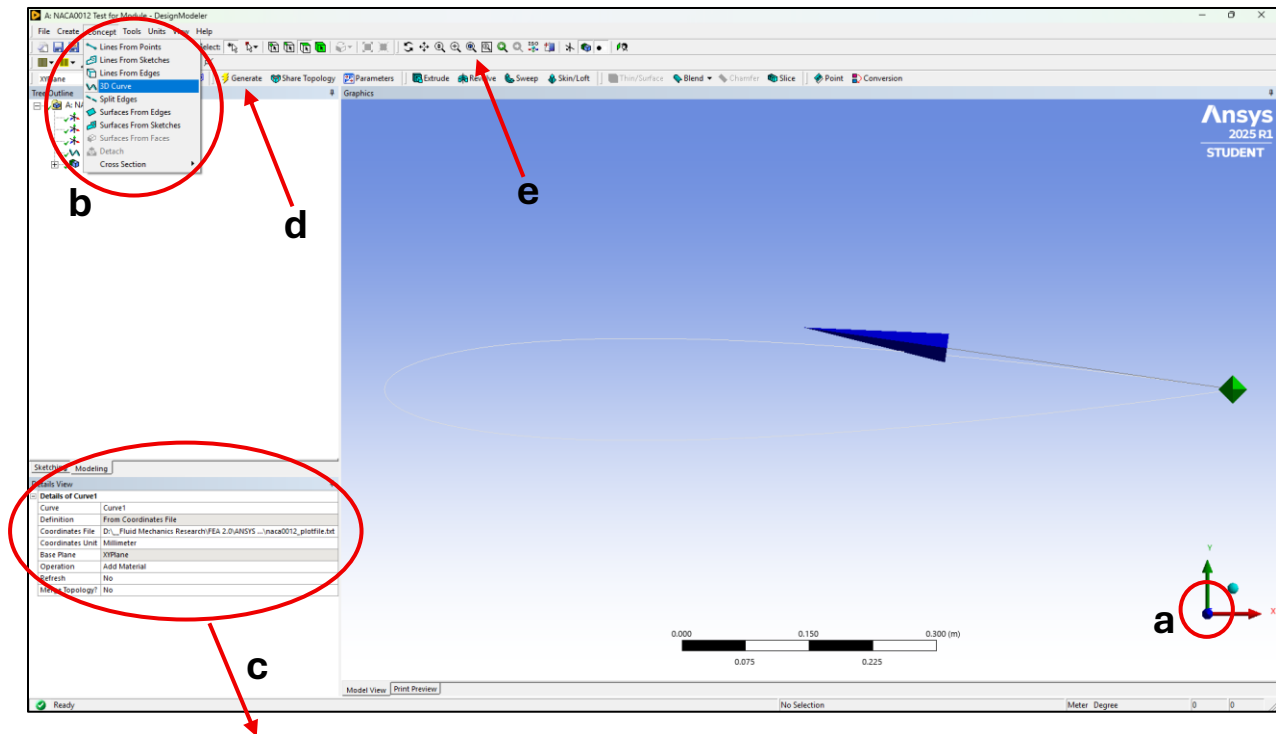
Open ANSYS Workbench and drag the Fluid Flow (Fluent) component into the bench to create a standalone system.

Very Important – This is a good point in the workflow to save the project (still early, I know). There is a proper way to save ANSYS projects – saving it in a dedicated folder. ANSYS creates a bunch of system and work specific files and settings and they will be lost in the abyss (not really, but you get it) if you don't save all of it under one particular folder. Try not to save anything (apart from the excel and text file for the plot I suppose) in that folder. I know, "Important Homework" folder seems tempting to be saved here, don't!

Before you open 'Design Modeler' for geometry setup, right click on geometry in the workbench and go to properties. There, change the 'Analysis Type' to 2D.

Once everything is set and done, right click on geometry and click 'New Design Modeler Geometry'.

Let the 'fun' begin.



Sketching Modeling	
Details View	
Details of Curve1	
Curve	Curve1
Definition	From Coordinates File
Coordinates File	D:_Fluid Mechanics Research\FEA 2.0\ANSYS ...\naca0012_plotfile.txt
Coordinates Unit	Millimeter
Base Plane	XYPlane
Operation	Add Material
Refresh	No
Merge Topology?	No

In the 'Design Modeler' window, do the following –

- Click on the 'Z-axis' to show XY plane (this is where we plot the airfoil curve)
- Click Concept – 3D Curve and select the text file we got from Excel
- Make sure all the options are just as shown (for example, we fixated on SI unit mm in our example, so coordinates unit is selected as such)
- Click generate
- Click fit to window and you can see that the curve has been generated

Now, go to Concept – Surfaces from Edges – Select the curve – Click Apply – Click Generate.

Your Airfoil is ready (to be extracted from the fluid domain) now. Hooray.

Fluid Domain Setup

Now, we make the fluid domain. Our 'spacetime' continuum for the simulation, if you may.

To construct the fluid domain, here are the basic workflow –

- Sketch the fluid domain shape (a domain with a 'C-Shaped' inlet)
- Convert the sketch into surfaces (including the airfoil)
- Boolean subtract (don't worry, we'll talk about it) airfoil from the domain
- Minor adjustments (minor in name, major in implication)
- Named selection (basically defining boundary conditions, makes things easier during the simulation setup)

Let's start.

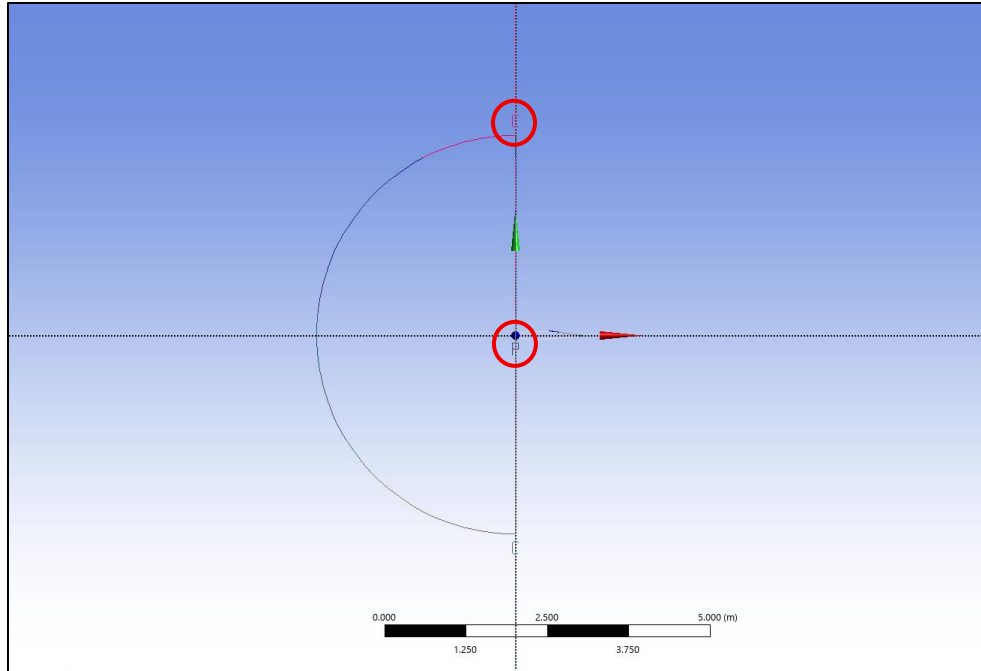
The C-Shaped Domain: Such domains offer a better convergence (that is what you need in the end) for results during the simulation. Start by drawing a C (arc by center) in front of the airfoil we generated, with the radius to be 7.5 times higher than the chord length. Then draw two horizontal lines (15 times the chord length each) and connect them vertically. Follow the steps shown below –

Unimportant for this study but really important for future reference: The dimensions are not arbitrary. When we design a fluid domain, we have to make sure the boundary walls do not compromise the analysis by being 'too close' to the airfoil, or do not blow our mortal machines running on 6 years old CPU by being 'too far'. From standard practices, backed by previous studies conducted by the likes of NASA, 5-10 chord length is advised around the airfoil and 10-ish chord length is advised for the wake region, for cases like ours (subsonic flow, simple airfoil). So, 7.5x and 15x are middle grounds. This value is dependent on a few things (for future studies) that includes (but not limited to) – shape and geometric complexity of the airfoil, flow velocity, very high AoA etc. Always check with a supervisor or professor before stating such dimensions. This is MEGA important. To sum up – 7.5x and 15x chord length are like 2% milk of CFD domain, good enough for most cases. But do check if you need oat milk instead in your future studies.

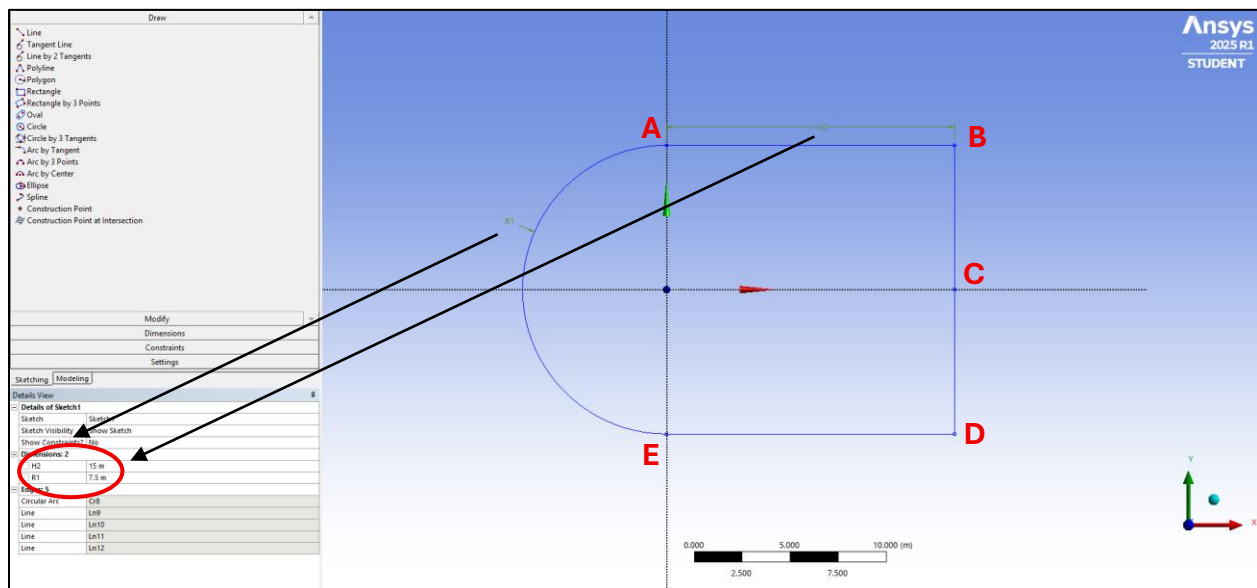
A few things to check while drawing the domain –

- The letter 'C' appears when your 'pencil' is coincidental with the axes.
- The letter 'P' appears when your 'pencil' is coincidental with the origin.

While looking at the XY-plane, click Sketching – Draw to bring out the options to select 'Arc by Center'. Once you center your arc in the origin (the pencil showing 'P'), draw it so that the letter 'C' appears on both top and bottom lines of the Y-axis. Click Dimensions – Radius to define the length of the radius.



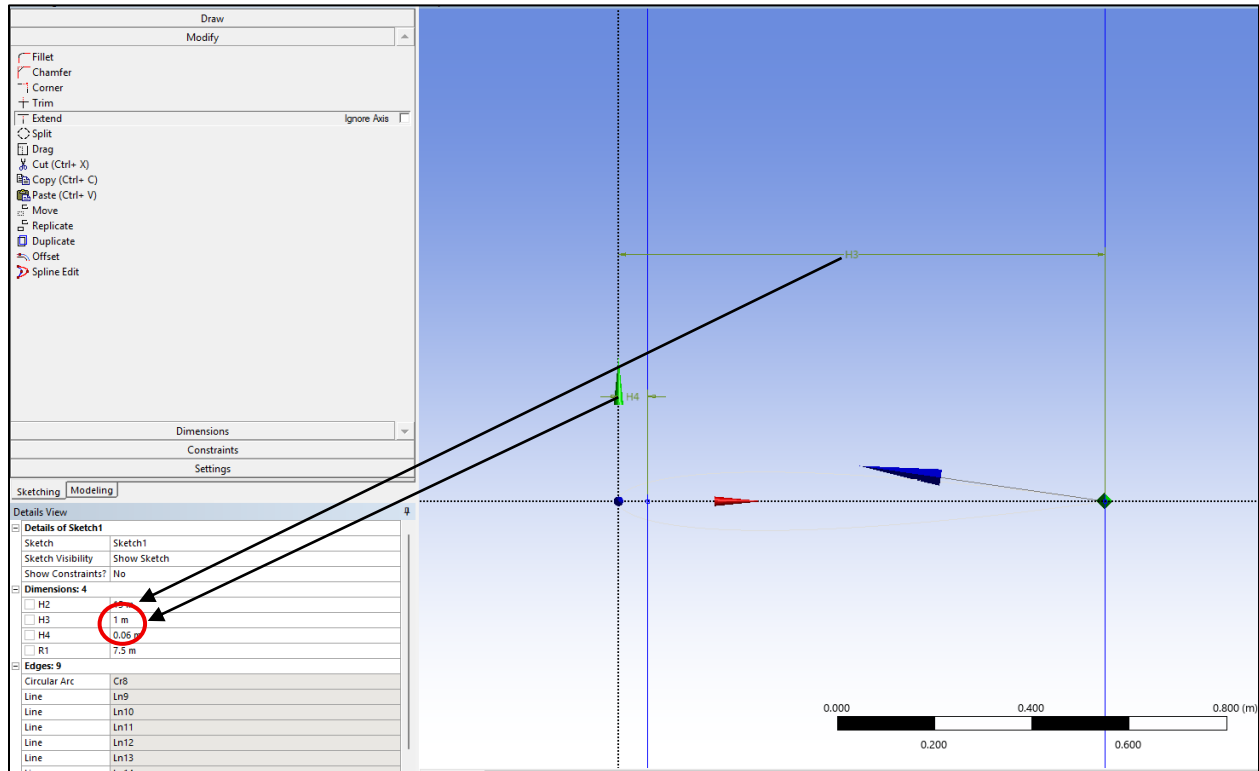
Once done, draw lines AB, BC, CD and ED in that order, to make sure they are all aligned with our dimensions. This way, you only need to define dimensions of AB (15m in our case) and follow visual guides (letter 'H' appearing for a horizontal line, letter 'V' appearing for a vertical line) to finish the construction.



There are a couple of lines we need to draw to complete the fluid domain construction. This two lines is paramount for proper meshing and result discussions. These two lines will 'define' dedicated domains for the leading and trailing edge, by offering a far better meshing structure – giving us a better convergence.

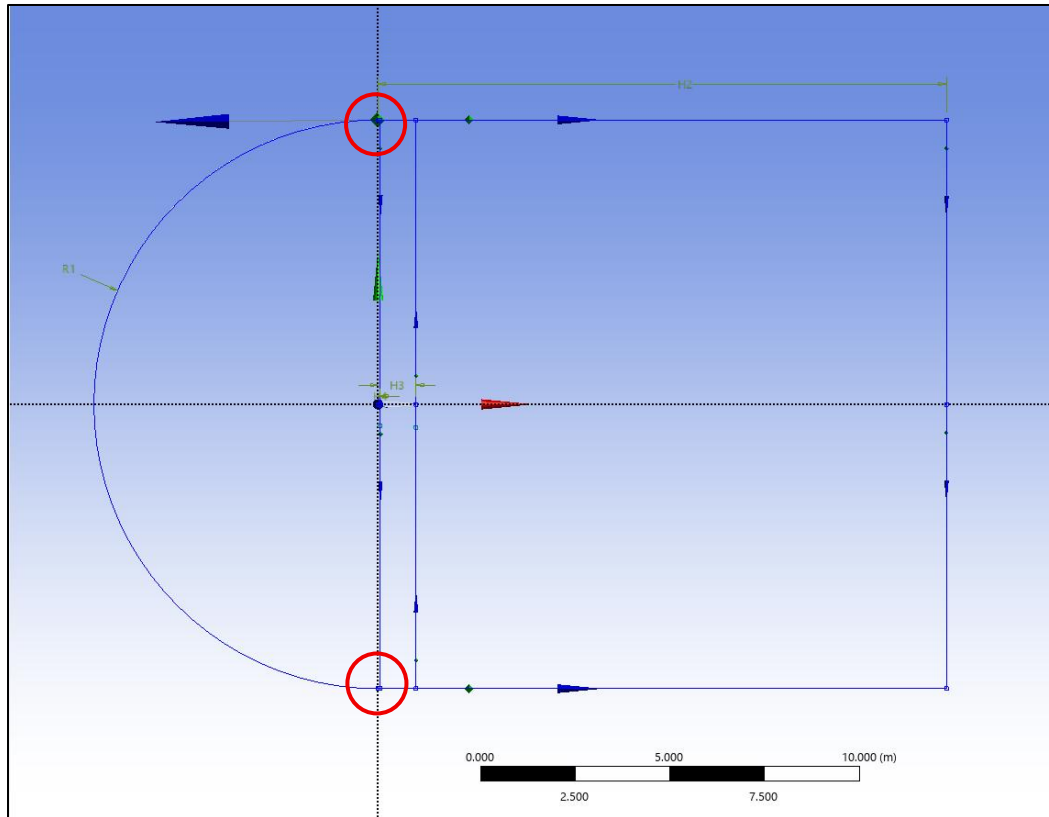
Start with the line at the trailing edge by drawing two vertical lines from the X-axis. Give a horizontal dimension equal to the chord length (1m in our example, so that it intersects with the trailing edge). Then, simply 'extend' the lines so that they meet the fluid domain. You can find 'extend' in Sketching – Modify.

Do the same for leading edge, but this time, the horizontal dimension should be about 40-80 mm. This is a standard range of values and different airfoil shapes may require different values. But the implication is minimum – feel free to do some trial and error if you want.



Once you sketch all the elements, go to Concept – Lines from Sketches – Select the Sketch from the model tree (in the Base Objects) – Hit Apply – Generate.

If all goes well so far, you will have something like this –

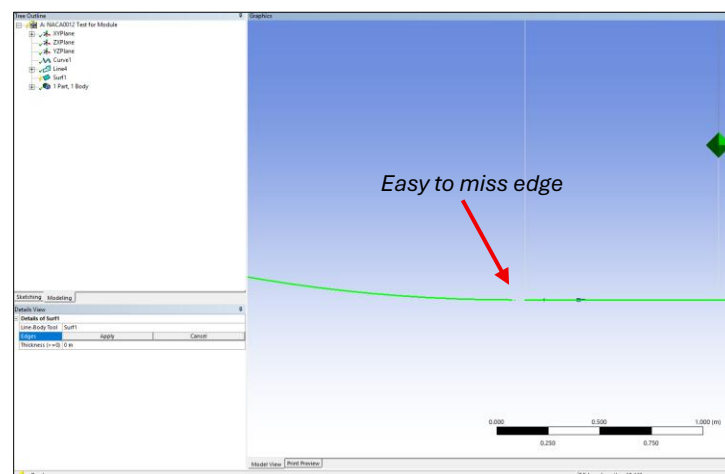


Congratulations. Now go get a cookie and come back when you are done.

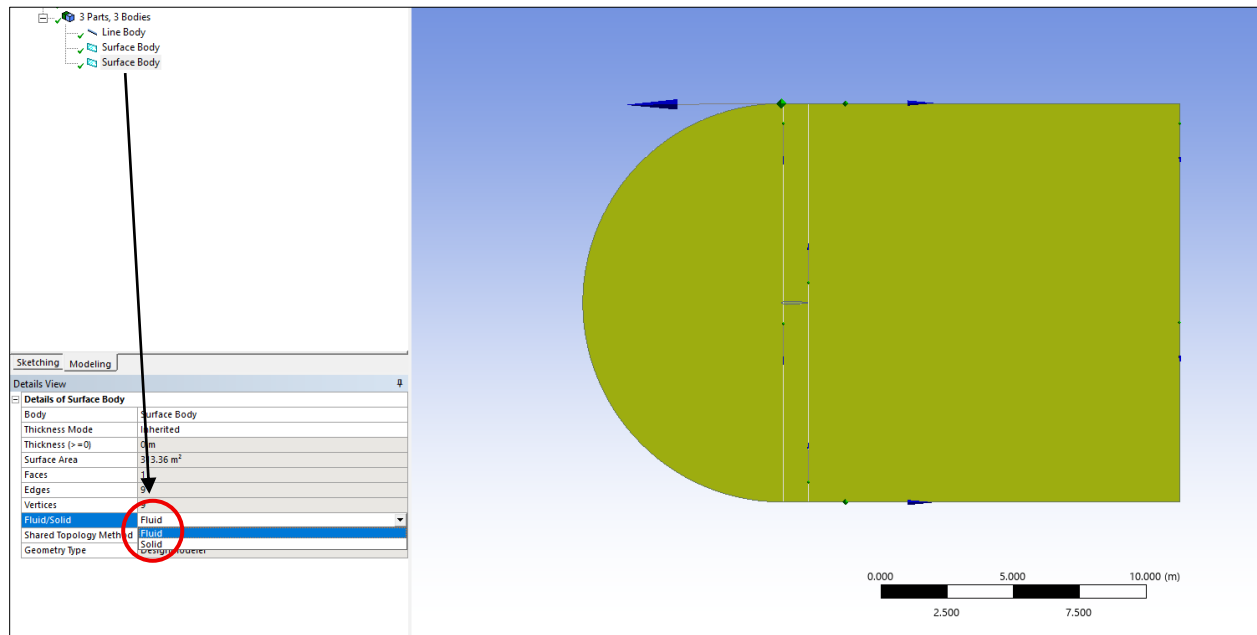
Surface Generation and Modification: Now that we have all the elements for surfaces being generated (the geometric property that will be the center point of our CFD analysis), let's turn them into surfaces and modify according to our needs.

Go to Concept – Surfaces from Edges – Select all the outer edges* – Click Apply – Generate.

*Please note, there are some small edges on the marked areas. Zoom in and select them as well, since they are very easy to miss. Hold 'Ctrl' to select all the edges.



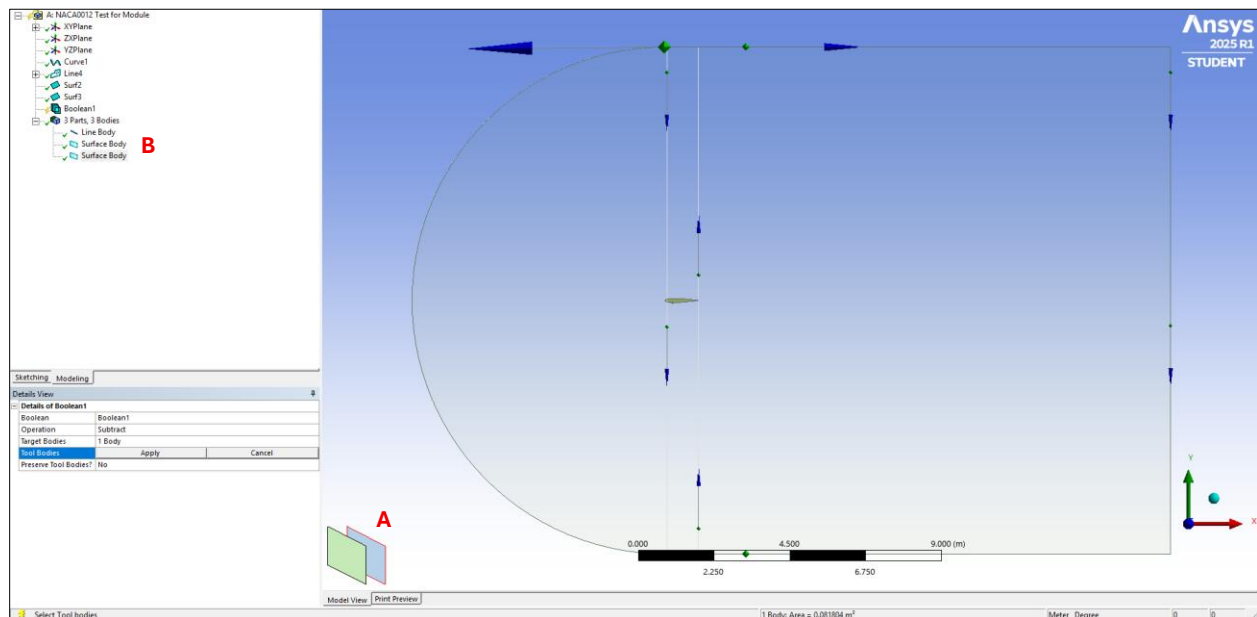
Once generated, click on the newly generated surface, Solid – Fluid (hit generate again).

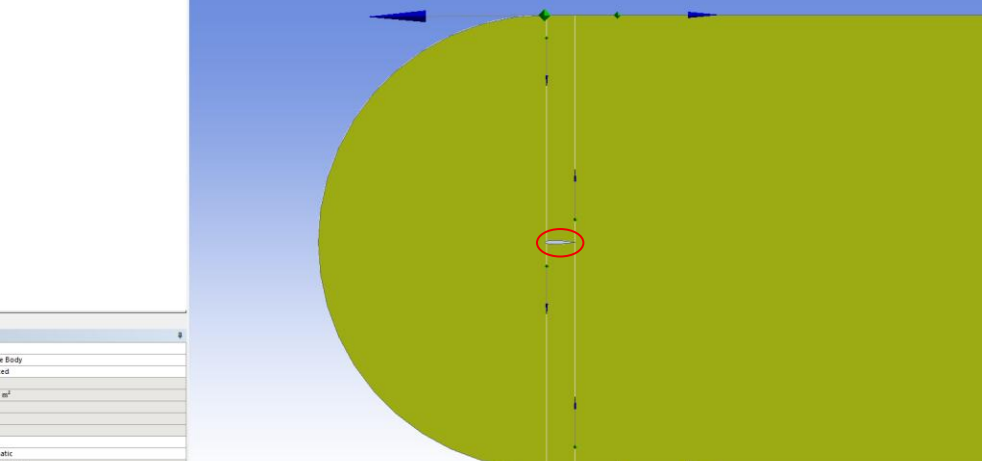


Now, we will make an airfoil shaped 'hole' in the fluid domain. To do that, we will use the Boolean Subtract tool.

Go to Create – Boolean – Select Subtract – Select Fluid Surface as the Target Body – Select the Airfoil Surface as the Tool Body* – Hit Apply – Generate

*To select the airfoil surface as the tool body, use the plane selector (marked A). You can also select the surfaces using the Tree (marked B).





ANSYS 2025 R1 STUDENT

0.000 2.500 5.000 7.500 10.000 (m)

Sketching Modeling

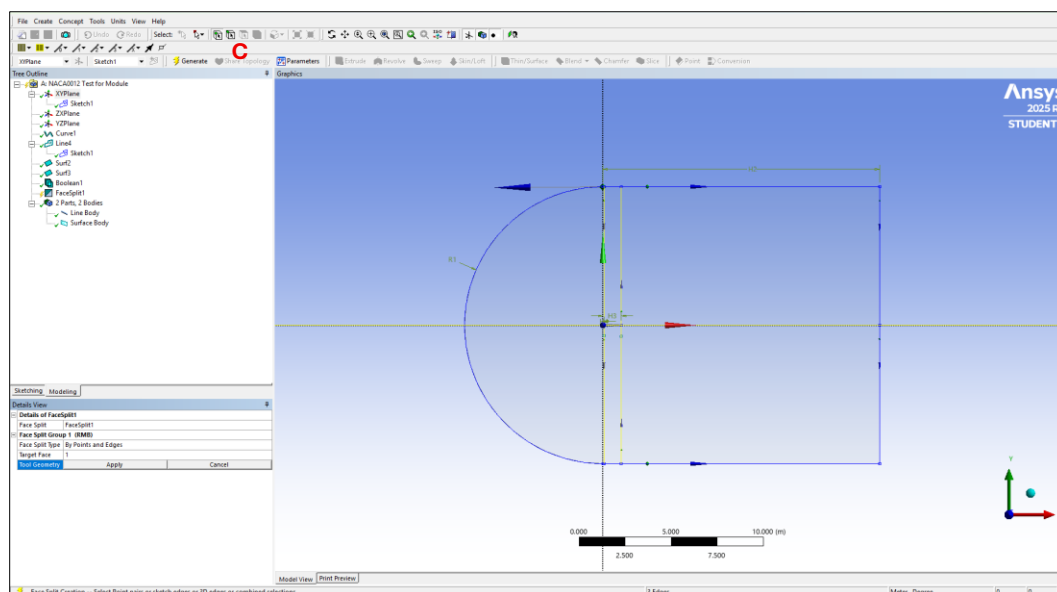
Details View

Details of Surface Body

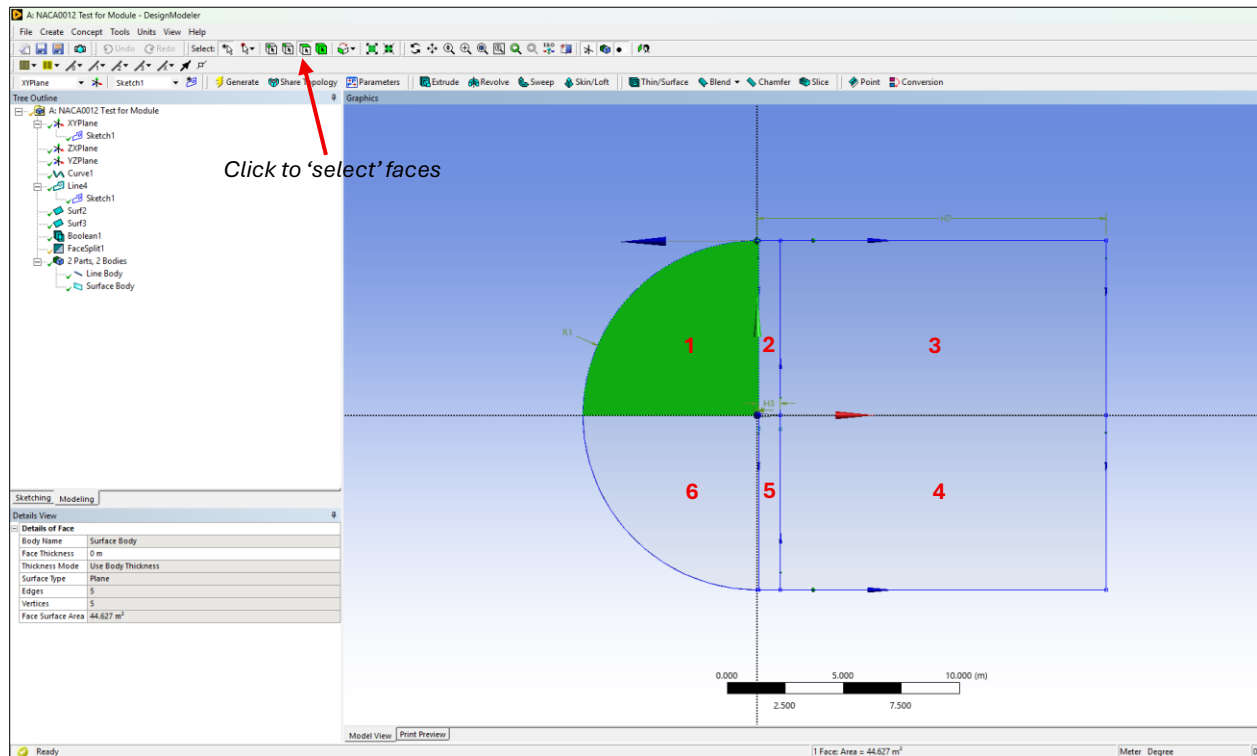
Body	Surface Body
Thickness Mode	Inherited
Thickness (t=)	0 m
Surface area	313.28 m ²
Faces	1
Edges	10
Vertices	9
Fluid/Solid	Fluid
Shared Topology Method	Automatic
Geometry type	DesignModeler

Go to Tools – Face Split – Select the Domain as Target Face – Select all the lines shown in the screenshot as Tool Geometry (holding Ctrl) and the X-Axis* – Click Apply – Hit Generate.

Alternatively, you can click the button shown in the screenshot (Marked C). If the X-axis isn't shown during the selection, click the XY Plane in the tree before start selecting.

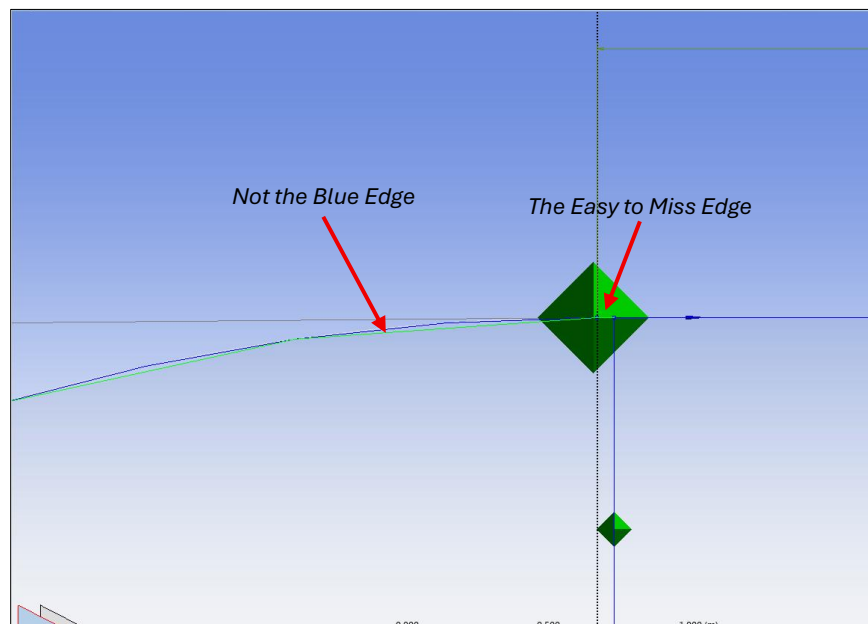


If done correctly, you will have a total of 6 faces.

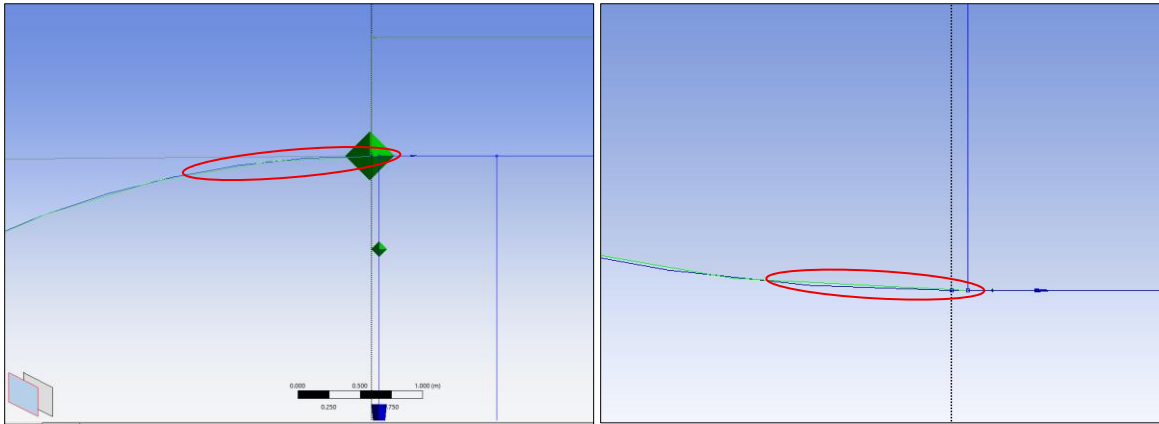


One final thing, we will merge a few edges. This is a very important part as it repeatedly gave me errors during my first try. So, pay attention to the edges being selected during the merge operation.

Select the edges shown below – Go to Tools – Merge – Hit Apply – Generate.



Do the same for the bottom portion of the fluid domain. Once both of these are merged, you will see that selecting edges (using the Selection Filter button for Edges) will show one edge being selected (each for top and bottom part of the fluid domain).



And we are done setting up the fluid domain. Close the Design Modeler window, go back to the workbench and click that floppy disk (save) button. Good practice, as always.

Now, it's time for some coffee since the fun is just starting.

(Oh my God how long is this? Well, the hardest part is done. Or is it?)

Mesh Setup

Okay jokes aside, I promise you things get a lot easier from now on. The fluid domain setup part was probably the most 'hmm what?' section for me when I first dabbled my feet in the Computational Fluid Dynamics domain. I pulled so many hairs doing these over and over again (yeah, I know it looks easy now, quite it!) just to see the mesh failing and giving me so many undesirable patterns down the line.

Good news is, if you have done all the things correctly so far, it's smooth sailing from here.

Why Mesh: Meshing is nothing but discretizing the fluid domain in smaller sections, where the physics run individually – meaning in each of those small section, general fluid mechanical differential equations (Navier-Stokes for example) transform into algebraic equation, so that your sad old PC can calculate forces without frying your outlet.

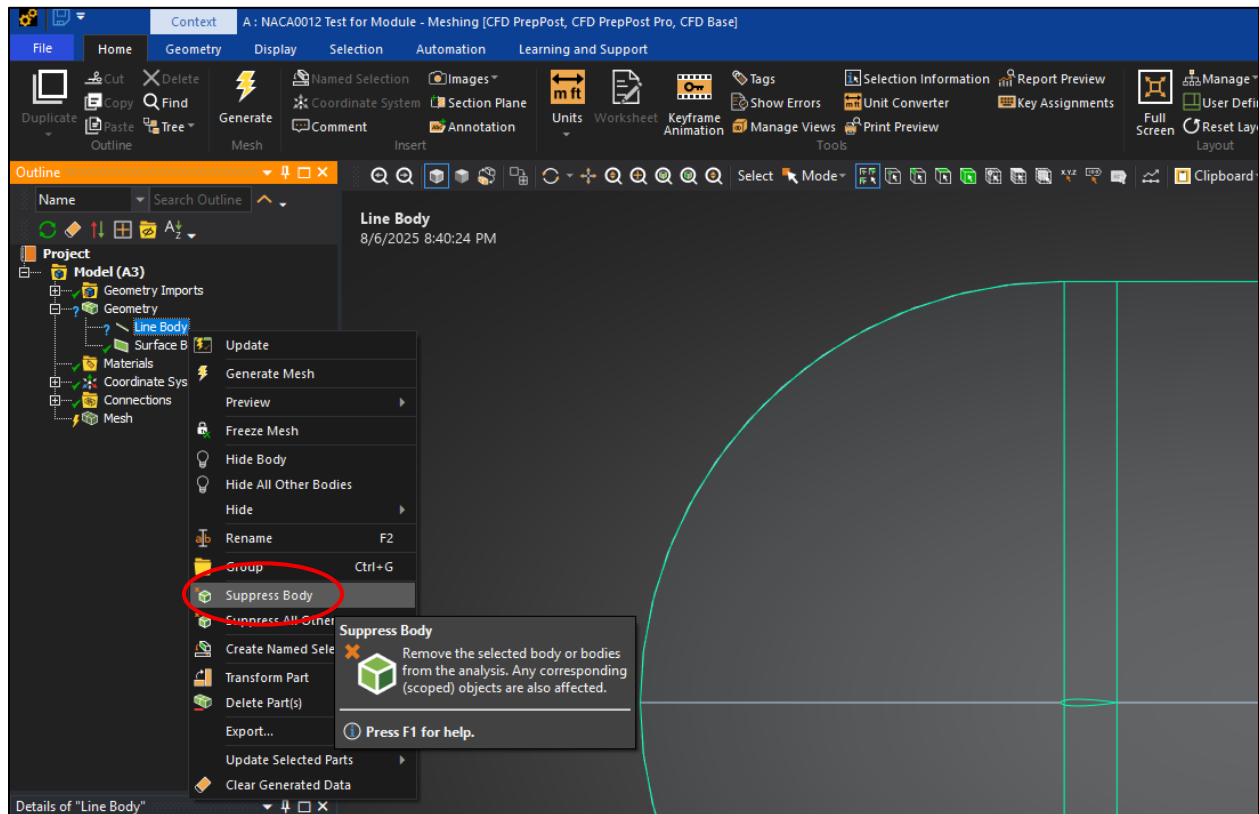
Mesh helps us refine the fluid domain according to our needs. For example, right around the airfoil where most of the work (pressure, force, moment) is done, alongside the existence of the boundary layer (a layer that 'soaks' the solid, where fluid particles have an astounding velocity of exactly 'not'), the domain must be discretized finely. In other places (far from the airfoil, along the fluid domain wall), the 'cells' do not have to be miniscule.

All the shenanigans we did after the basic sketches of the fluid domain, we did them to achieve just that. Pat yourself on the back.

Let's get to work.

From the workbench, right click Mesh and click edit.

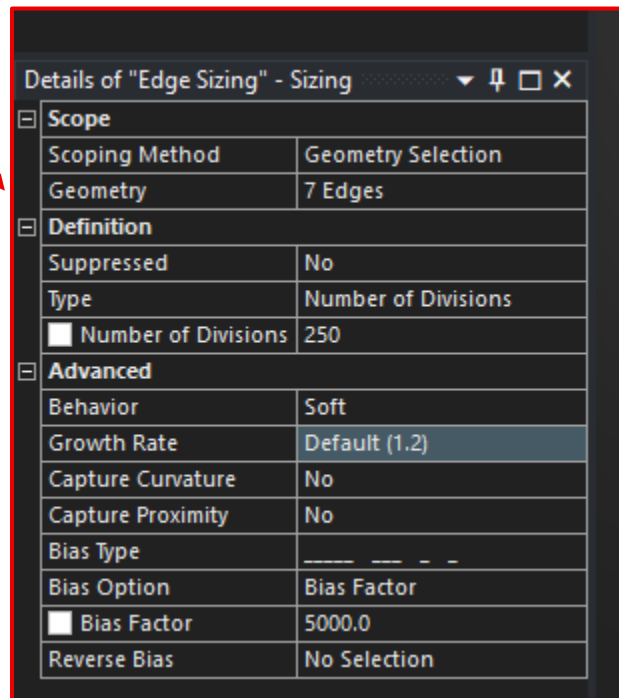
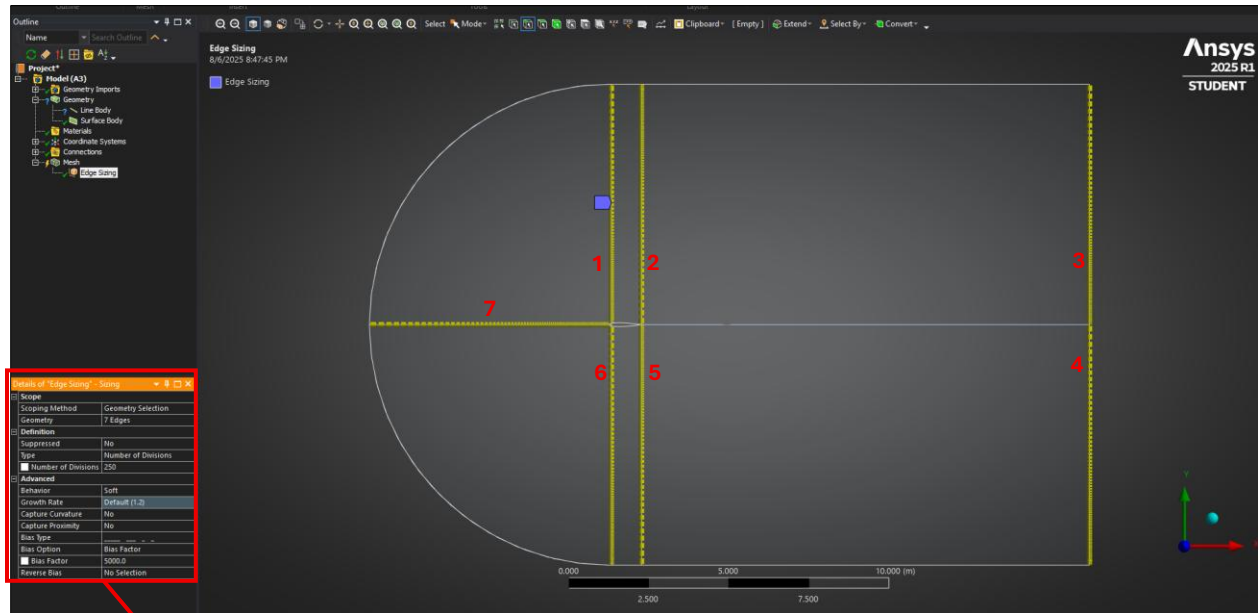
Once in the ANSYS Mechanical, select the 'line body' from the tree and suppress it. We don't need that anymore. Adios, Mr. Line.



Sizing: This is where we fine tune the fluid domain so that the resultant mesh is exactly what we want. Sizing will help us define 'density' of the cells, as in how small or big they should be in a given area.

Right click on mesh, select sizing from the insert menu, make sure selection filter is in edge (Ctrl+E) and select the edges shown in the screenshot below (marked with dotted yellow lines). Carefully make sure all the settings are exactly what it says. And then, hit apply.

DO NOT hit generate yet, as I am sure you want to click that shiny button. We will look into it later.

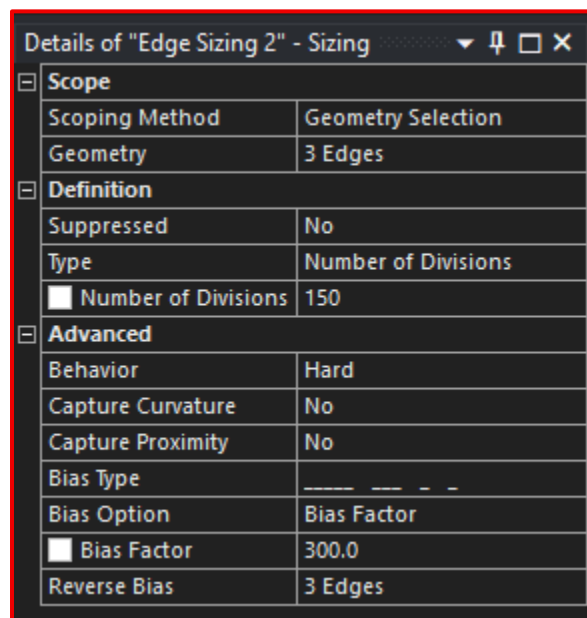
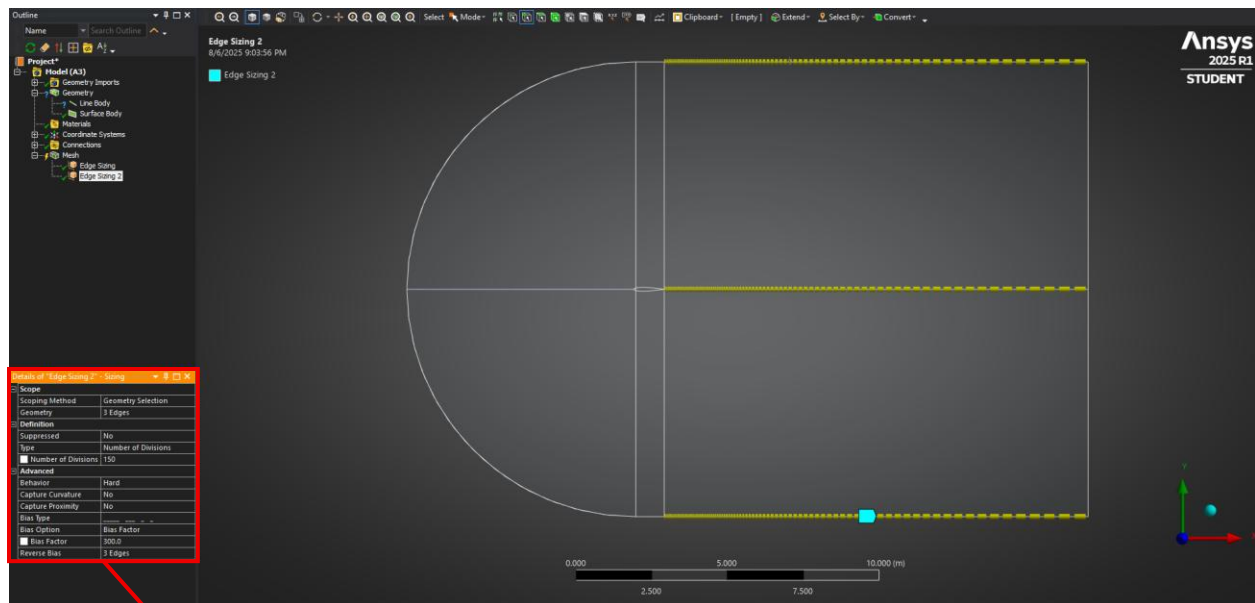


(Note: If your version of ANSYS doesn't show these options as default in the sizing selection, just turn the 'Capture Curvature' to 'No'. This was an option I struggled for so long since I wasn't paying attention to all the details buried under that clunky UI.)

The Bias option in sizing is nothing but a 'converging density pattern' that will yield a denser mesh as the domain approaches the airfoil (or the chord line, to be more specific).

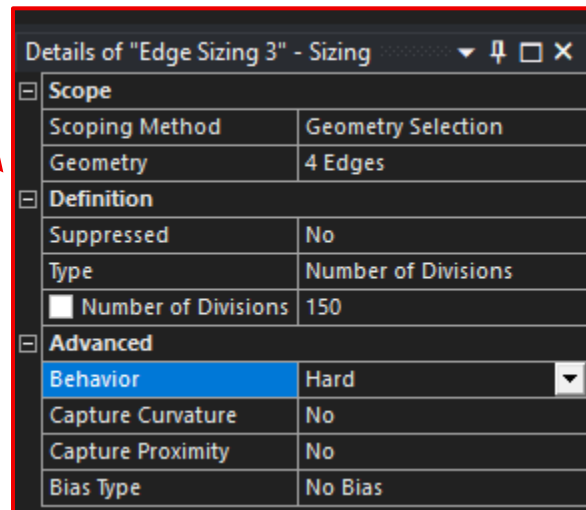
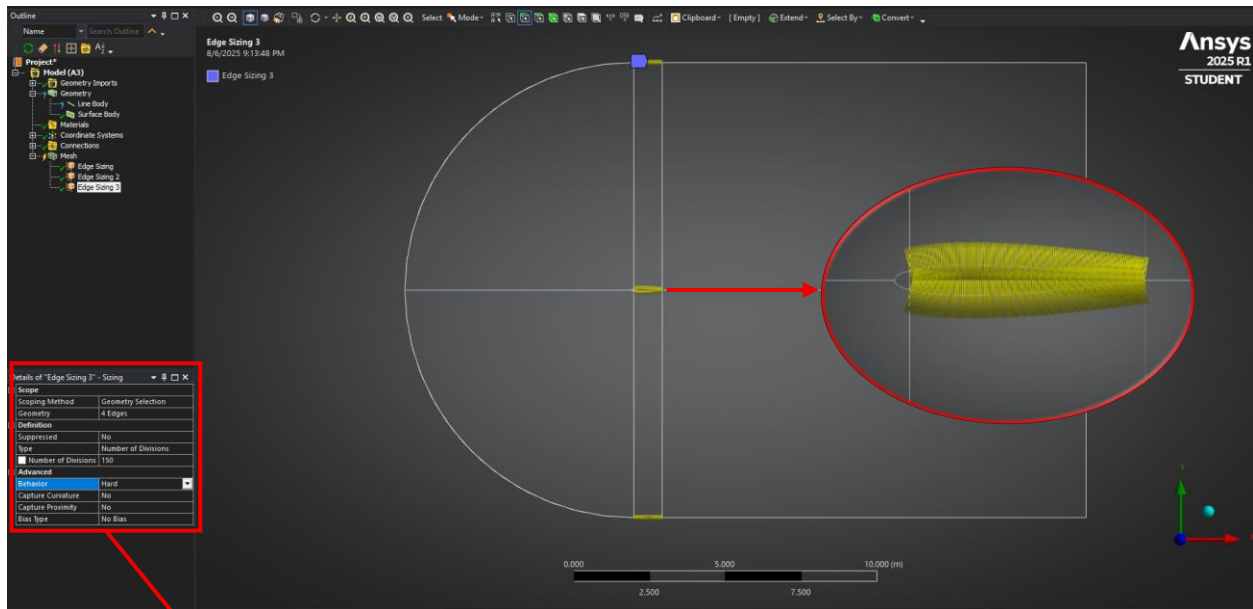
Now, you will probably see some of the lines (in our case, line no 2, 4 and 6) is showing a flipped density pattern. Just select these three and apply the Reverse Bias option.

Now, time to add the next sizing. Screenshots below –

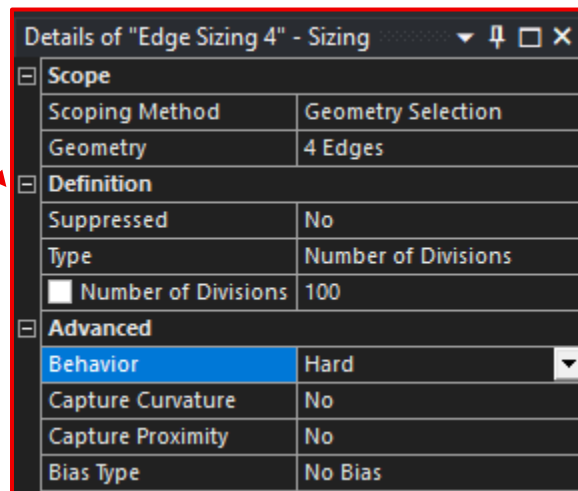
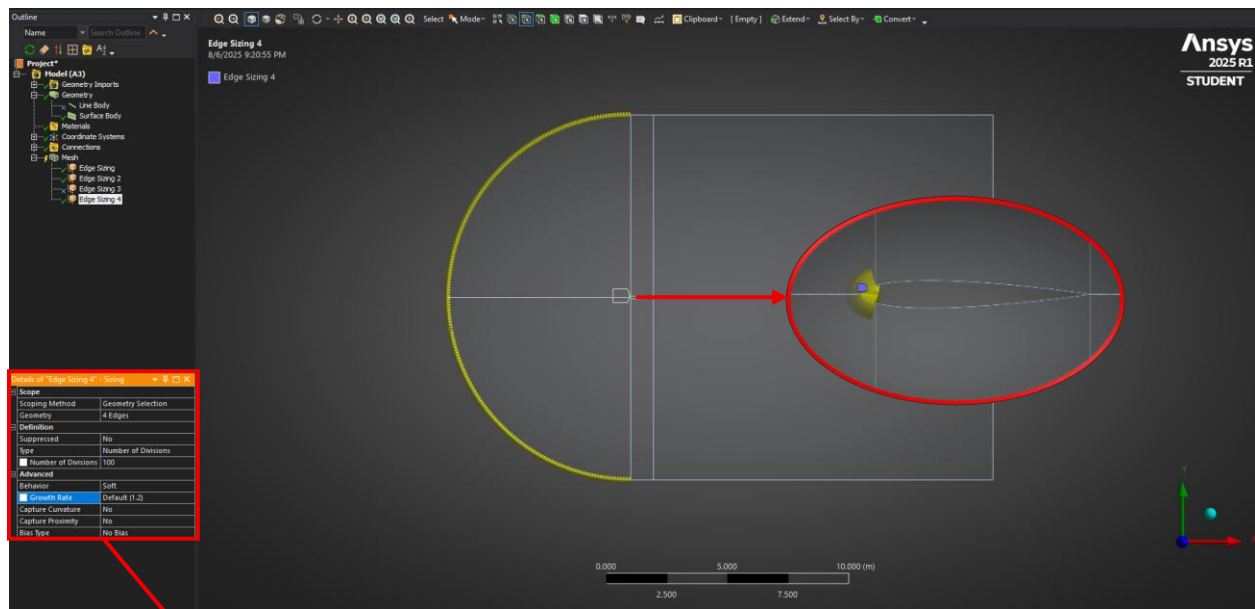


Point to note, the behavior for this sizing is set to hard. This is useful for simple shapes and objects. Previously, it was set to soft to account for the minute curvatures of the airfoil. It allows the solver to go easy (soft) or hard on the number of divisions of elements. It is good practice to allow some 'room' by selecting soft around the airfoil for additional elements if need be. This will allow us to deal with errors. On the other hand, we are setting hard for computational efficiency and a structured mesh elsewhere.

Onto the next sizing shenanigans –



And lastly, this one –



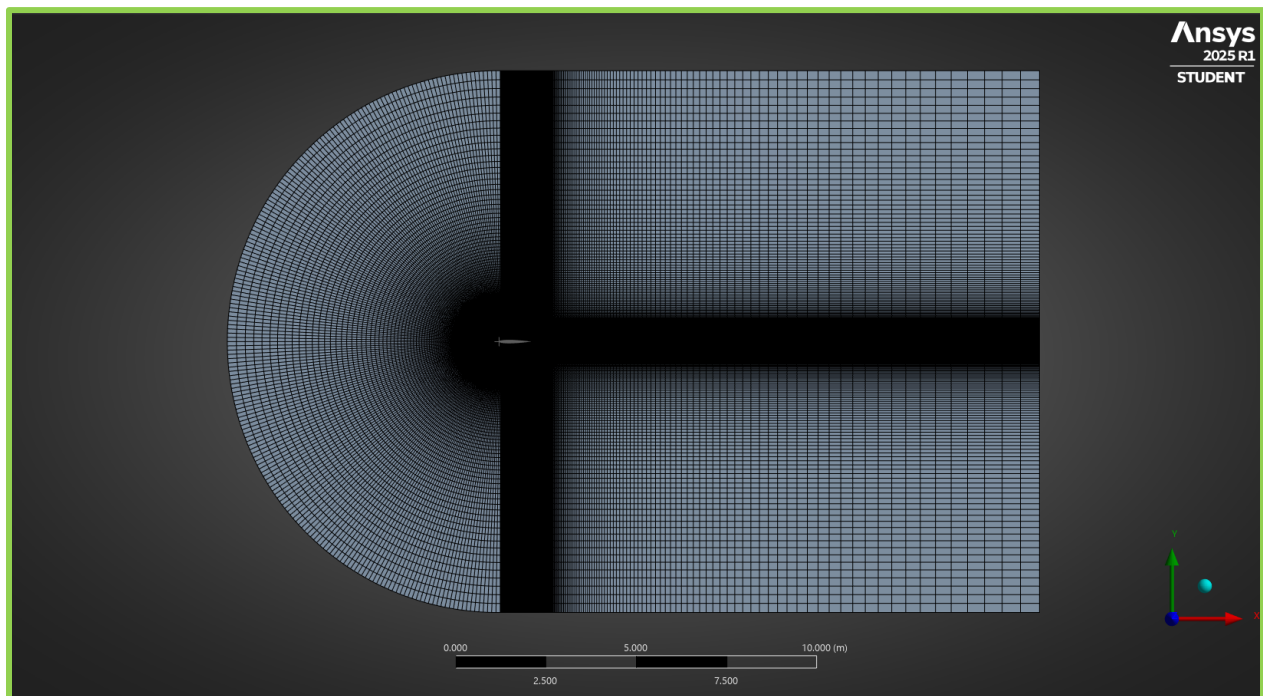
Now that all the edge sizing is done, we will do one final sizing. This is to define the shape of the mesh cells as quadrilateral instead of triangular.

Right click on Mesh – Insert – Face Meshing – Ctrl+A (to select all the faces) – Click Apply. Once done, do a quick check that the options look like this –

Details of "Face Meshing" - Mapped Face

Scope	
Scoping Method	Geometry Selection
Geometry	6 Faces
Definition	
Suppressed	No
Mapped Mesh	Yes
Method	Quadrilaterals
Constrain Boundary	No
Advanced	
Specified Sides	No Selection
Specified Corners	No Selection
Specified Ends	No Selection
MultiZone Semi-Structured	No

Hit Generate. And...



Pretty, isn't she?

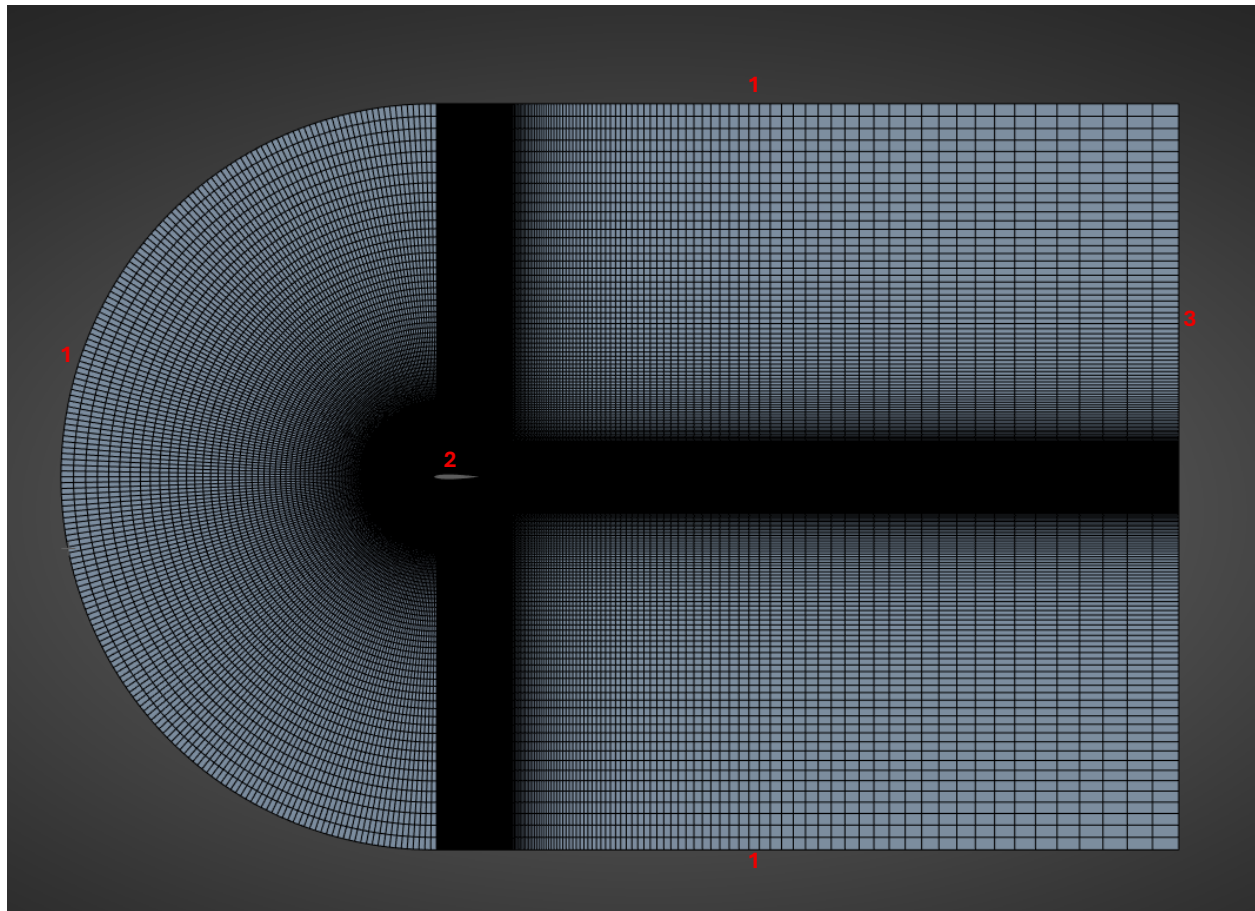
Seriously now, congratulations are in order. Look at that symmetry! If you managed to get this (which I'm sure you did, given you followed the instructions, in order, and word by word), you have done 90% of the simulation.

Take that whoever told us we can't do CFD. Go cry in a corner while we plot vectors from now on. Weep in our glory while secretly wishing to join us! (okay that was enough)

But before we jump into the simulation, here is one final thing that requires attention. Named Selection. Basically, we will be assigning the inlet, outlet, airfoil and wall to our model, i.e. setting up the boundary conditions (sort of) right in the mesh settings. You can skip this if you want, but this small thing saves so much time down the line when we setup the simulation.

Click the edges (with Edge Selector filter turned on) shown below – right click – create named selection. Alternatively, you can select the edges and press ‘N’ in the keyboard.

Consult the following screenshot –

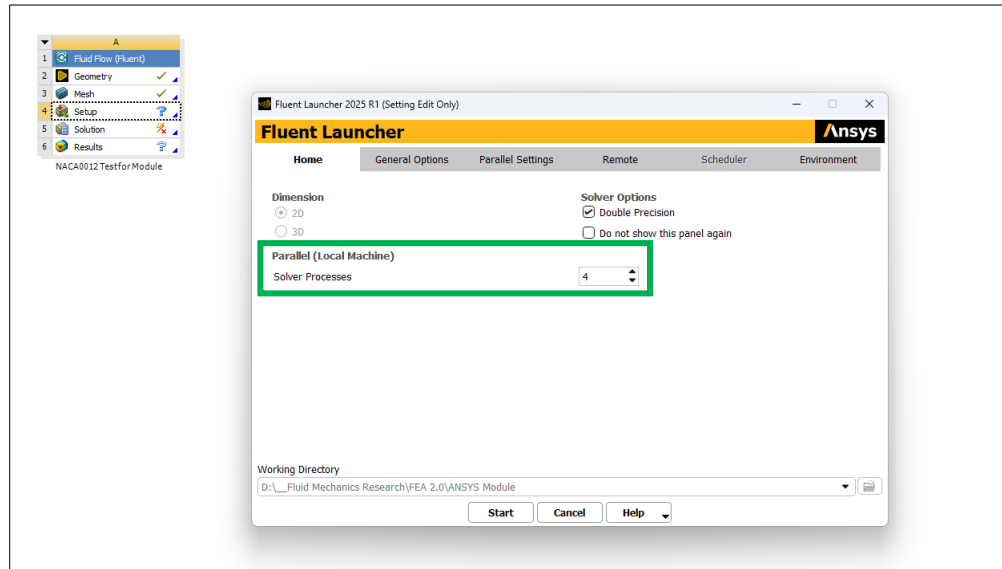


1. Inlet (make sure to select all the outer edges except the ones at the back)
2. Airfoil (there are a total of 4 edges to select)
3. Outlet (2 edges, in the back)

Before we dive into the simulation setup, there is a very important concept to check for at this stage of our CFD journey – Y^+ value. It's a dimensionless number that has significant impact on the result. But for now, we can skip it. Considering all the steps we took, the Y^+ value should be okay for our case. I will discuss about it in a separate module in the future.

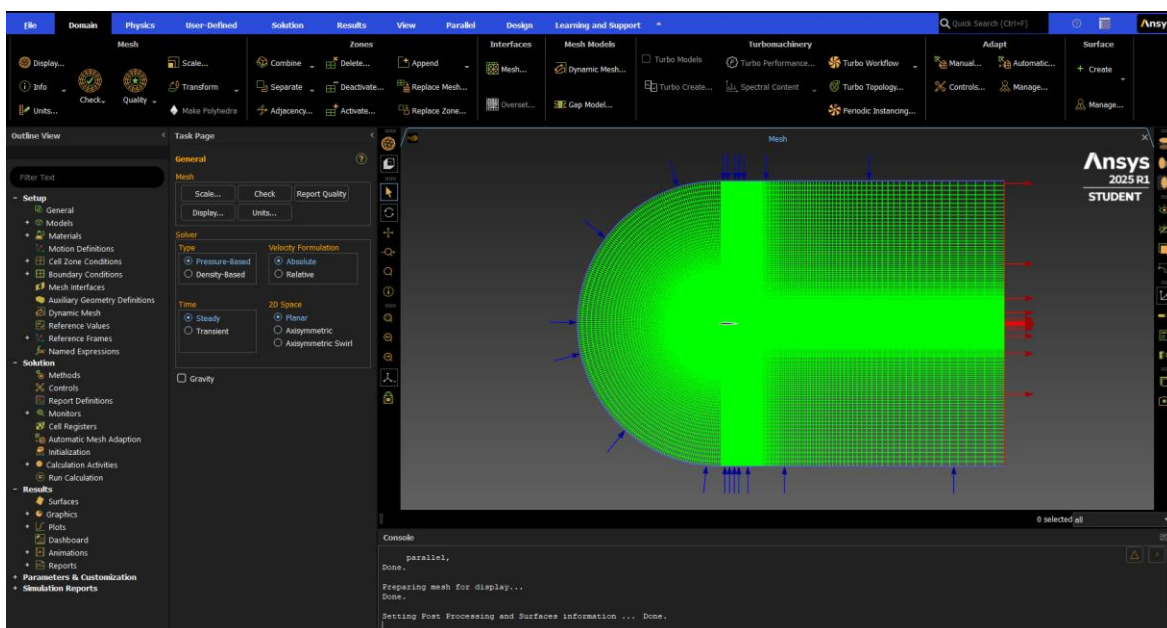
Simulation and Boundary Condition Setup

Now, let's head back to workbench and save the project (for good measure). Right click Setup and select edit, consult the following screenshot and hit start.



Please note, more solver processes do not mean better. In fact, for low fidelity models (like this 2D model), more processes might even be worse. The explanation is, ANSYS will divide the computing load to individual cores of your processor and then work parallelly under the hood. Imagine you have 100,000 mesh cells. If you select 4 processes (cores), each will deal with 25,000 cells when in reality, each one can comfortably work with 60-70 thousand cells. So, you will be simply be idling the working cores while engaging more. The values are arbitrary of course, but the logic still stands.

Once ANSYS Fluent loads, it should look like this –

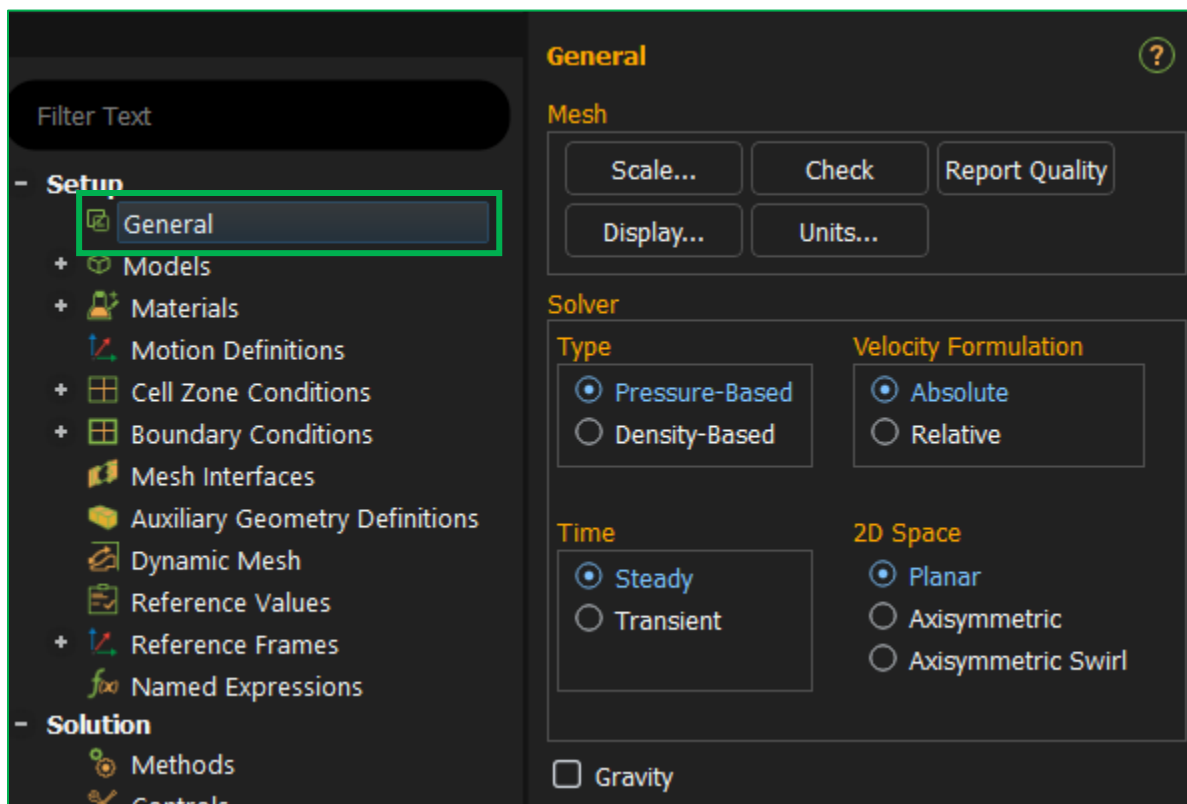


Ignoring the 'dark mode', you can see the solver already assigning vector arrows to represent inlet and outlet (thanks to our little trick with named selection earlier). This makes things much easier.

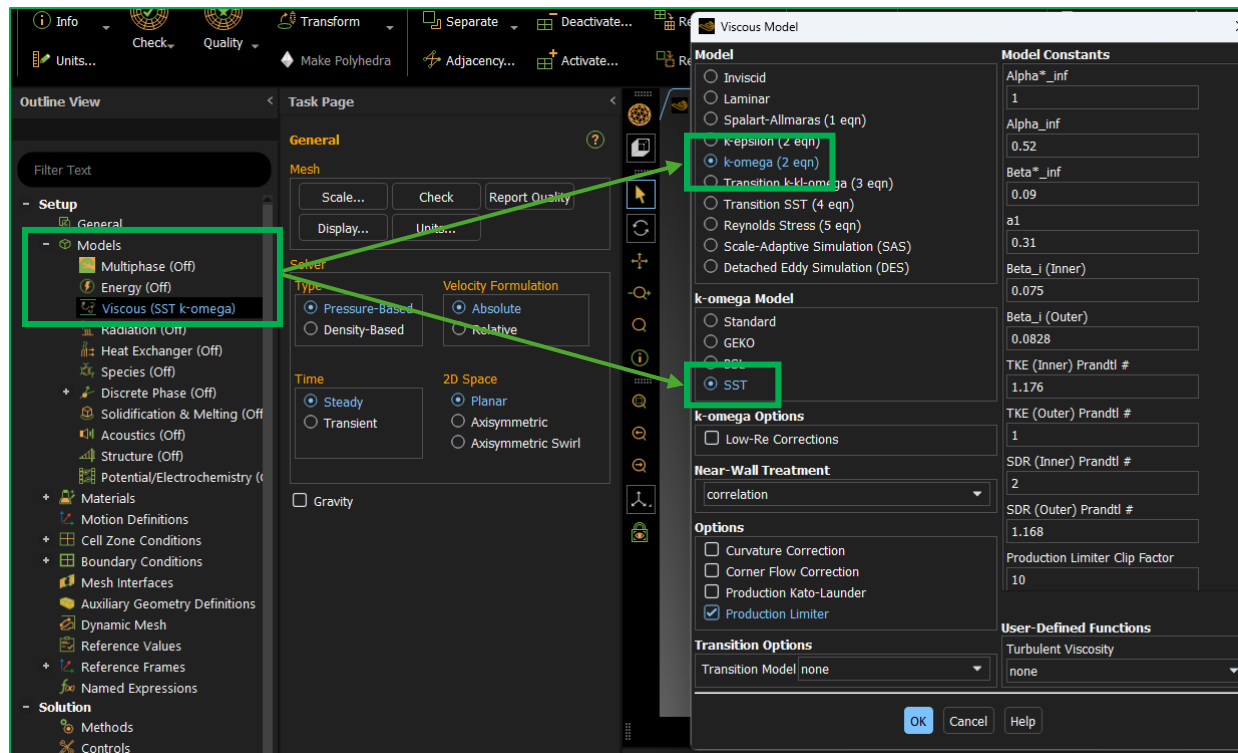
I promise we are very close to run the simulation. Thank you for your patience so far.

Pro Tip – If for some reason, you need to update the mesh (or say, forgot to use named selection) because you see something completely different in the solver window, close the window, go back to the workbench, right click on setup and click 'clear generated data'. Then you can go back to updating the mesh and come back to it. Unsurprisingly, that is something that happened during my trials once!

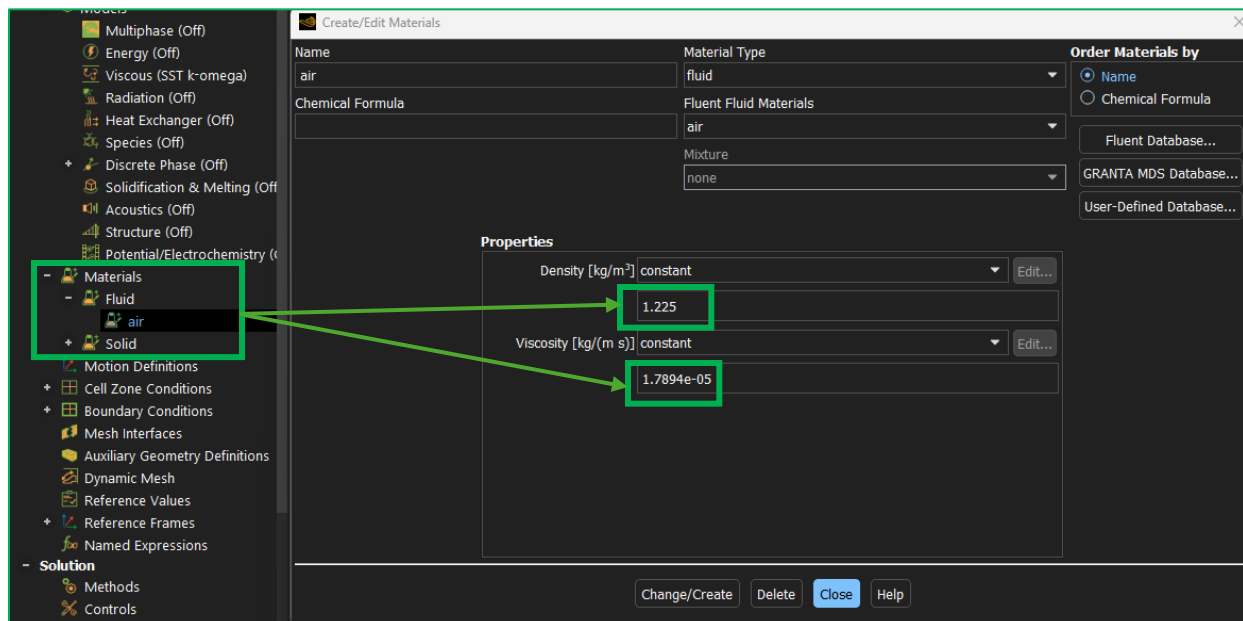
Now, let us define some values for the solver. Just follow the screenshots, little to no discussions are required. Although I encourage everyone to read about some of the values, parameters and what they mean – for a better understanding of Computational Fluid Dynamics.



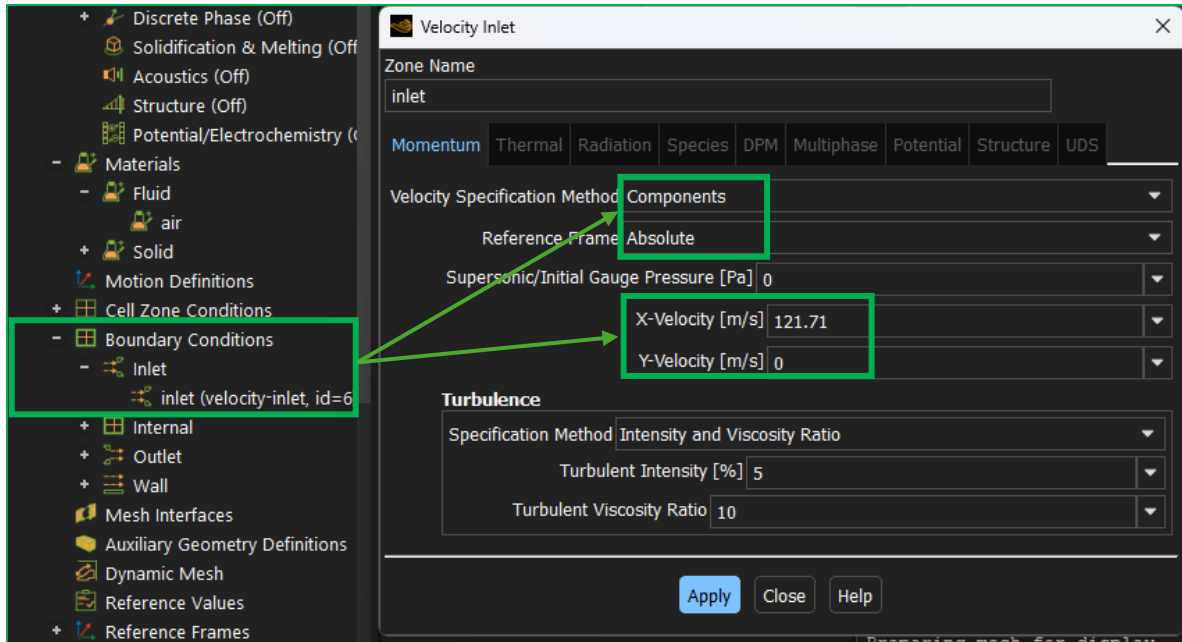
The general settings of the solver. You can leave 'Gravity' unchecked. For future references, gravity plays important roles in natural convection or buoyancy-driven flows (heated bodies for example), multiphase flows and similar conditions. For now, we can leave it unchecked.



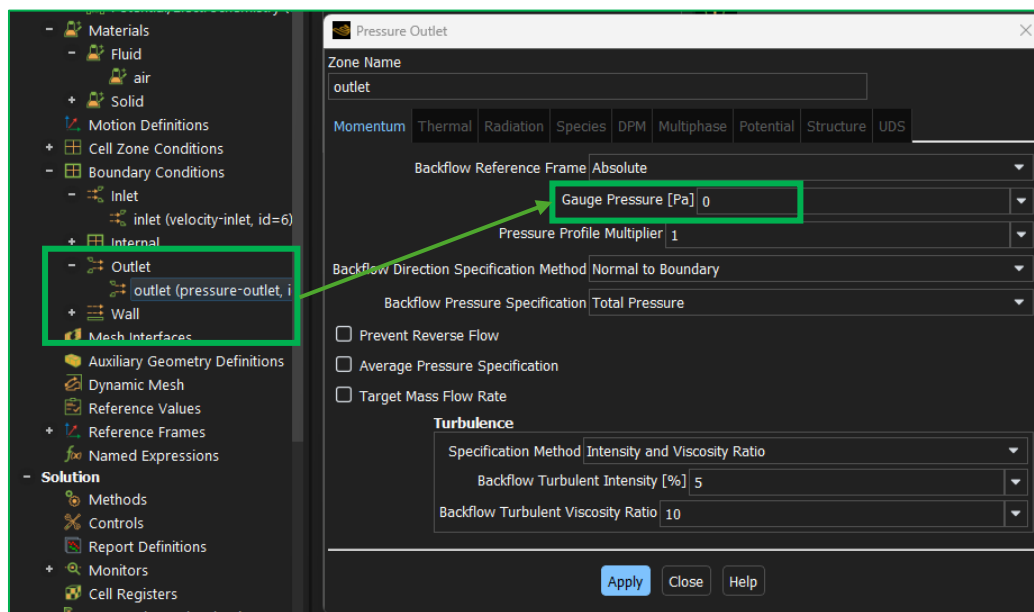
Reding about the SST k-omega equations are encouraged (we are becoming experts in CFD, why not season it with some mathematics?).



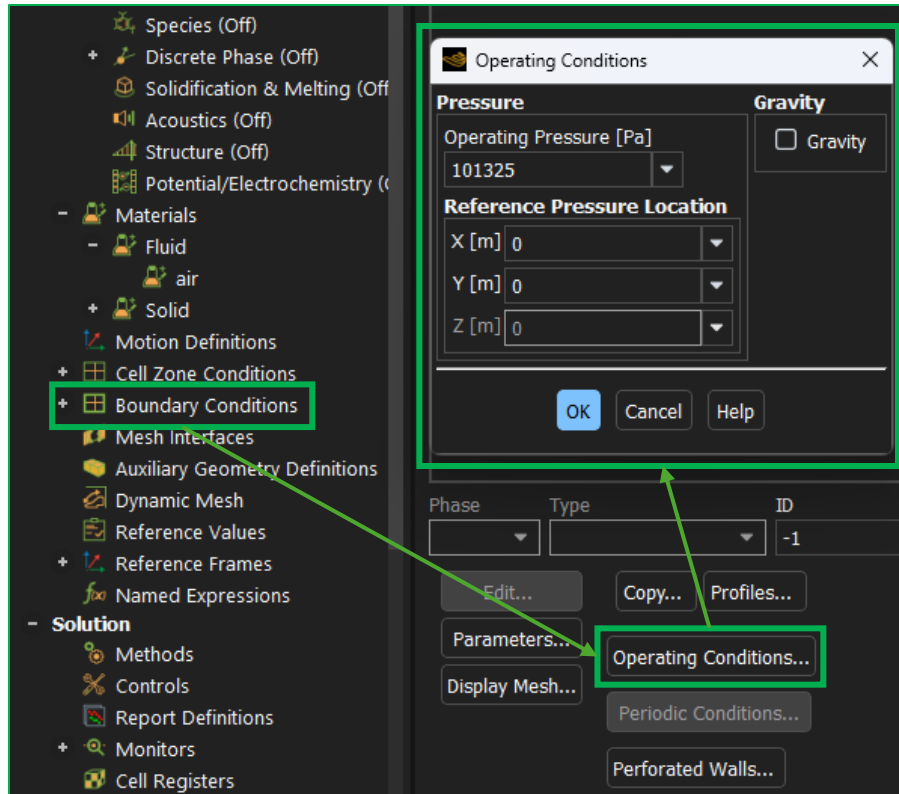
Assign the values according to the experimental parameters.



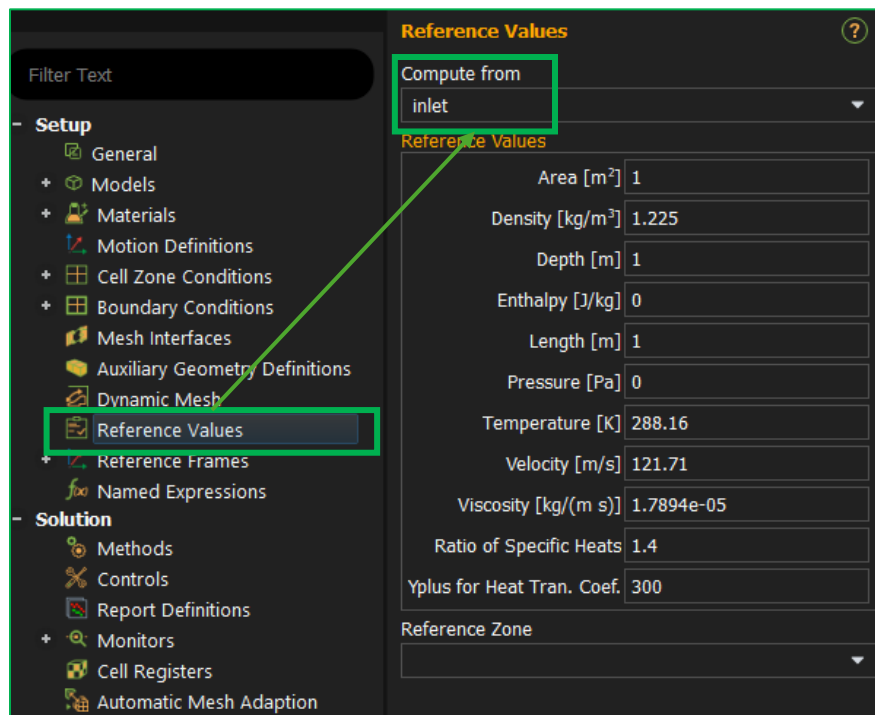
This is your dashboard to command the fluid flow. Using magics later on (trigonometry, and if you are a little crazy, some Python scripts!), we can modify the Angle of Attack by manipulating the components of the inlet velocity (Y-Velocity is 0 here since we are doing 0° AoA). And since this is relatively a 'low' speed simulation, we can skip Supersonic/Initial Gauge Pressure (which is basically any pressure present above 1 atm) and Turbulence parameters as default. Please refer to [this post](#) to understand how to manipulate Angel of Attack in ANSYS Fluent, with a bit of trigonometry and of course, magic (Python).



Leave as default. Just make sure there are no 'back pressure', i.e. – GP should be 0.

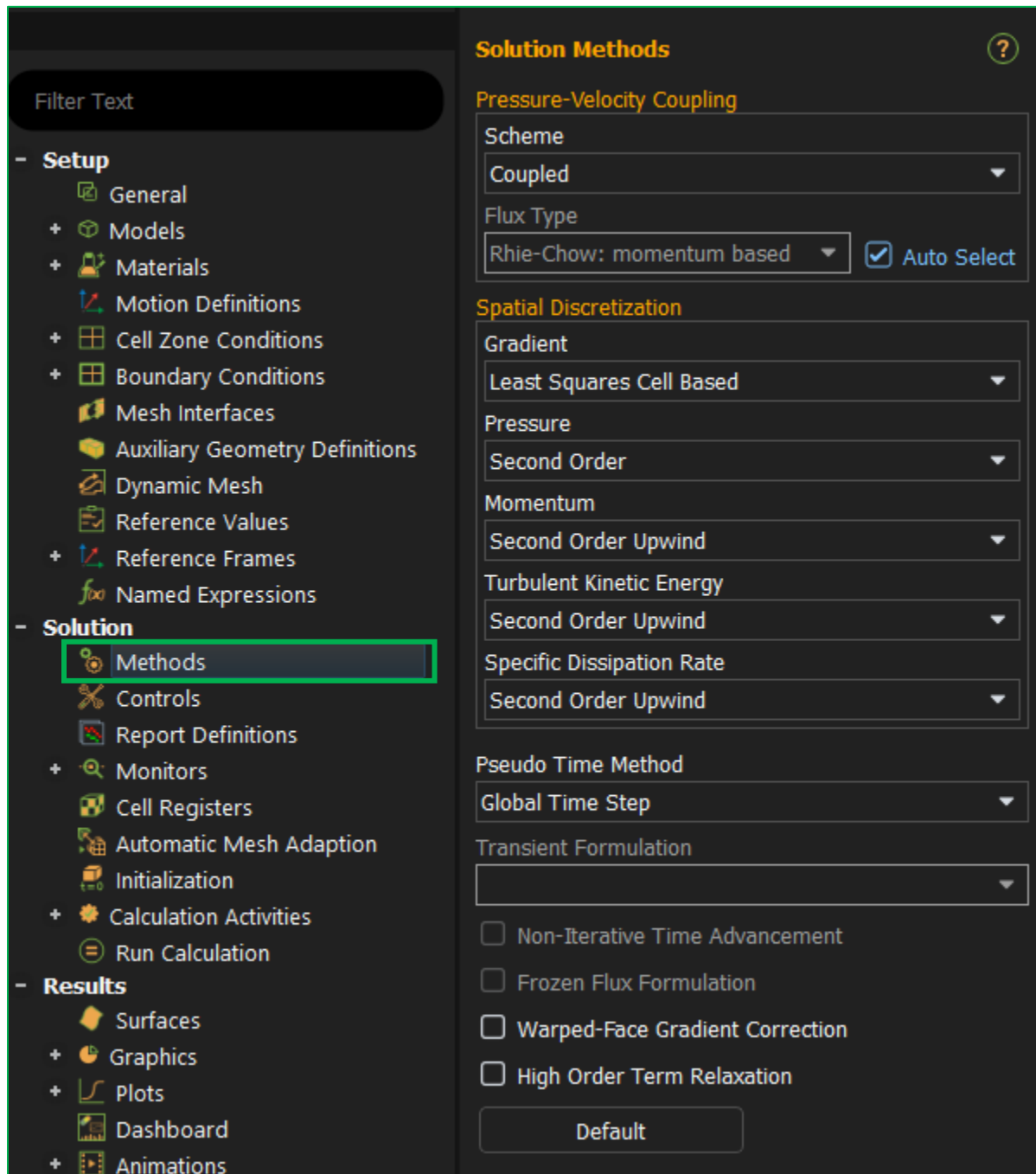


Double click on Boundary Conditions, click Operating Conditions and check if the values match (for an ideal fluid dynamic condition), shown above.

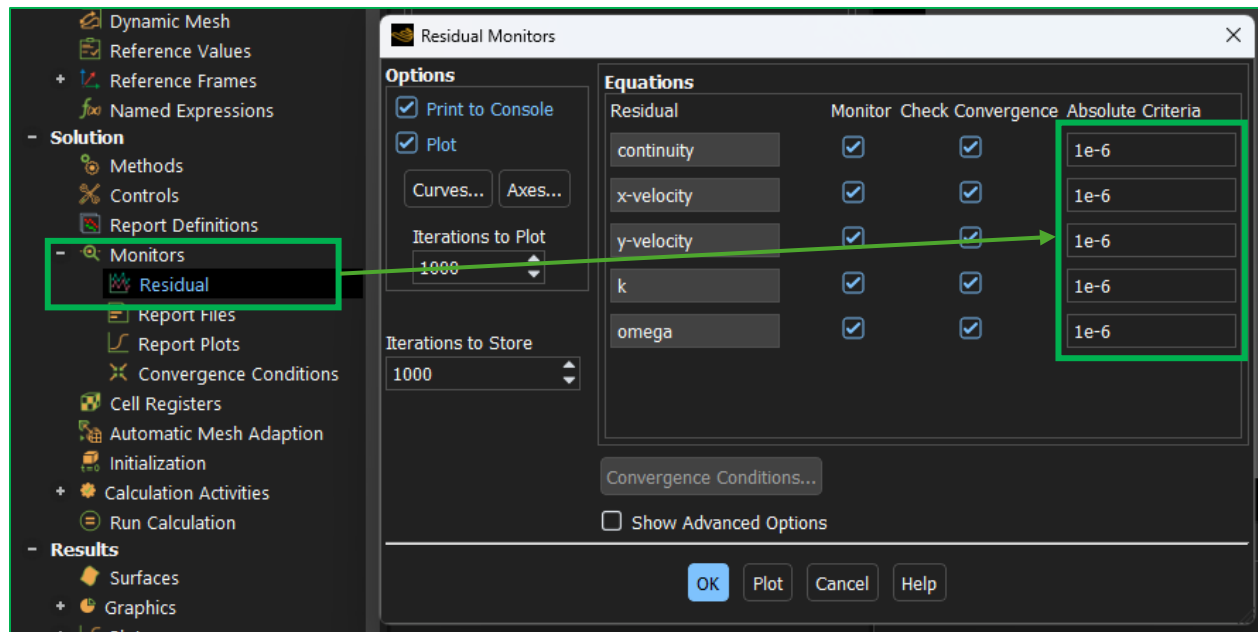


And finally, check the reference values and make sure it computes from inlet.

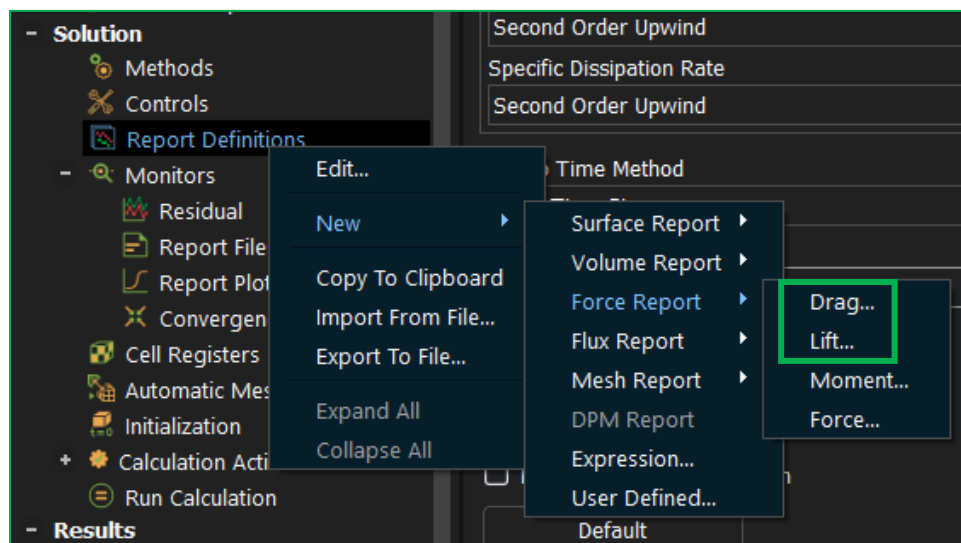
Now, we will define the solutions and ‘Initialize’ the solver for the party piece, the CFD itself (finally, oh my God this instruction is huge why did I decide to do it at the first place?).



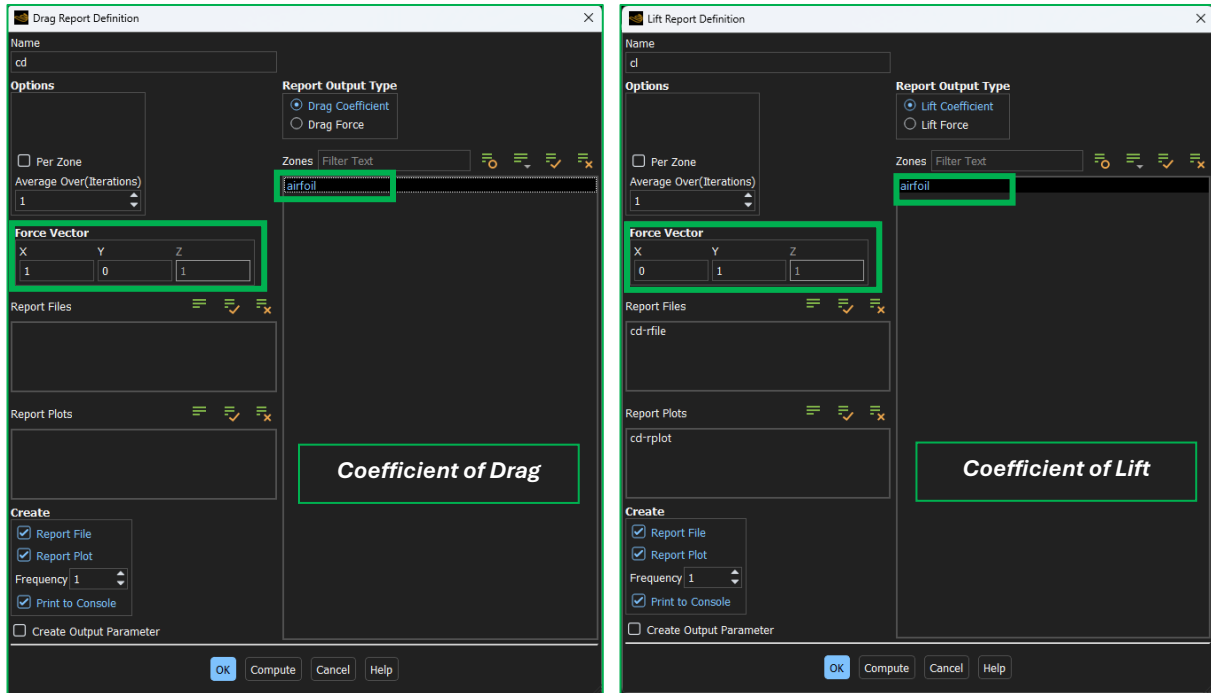
Selecting ‘Methods’ is just defining the mathematics the solver will be using to compute. By default, it should look like the screenshot. If not, please make sure all the options are as shown (double check if required). And I highly encourage to do some reading on ‘Second Order Upwind’ equations and how they ‘solve’ complex fluid dynamical problems. Navier-Stokes is a tough cookie, but not for us, is it? Yes, it still is!



Residuals measure how well the governing equations (mass, momentum, energy, etc.) are being satisfied at each iteration. Lower residuals mean the equations are closer to being perfectly balanced, i.e., less numerical error remains. By setting stricter residual targets, we are forcing the solver to reduce these imbalances further before stopping.



Before the solution, finally (this is the last step, I promise) we will define Drag and Lift coefficients – to see the output of the simulation and get an idea in real time on how the results are converging.



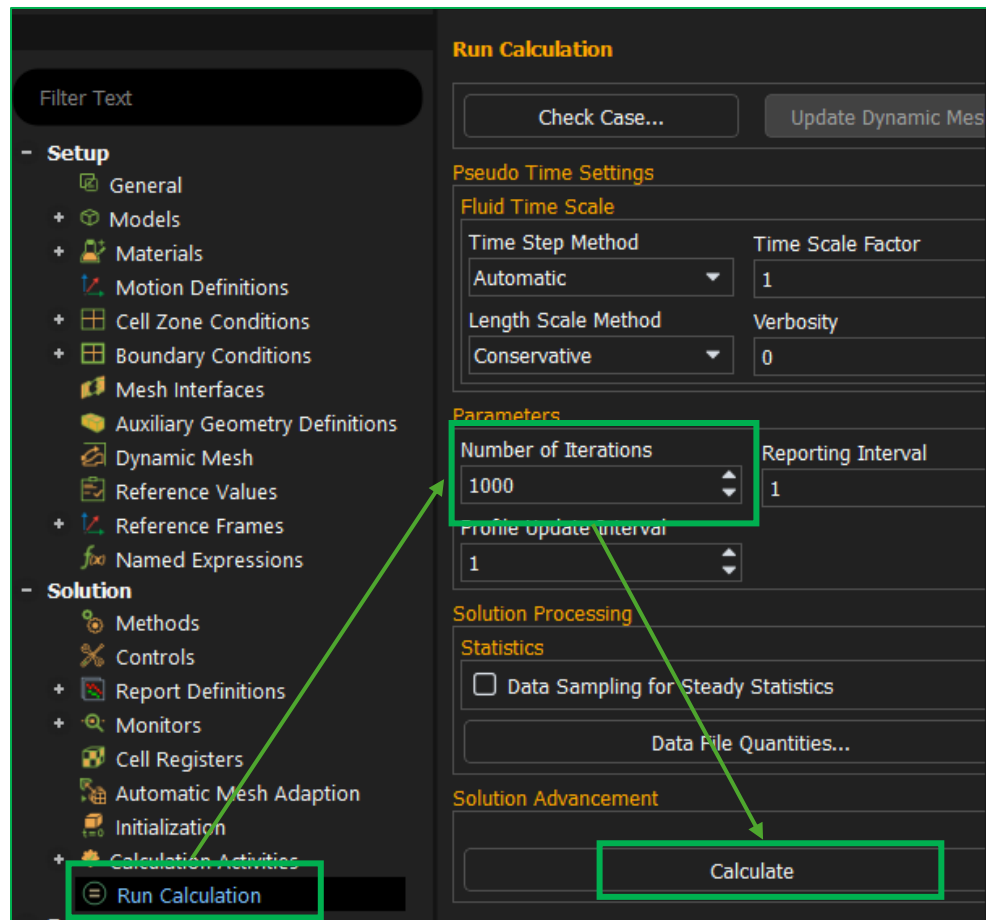
For our current setup (0° AoA), we will leave the values of force vectors as default. Later when we change the angle of attack, there will be some modifications to this section.

Initialization and Run

And just like that, we are at the final checkpoint of our CFD journey (for now of course).

Go to 'Initialization', select Hybrid and initialize the solution.

Once done, go to 'Run Calculation', setup Number of Iterations (1000 in our case) and hit Calculate.



Now go get some frappe. We are done! Finally!

(crowd goes wild)

As you are sipping your iced drink, it's time to see the solver in action – slowly but surely converging the result, one iteration at a time.

Very Important – once the solution is complete, or the solver hits the number of iterations (whichever is quicker), close the fluent solver properly by going to File (top left) – Close Fluent. Once in the workbench, save the project.

Massive achievement for the hopeful and clueless CFD students (including myself)!

([Result Discussions](#) for this particular simulation can be found on my website)

Omar Saif

CFD Enthusiast

August 15, 2025