# Numerical Heat Transfer

## Chapter 13 Application examples of fluent for basic flow and heat transfer problem

**Instructor Wen-Quan Tao; Qinlong Ren; Li Chen**

**CFD-NHT-EHT Center**
**Key Laboratory of Thermo-Fluid Science & Engineering**
**Xi'an Jiaotong University**

**Xi'an, 2019-Dec.-17**

# 数值传热学

## 第 13 章 求解流动换热问题的Fluent软件基础应用举例

主讲 陶文铨

辅讲 任秦龙，陈 黎

西安交通大学能源与动力工程学院
热流科学与工程教育部重点实验室

2019年12月17日，西安

# Chapter 13 Application examples of fluent for basic flow and heat transfer problem

**13.1 Heat transfer with source term**

**13.2 Unsteady cooling process of a steel ball**

**13.3 Lid-driven flow and heat transfer**

**13.4 Flow and heat transfer in a micro-channel**

**13.5 Flow and heat transfer in chip cooling**

**13.6 Phase change material melting with fins**

# 第 13 章 求解流动换热问题的Fluent软件应用举例

13.1 有内热源的导热问题

13.2 非稳态圆球冷却问题

导热问题

13.3 顶盖驱动流动换热问题

13.4 微通道内流动换热问题

混合对流问题

13.5 芯片冷却流动换热问题

微通道问题

13.6 肋片强化相变材料融化

相变传热

# 13.3 Lid-driven flow and heat transfer

## 顶盖驱动流动换热问题

**Focus:** compared with previous examples, the focus of this example is that fluid flow is further considered and moving wall boundary condition is adopted.

## 13.3 Lid-driven flow

**Known：**

An infinite long solid plate with uniform temperature $T_{w1} = 80$°C is moving at the top of a square cavity with velocity $u$=0.1m/s. The left and right walls of the cavity are adiabatic (绝热), while the temperature of bottom wall is fixed at $T_{w2} =$ 100°C. The effect of gravity is neglected.

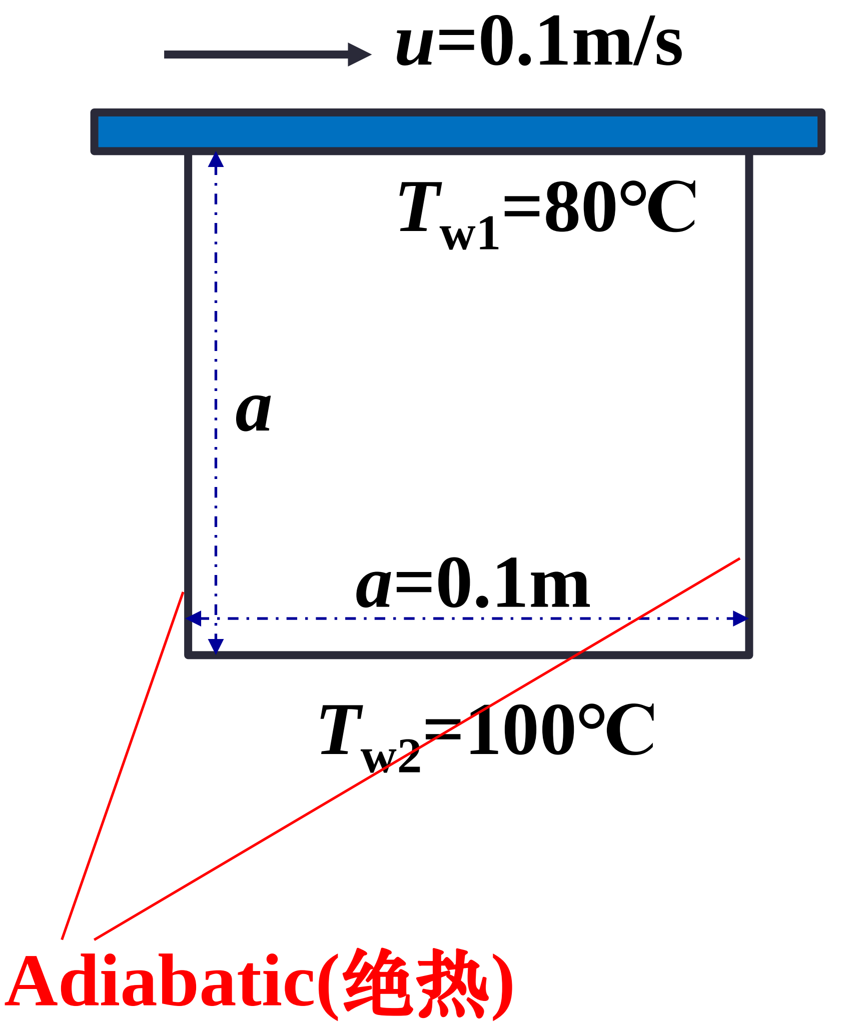



**Fig.1 Computational domain**

**Find:** **Velocity and temperature distribution**

**Solution:**

**Continuity:**

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0$$

**Momentum:**

$$u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}=-\frac{1}{\rho}\frac{\partial p}{\partial x}+\nu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)$$

$$u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}=-\frac{1}{\rho}\frac{\partial p}{\partial y}+\nu\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right)$$

**Energy:**

$$\frac{\partial(\rho C_p uT)}{\partial x}+\frac{\partial(\rho C_p vT)}{\partial y}=\lambda\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}\right)$$

**We should estimate Re to determine laminar or turbulent state.**

**Know:**

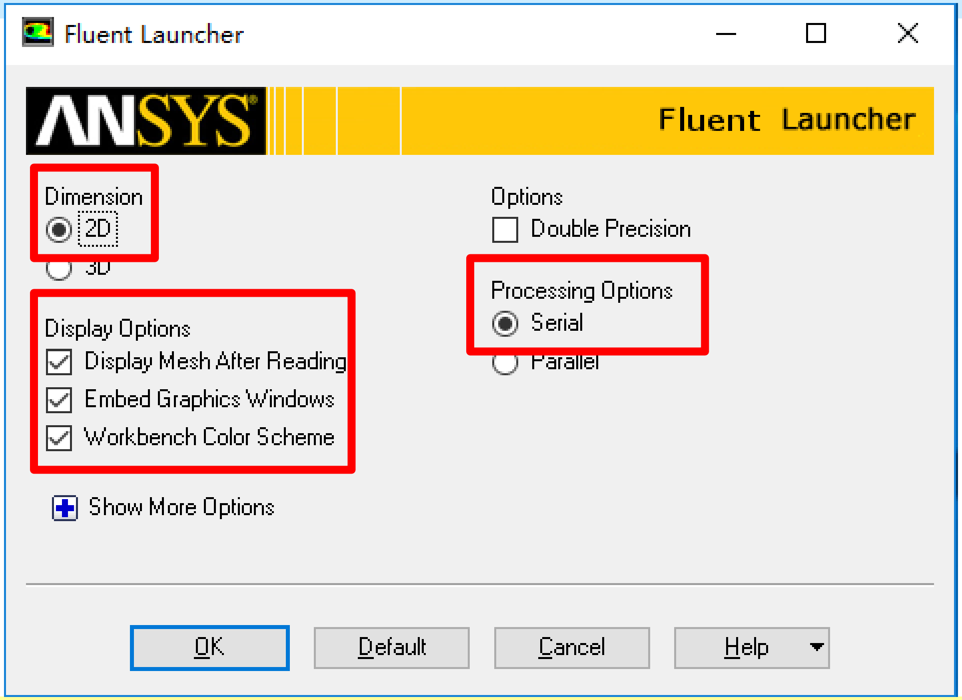
$u_{max} = 0.1\mathrm{m/s}$ ,$l$= 0.1m, $\nu = 1.46\mathrm{E}-6\mathrm{m}^2\mathrm{s}$

$$Re = \frac{ul}{\nu} = 684$$

**Laminar flow**

**Remark:** **in this problem, we just take into account the forced convection. Natural convection is neglected. You can further study the effects of natural convection!**

# Start the Fluent software



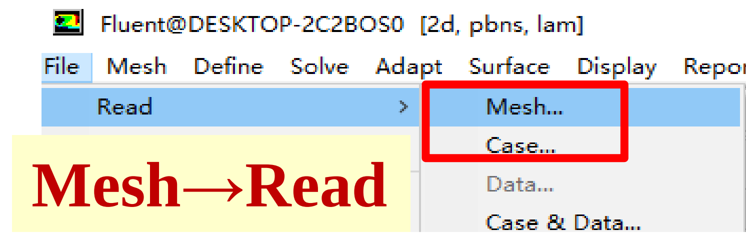

1. Choose 2-Dimension
2. Choose display options
3. Choose Serial processing option

**Note: Double precision or Single precision**

For most cases the single precision version of Fluent is sufficient. For heat transfer problem, if the thermal conductivity between different components is high, Double precision version is better.

# 1st step: Read and check the mesh

- The mesh is generated by pre-processing software such as **ICEM** and **GAMBIT. The document is with suffix（后缀名）“.msh”**
- This step is similar to the Grid subroutine (UGRID, Setup1) in our general code.



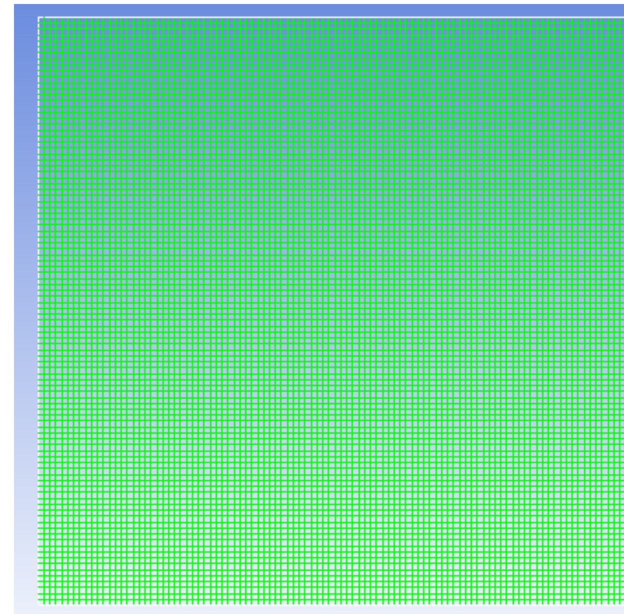

```
> Reading "E:\fluent-case\flow-5\flow2.cas"...
Done.
     9801 quadrilateral cells, zone  8, binary.
    19404 2D interior faces, zone  9, binary.
       99 2D wall faces, zone 10, binary.
       99 2D wall faces, zone 11, binary.
      198 2D wall faces, zone 12, binary.
    10000 nodes, binary.
    10000 node flags, binary.

Building...
     mesh
     materials,
     interface,
     domains,
        mixture
     zones,
        fixed-wall
        bottom-wall
        move-wall
        int_solid
        fluid
Done.
```

# 1st step: Read and check the mesh

## Mesh→Check/Report quality

- Check the quality and topological information of the mesh

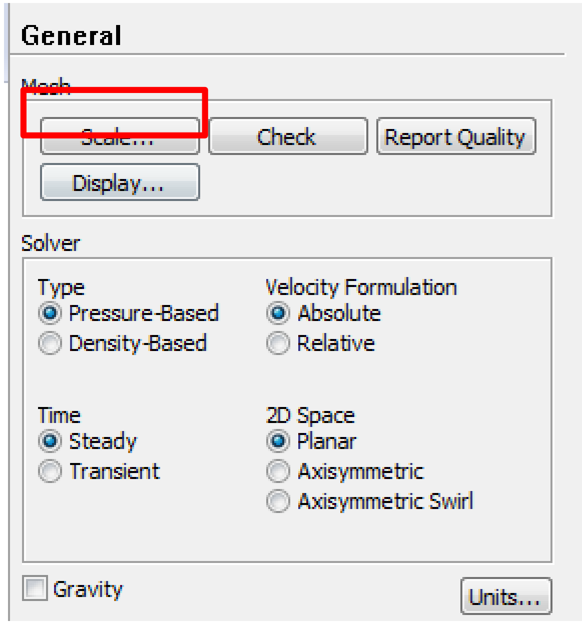


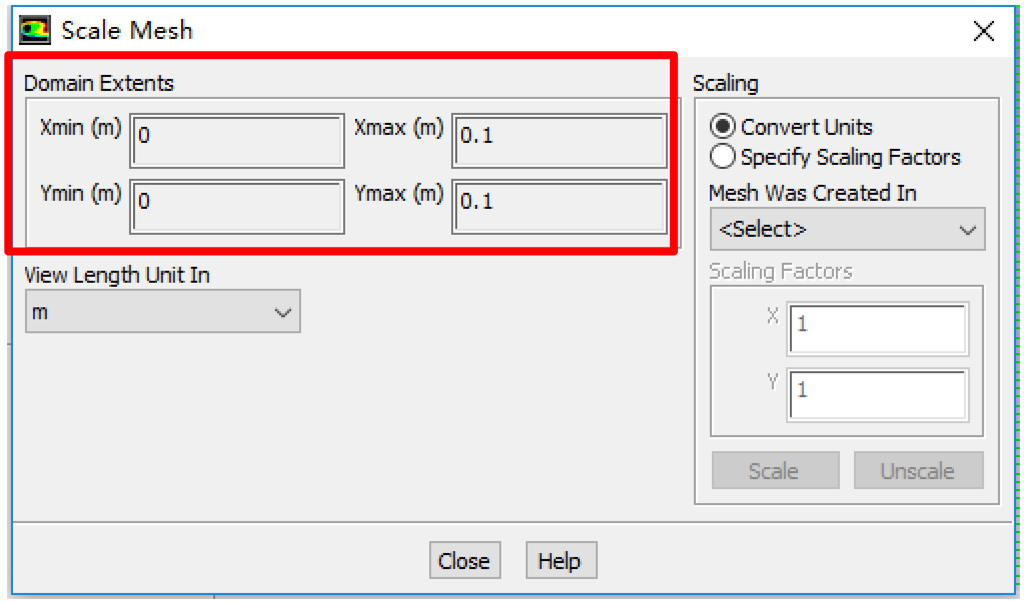
```
Domain Extents:
  x-coordinate: min (m) = 0.000000e+00, max (m) = 1.000000e-01
  y-coordinate: min (m) = 0.000000e+00, max (m) = 1.000000e-01
Volume statistics:
  minimum volume (m3): 1.020304e-06
  maximum volume (m3): 1.020304e-06
    total volume (m3): 1.000000e-02
Face area statistics:
  minimum face area (m2): 1.010101e-03
  maximum face area (m2): 1.010101e-03
Checking mesh.........................
Done.
```

```
Mesh Quality:
 Orthogonal Quality ranges from 0 to 1, where values close to 0 correspond to low quality.
Minimum Orthogonal Quality =  1.00000e+00
Maximum Aspect Ratio =  1.41422e+00
```

# 2st step: Scale the domain size

## General→Scale

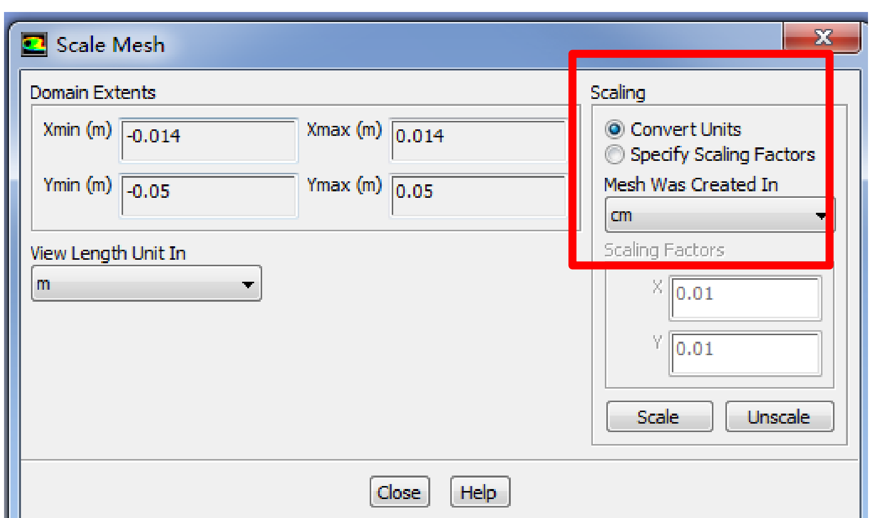


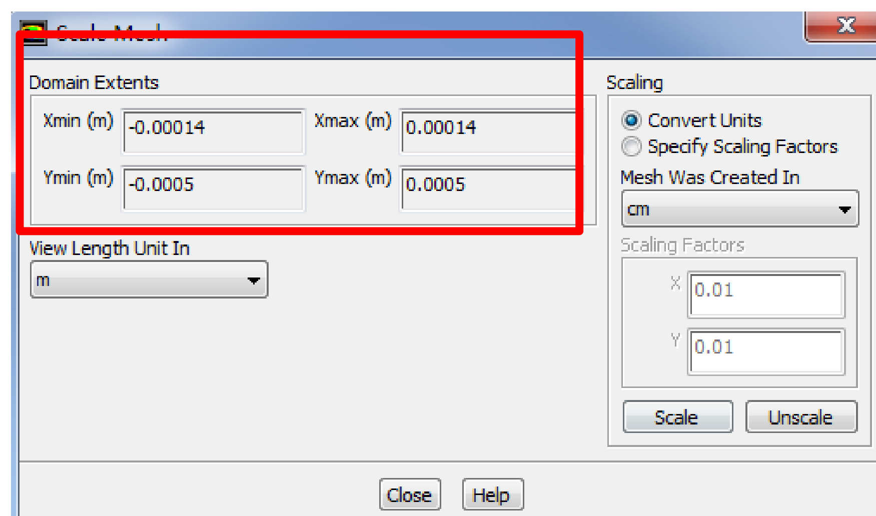




- Fluent stores the mesh in units as “m”, SI unit. You can show it in different units such as cm, mm, in, or ft.
- This time ,we don’t need to scale the mesh.

- **You also can scale the domain size use "Convert Units" or " Specify Scaling Factors" command.**

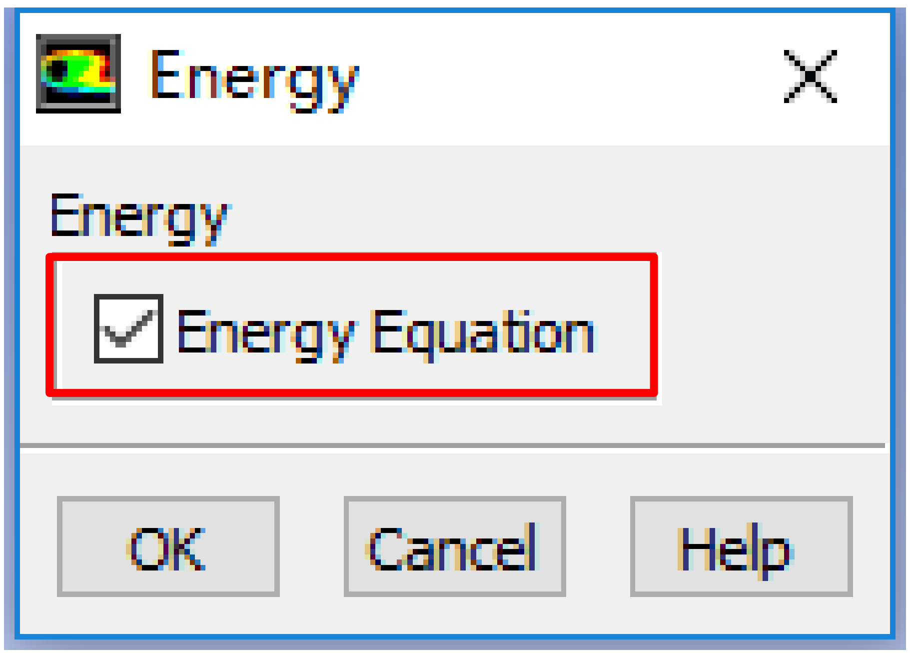
**Remark:** **Fluent thought you create the mesh in units of m. However, if your mesh is created in a different unit, such as cm, you must use Convert Units Command to change the mesh into the right size. The values will be multiplied by the Scaling Factor.**

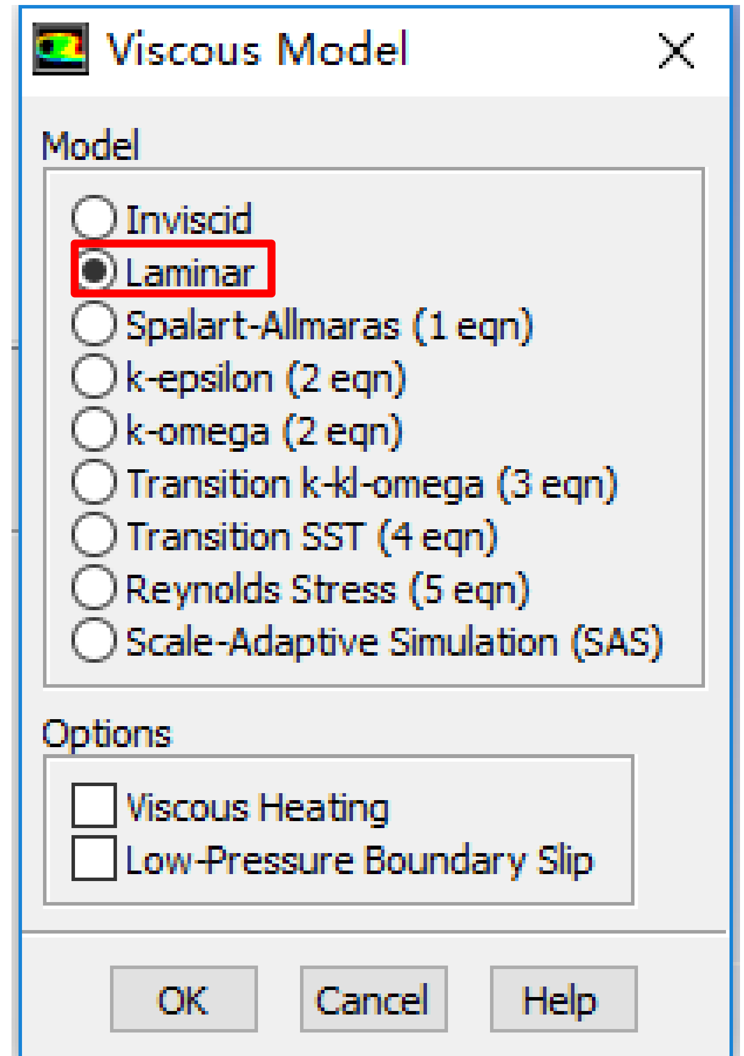


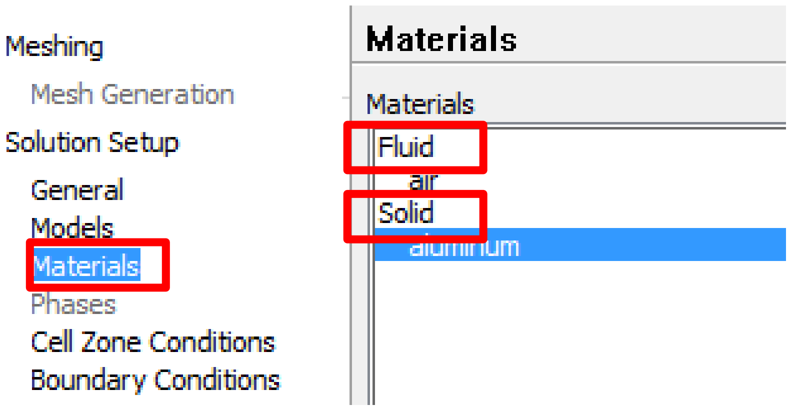
## Step 3: Choose the physicochemical model

**Based on the governing equations you are going to solve, select the related model in Fluent.**



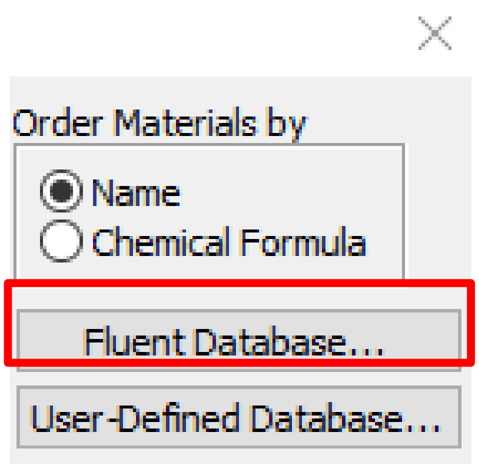



# Step 4: Define the materials




Click "Fluid" or "Solid"

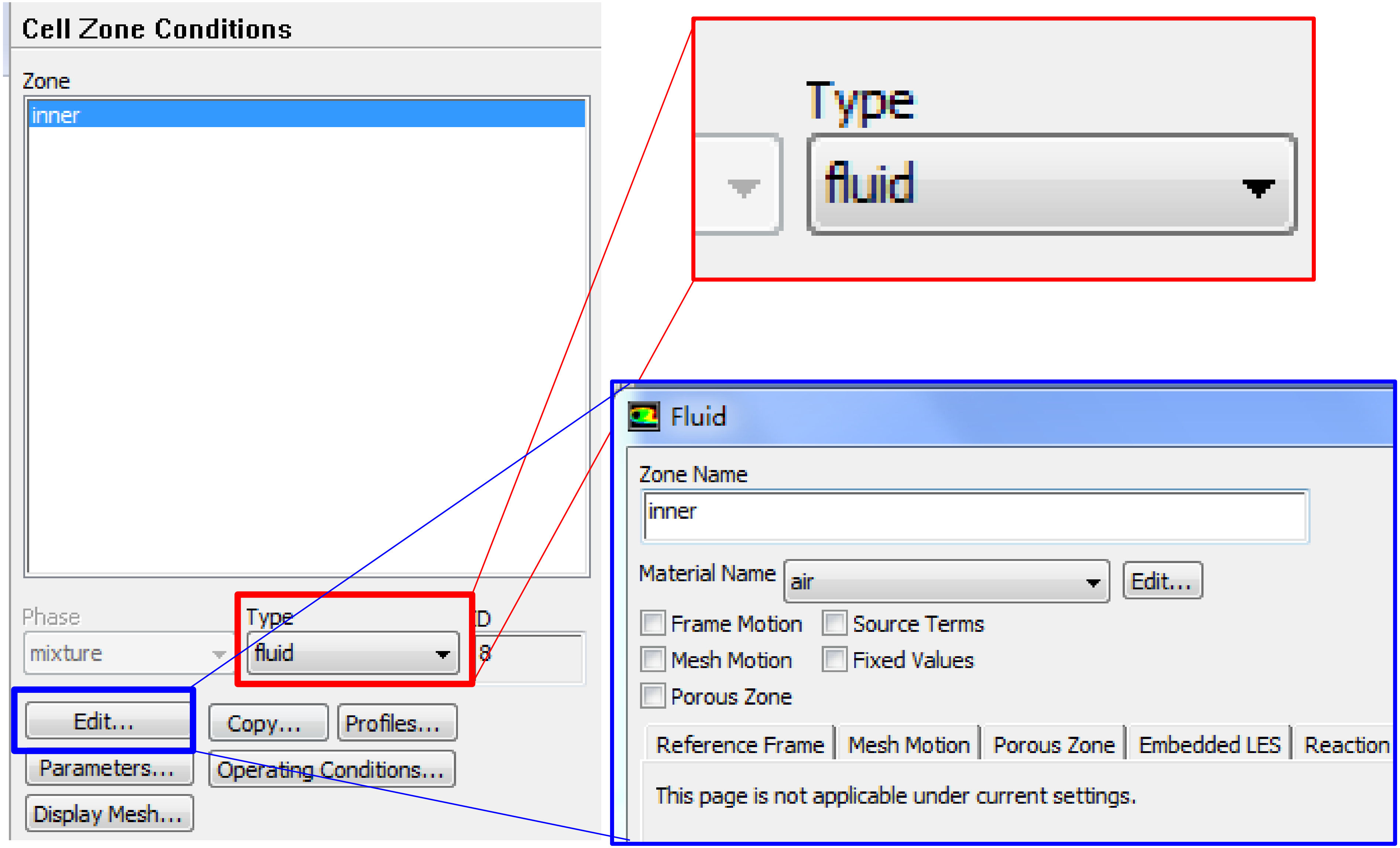
or select the "create/edit"

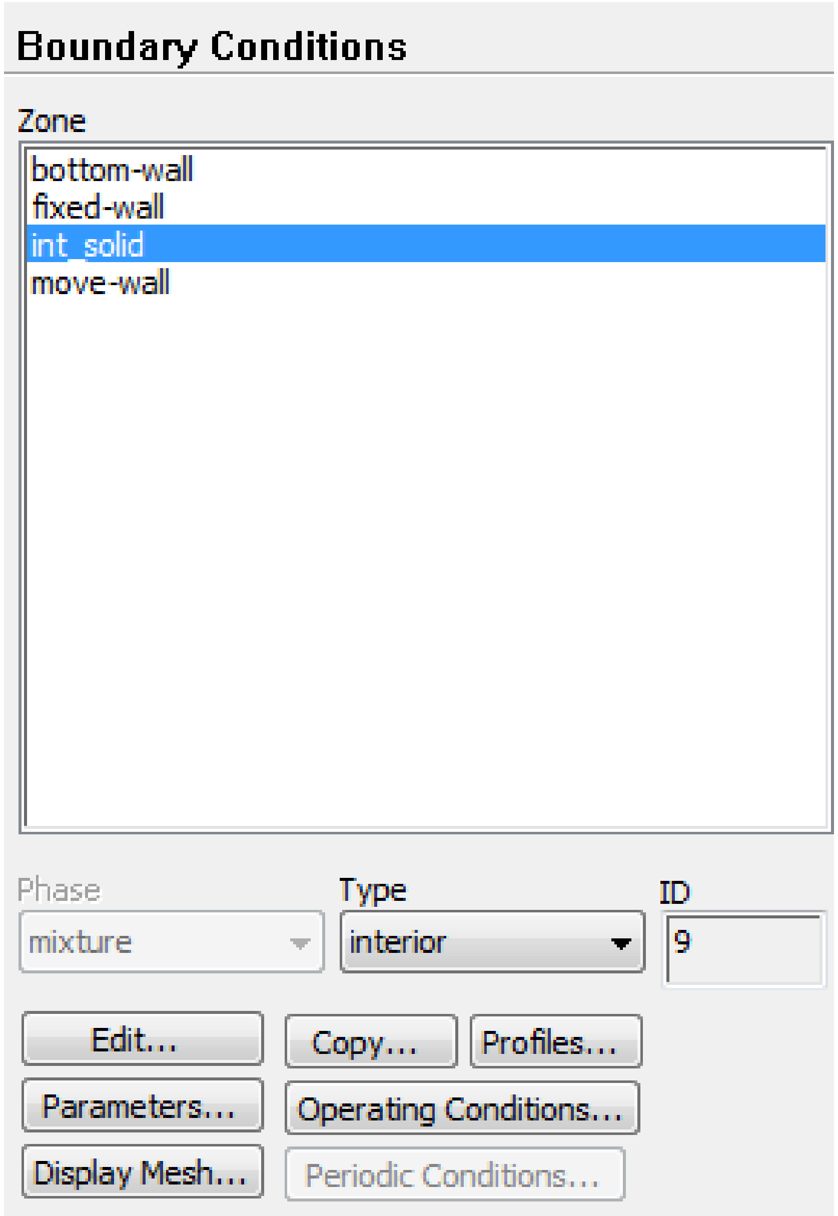

Fluent provide a lot of materials in its database. Usually, You can find the material you need in the database.

**However, it will happen that the material you need is not in the database. You can input it manually.**

# 5st step: Define the cell-zone condition



# 6st step: Define the Boundary conditions

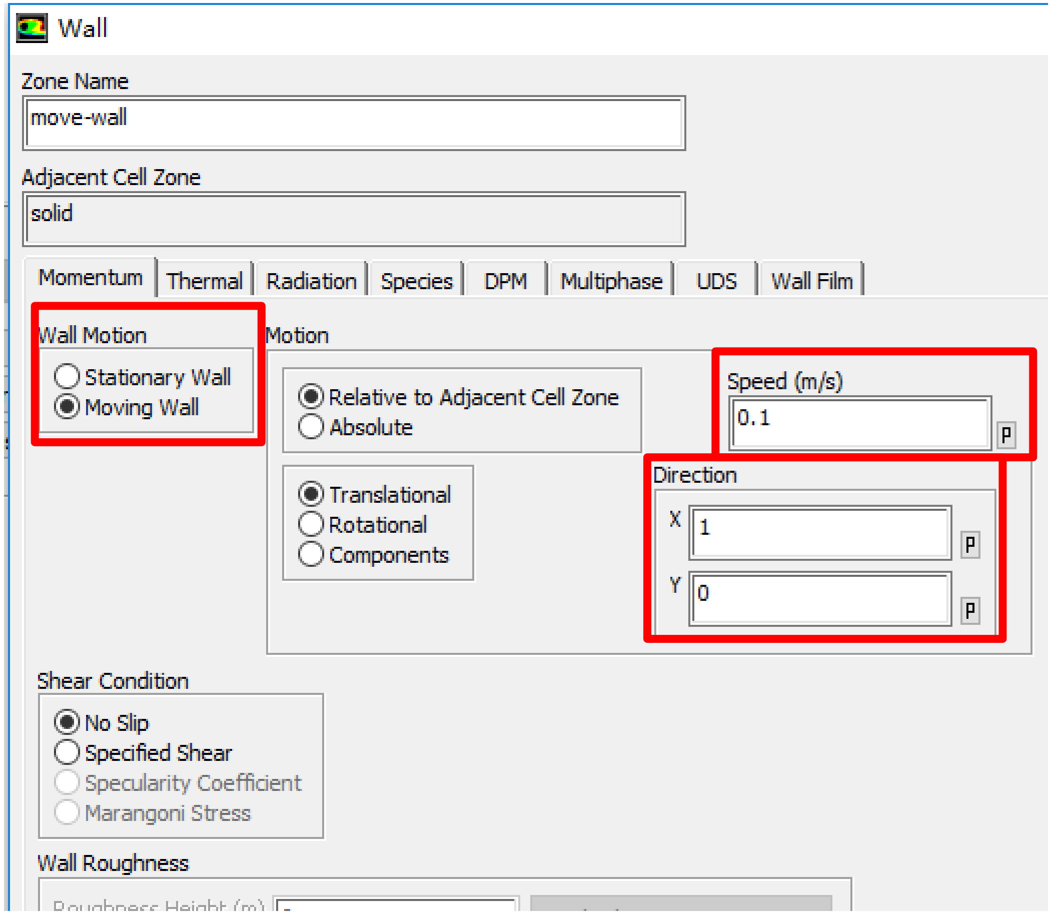



**The bottom wall is not moving and its temperature is 100°C. The left and right wall is adiabatic.**

**All these boundary conditions are easy to set in Fluent.**

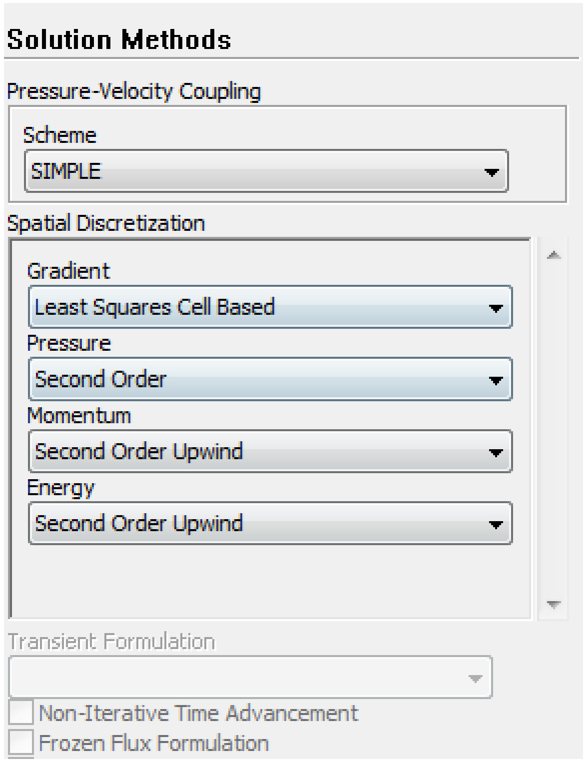
**The top wall is moving. We will discuss it in detail.**

**"Moving wall" is used to include tangential (切向) motion of the wall. This function cannot be used to include the normal (法向) motion of a wall.**



# 7st step: Define the solution

For algorithm and schemes, keep it as default. For more details of this step, one can refer to Example 1 of Chapter 13.




**Algorithm**: simple

**Gradient**: Least Square Cell Based

**Pressure**: second order

**Momentum**: second order upwind

**Energy**: second order Upwind

## 7st step: Define the solution

For under-relaxation factor, keep it default. For more details, refer to Example 1.

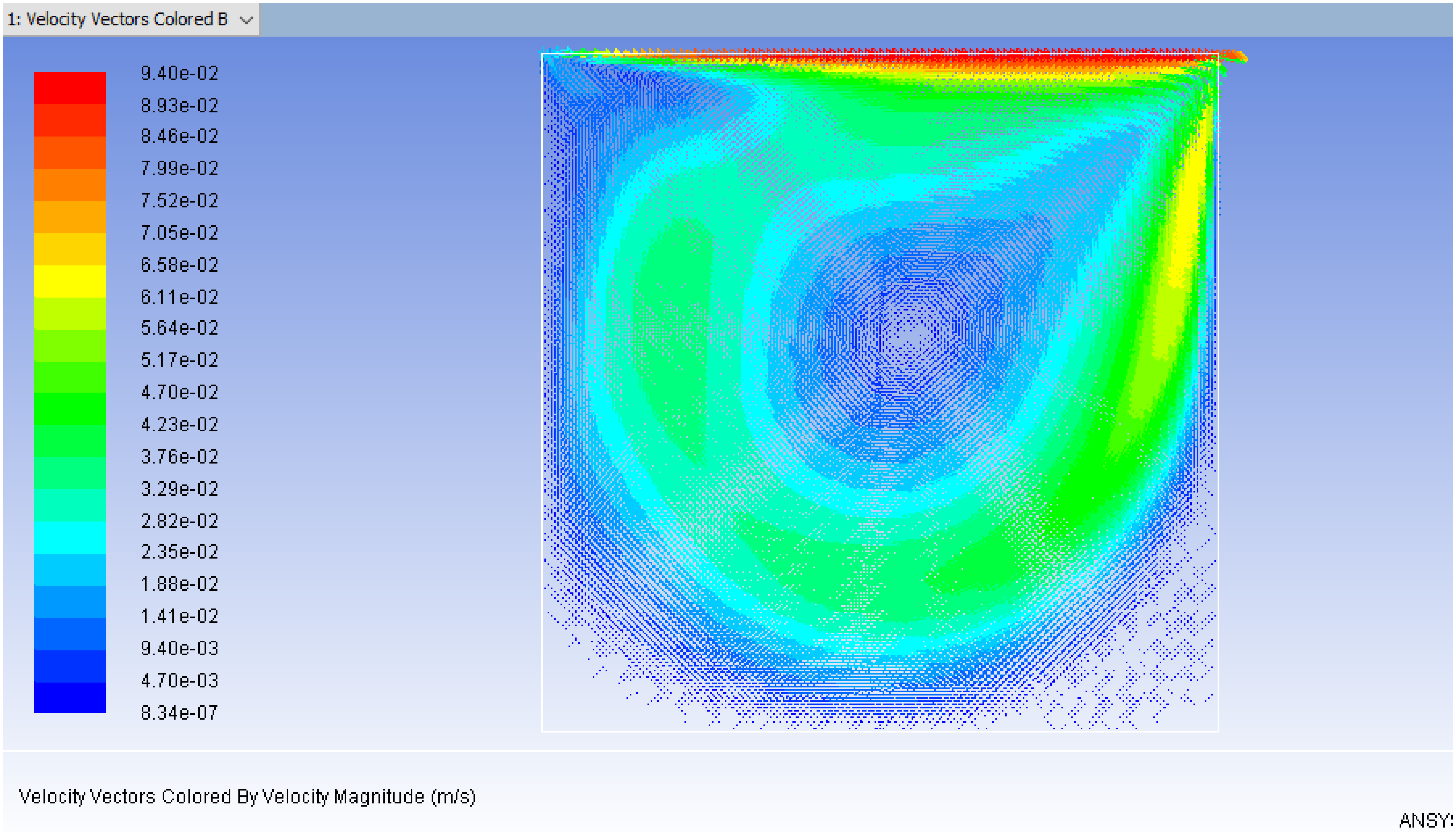
## 8st step: Initialization

Use the standard initialization, for more details of Hybrid initialization, refer to Example 1.

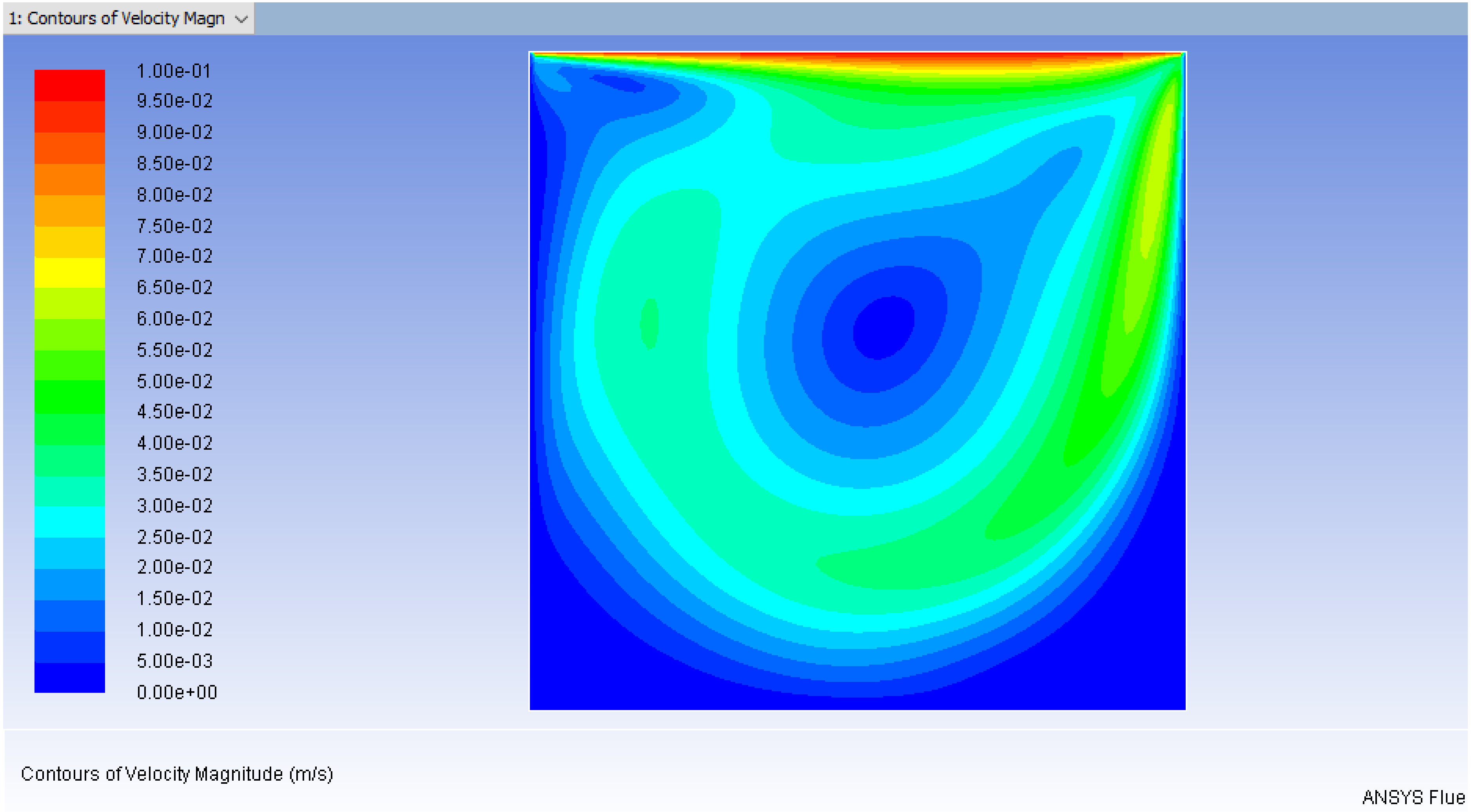
## Step 9: Run the simulation

## Step 10: Post-processing results

# Velocity Vector



# Velocity magnitude



# Temperature