Cartesian Heat Equation 2D

Solving the Equation

Boundary Condition

In[49]:=
```
v[t_] := 2;
```

Thermal Diffusivity Constant

In[50]:=
```
k = 1;
```

Finding the solution to the equation with conditions:

In[51]:=
```
solution = NDSolve[{
    D[u[x, y, t], t] == k * (D[u[x, y, t], x, x] + D[u[x, y, t], y, y]),
    (*Heat Equation in Cartesian*)
    u[x, y, 0] == 0, (*Initial Condition*)
    u[0, y, t] == 0, (*Left Side Boundary Condition*)
    u[1, y, t] == v[t] , (*Right Side*)
     u[x, 0, t] == 0, (*Lower Side*)
    u[x, 1, t] == 0}, (*Upper Side*)
   u[x, y, t], {x, 0, 1}, {y, 0, 1}, {t, 0, 10}, MaxStepSize → 0.01];

distribution[x_, y_, t_] = First[u[x, y, t] /. solution];
```

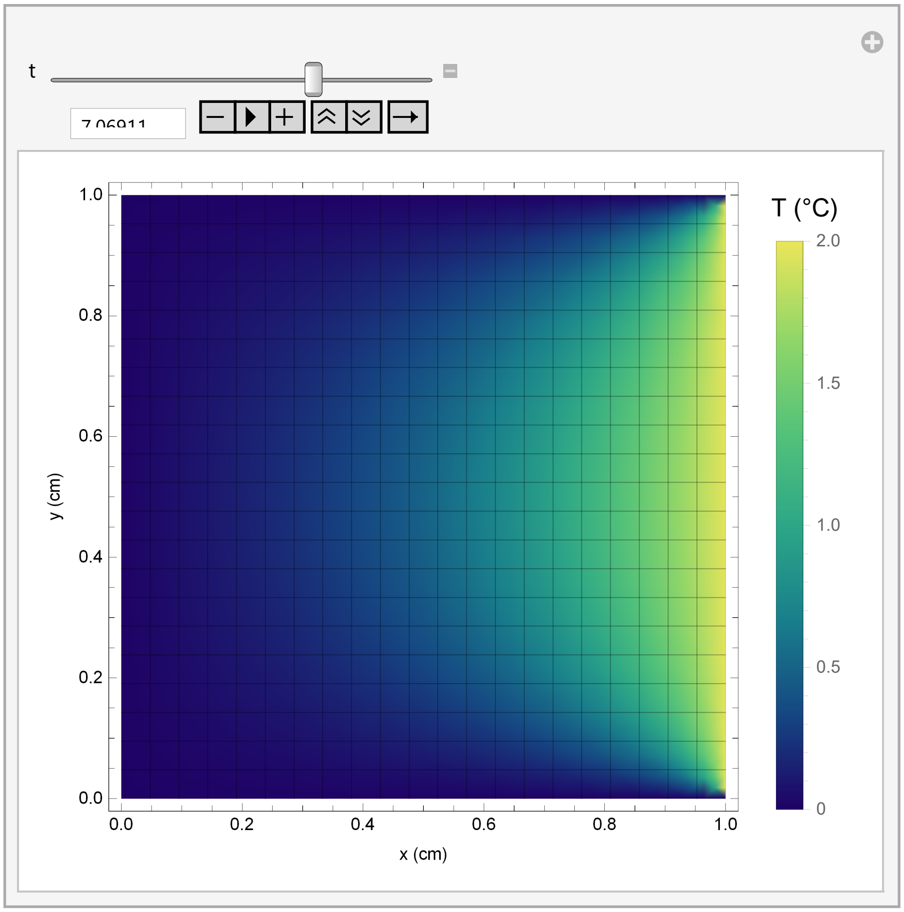
... NDSolve: Warning: boundary and initial conditions are inconsistent.

... NDSolve: Warning: scaled local spatial error estimate of 35.75439300084772` at t = 10.` in the direction of independent variable x is much greater than the prescribed error tolerance. Grid spacing with 100 points may be too large to achieve the desired accuracy or precision. A singularity may have formed or a smaller grid spacing can be specified using the MaxStepSize or MinPoints method options.

Visualizing the Solution:

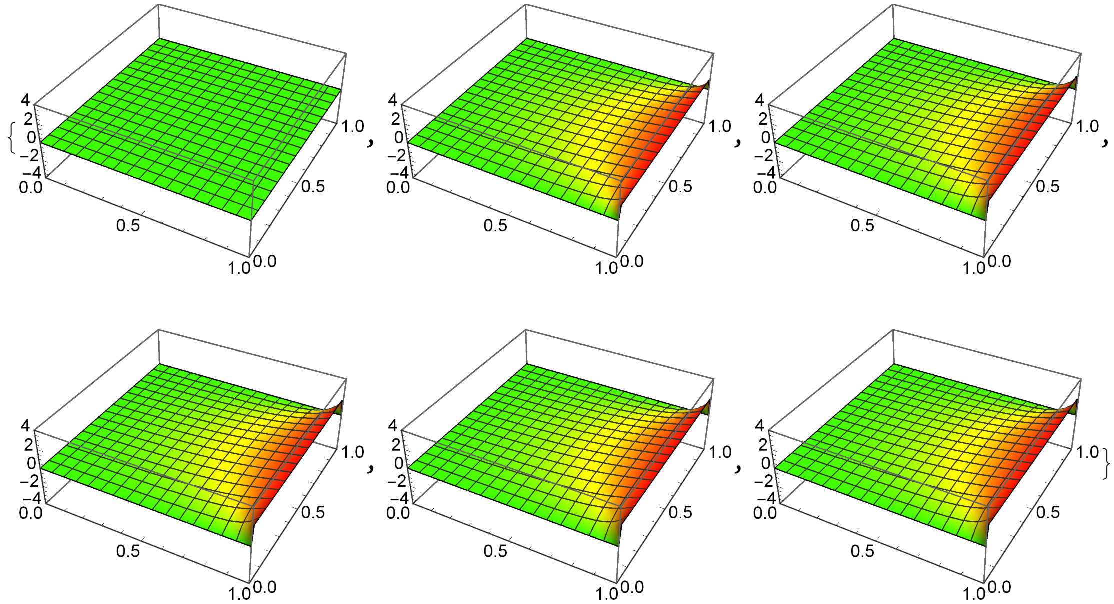
In[55]:=
```
Manipulate[DensityPlot[distribution[x, y, t],
  {x, 0, 1}, {y, 0, 1}, Mesh → 20, FrameLabel → {"x (cm)", "y (cm)"},
  ColorFunction → "BlueGreenYellow", ColorFunctionScaling → True,
  PlotLegends → BarLegend[Automatic, LegendMarkerSize → 320, LegendLabel → "T (°C)"],
  PerformanceGoal → "Quality"], {t, 0, 10}]
```

Out[55]=



In[ ]:=
```
a = Table[Plot3D[-First[u[x, y, t] /. solution],
    {x, 0, 1}, {y, 0, 1}, Mesh → 15, PlotRange → {{0, 1}, {0, 1}, {-4, 4}},
    ColorFunction → Function[{x, y, z}, Hue[.3 (1 - z)]]], {t, 0, 5}]
```

Out[ ]=



In[54]:=
```
Manipulate[Plot3D[-distribution[x, y, t], {x, 0, 1},
  {y, 0, 1}, Mesh → 15, PlotRange → {{0, 1}, {0, 1}, {-4, 4}},
  ColorFunction → Function[{x, y, z}, Hue[.3 (1 - z)]]], {t, 0, 10}]
```

Out[54]=