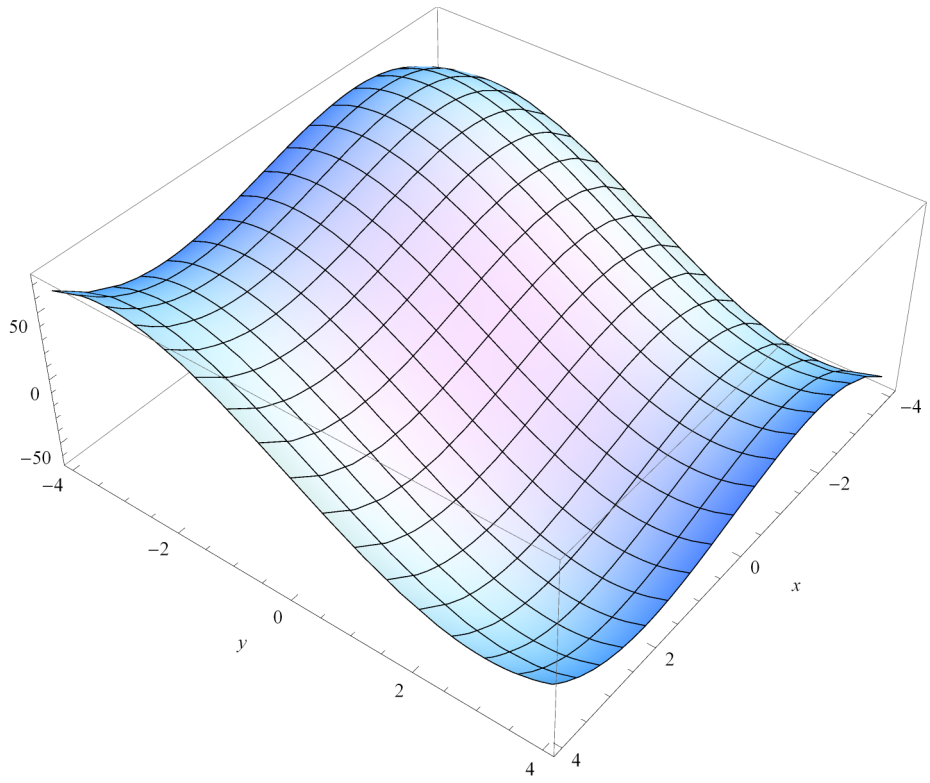
In[1]:=
```
f[x_, y_] := x^3 - 12 x + y^3 - 27 y + 5
```

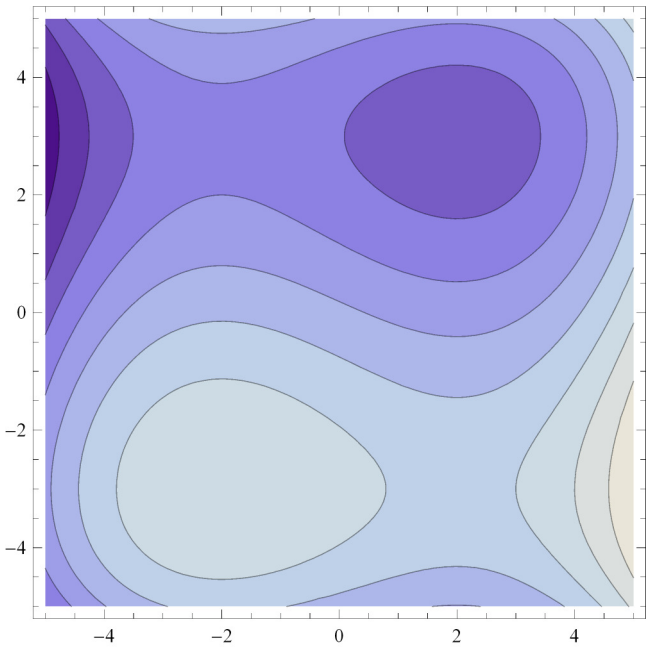
In[2]:=
```
Plot3D[f[x, y], {x, -4, 4}, {y, -4, 4}, AxesLabel → {x, y}]
```

Out[2]=



In[3]:=
```
ContourPlot[f[x, y], {x, -5, 5}, {y, -5, 5}]
```

Out[3]=



In[4]:=
```
DSolve[y'[x] - y[x] / x == Log[x], y[x], x]
```

Out[4]=

$$\left\{\left\{y[x] \to x\, C[1] + \frac{1}{2}\, x\, \text{Log}[x]^2\right\}\right\}$$

In[5]:=
```
DSolve[y'[x] - y[x] / x == E^x, y[x], x]
```

Out[5]=
```
{{y[x] → x C[1] + x ExpIntegralEi[x]}}
```

In[6]:=
```
DSolve[{y'[x] == 3 / 2 y[x]^(1 / 3), y[0] == 0}, y[x], x]
```

Out[6]=
```
{{y[x] → x^(3/2)}}
```

In[7]:=
```
DSolve[y'[x] == (x^2 + y[x]^2) / (x y[x]), y[x], x]
```

Out[7]=
```
{{y[x] → -Sqrt[x^2 C[1] + 2 x^2 Log[x]]}, {y[x] → Sqrt[x^2 C[1] + 2 x^2 Log[x]]}}
```

In[8]:=
```
DSolve[{y'[x] == (x^2 + y[x]^2) / (x y[x]), y[1] == 2}, y[x], x]
```

DSolve::bvnul :
For some branches of the general solution, the given boundary conditions lead to an empty solution. ≫

Out[8]=
```
{{y[x] → Sqrt[2] Sqrt[x^2 (2 + Log[x])]}}
```

In[9]:=
```
DSolve[{y'[x] == (x^2 + y[x]^2) / (x y[x]), y[1] == -2}, y[x], x]
```

DSolve::bvnul :
For some branches of the general solution, the given boundary conditions lead to an empty solution. ≫

Out[9]=
```
{{y[x] → -Sqrt[2] Sqrt[x^2 (2 + Log[x])]}}
```

In[10]:=
```
DSolve[{y'[x] == E^(-x y[x])}, y[x], x]
```

Solve::ifun :
Inverse functions are being used by Solve, so some solutions may not be found; use Reduce for complete solution information. ≫

Solve::ifun :
Inverse functions are being used by Solve, so some solutions may not be found; use Reduce for complete solution information. ≫

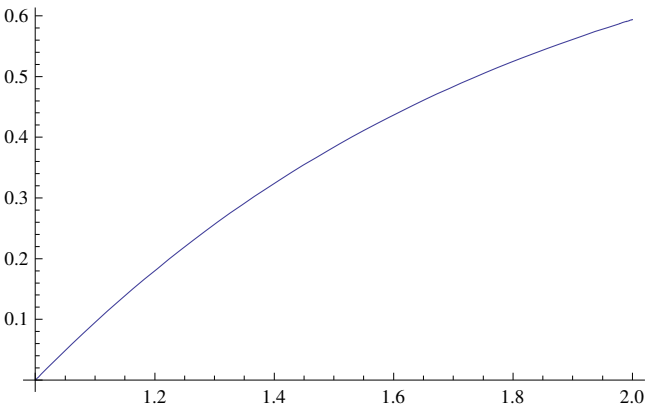
Out[10]=
```
DSolve[{y'[x] == e^(-x y[x])}, y[x], x]
```

In[11]:=
```
s = NDSolve[{y'[x] == e^(-x y[x]), y[1] == 0}, y, {x, 1, 2}]
```

Out[11]=
```
{{y → InterpolatingFunction[{{1., 2.}}, <>]}}
```

In[12]:=
```
Plot[Evaluate[y[x] /. s], {x, 1, 2}, PlotRange → All]
```

Out[12]=



In[13]:=
```
y[2] /. s
```

Out[13]=
```
{0.593785}
```

In[14]:=
```
n = 1000; x = 1; y = 0;
For[i = 1, i ≤ n, i++, {x = x + 1 / n, y = N[y + 1 / n E^(- x y)]}]; Print[{x, y}]
```

```
{2, 0.593882}
```

In[16]:=
```
Clear[x, y]
```

In[17]:=
```
DSolve[y''[x] - y'[x] - 6 y[x] == E^x, y[x], x]
```

Out[17]=
$$\left\{\left\{y[x] \to -\frac{e^x}{6} + e^{-2x} C[1] + e^{3x} C[2]\right\}\right\}$$