1 - 8 Scalar fields in the plane

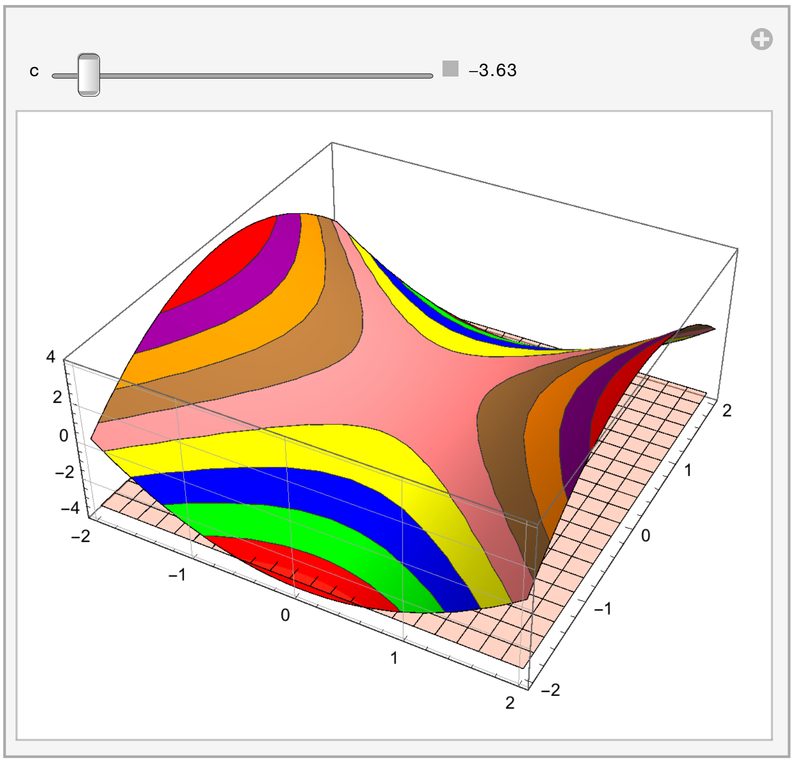
Let the temperature T in a body be independent of z so that it is given by a scalar function $T = T(x, t)$. Identify the isotherms $T(x, y) =$ const. Sketch some of them.

1. $T = x^2 - y^2$

```
Clear["Global`*"]
```

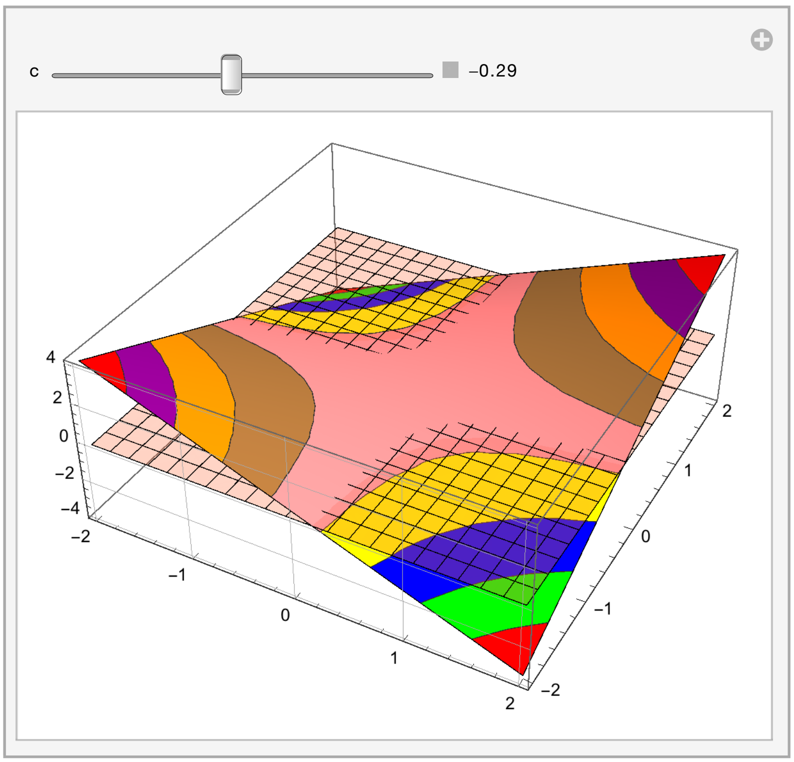
Below: Some of the isotherms are shown, (8). The idea behind the controllable plane is that changing the altitude of the grid strengthens awareness about the behavior of the function in its volume.

```
Manipulate[v1 = {1, 0, 0};
v2 = {0, 1, 0};
p0 = {0, 0, c};
plane1 = p0 + s * v1 + t * v2;
Show[Plot3D[{x^2 - y^2}, {x, -2, 2}, {y, -2, 2},
   Mesh → 8, MeshFunctions → {#3 &}, FaceGrids → {{0, -1, 0}},
   ColorFunction → Function[{x, y, z}, Hue[z]],
   ColorFunctionScaling → True, MeshShading →
    {Red, Green, Blue, Yellow, Pink, Brown, Orange, Purple}],
  Plot3D[{plane1}, {x, -2, 2}, {y, -2, 2},
   PlotStyle → Directive[RGBColor[1, 0.5, 0.35, 1], Opacity[0.3]],
   MeshStyle → Black, Mesh → 20]],
 {{c, -4.00}, -4, 4, Appearance → "Labeled"}]
```



2. T=xy.

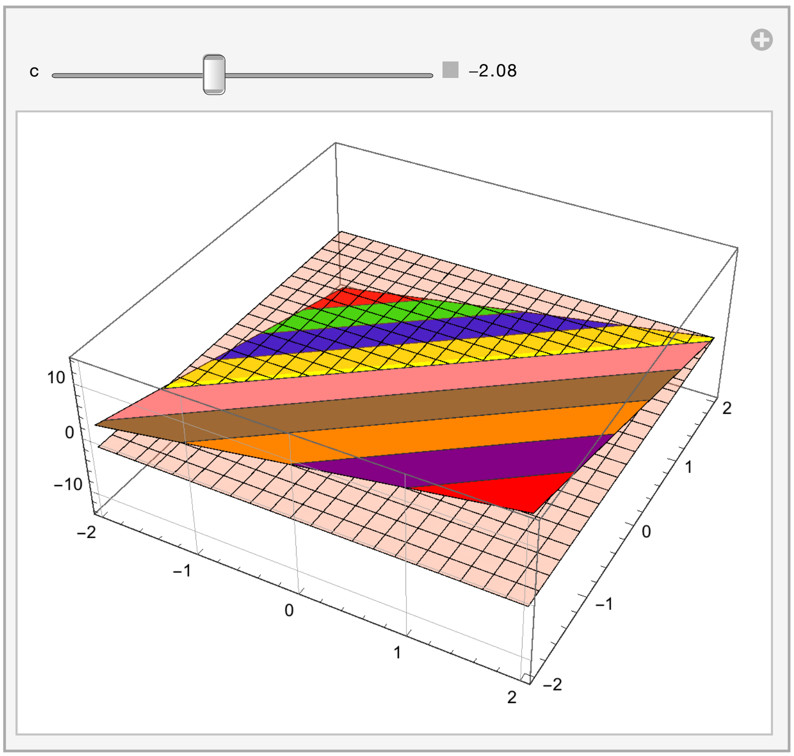
```
Manipulate[v1 = {1, 0, 0};
v2 = {0, 1, 0};
p0 = {0, 0, c};
plane1 = p0 + s * v1 + t * v2;
Show[
  Plot3D[{x y}, {x, -2, 2}, {y, -2, 2}, Mesh → 8, MeshFunctions → {#3 &},
   FaceGrids → {{0, -1, 0}}, ColorFunction → Function[{x, y, z}, Hue[z]],
   ColorFunctionScaling → True, MeshShading →
    {Red, Green, Blue, Yellow, Pink, Brown, Orange, Purple}],
  Plot3D[{plane1}, {x, -2, 2}, {y, -2, 2},
   PlotStyle → Directive[RGBColor[1, 0.5, 0.35, 1], Opacity[0.3]],
   MeshStyle → Black, Mesh → 20]],
 {{c, 0.00}, -4, 4, Appearance → "Labeled"}]
```



3. T = 3 x - 4 y

```
Clear["Global`*"]
```

```
Manipulate[v1 = {1, 0, 0};
v2 = {0, 1, 0};
p0 = {0, 0, c};
plane1 = p0 + s * v1 + t * v2;
Show[Plot3D[{3 x - 4 y}, {x, -2, 2}, {y, -2, 2},
   Mesh → 8, MeshFunctions → {#3 &}, FaceGrids → {{0, -1, 0}},
   ColorFunction → Function[{x, y, z}, Hue[z]],
   ColorFunctionScaling → True, MeshShading →
    {Red, Green, Blue, Yellow, Pink, Brown, Orange, Purple}],
  Plot3D[{plane1}, {x, -2, 2}, {y, -2, 2},
   PlotStyle → Directive[RGBColor[1, 0.5, 0.35, 1], Opacity[0.3]],
   MeshStyle → Black, Mesh → 20]],
 {{c, -10.00}, -12, 12, Appearance → "Labeled"}]
```

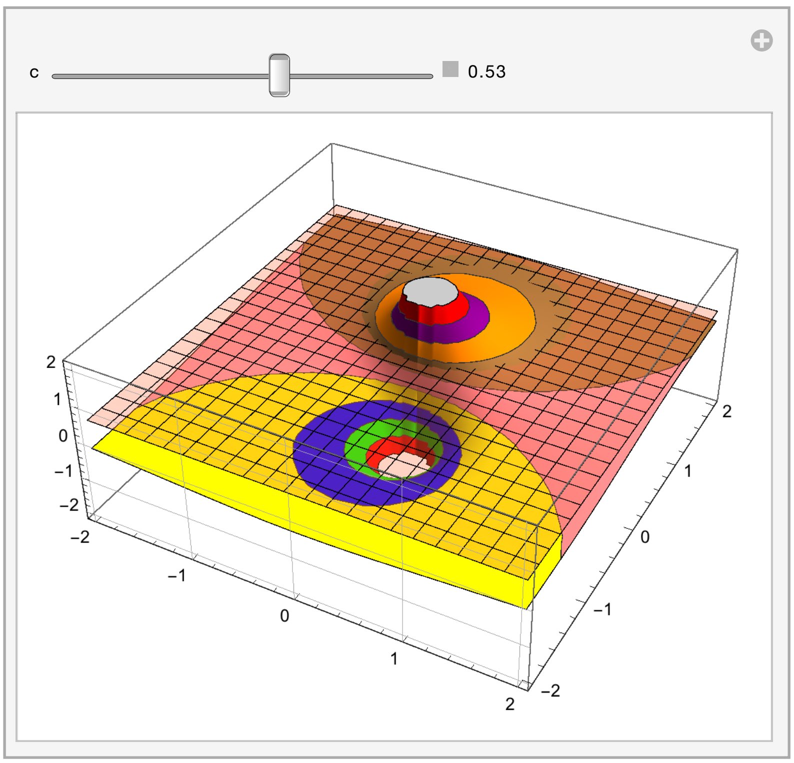


5. $T = \dfrac{y}{(x^2 + y^2)}$

```
Clear["Global`*"]
```

Below: this one goes into a strange shape-shifting mode while the slider is being manipulated.

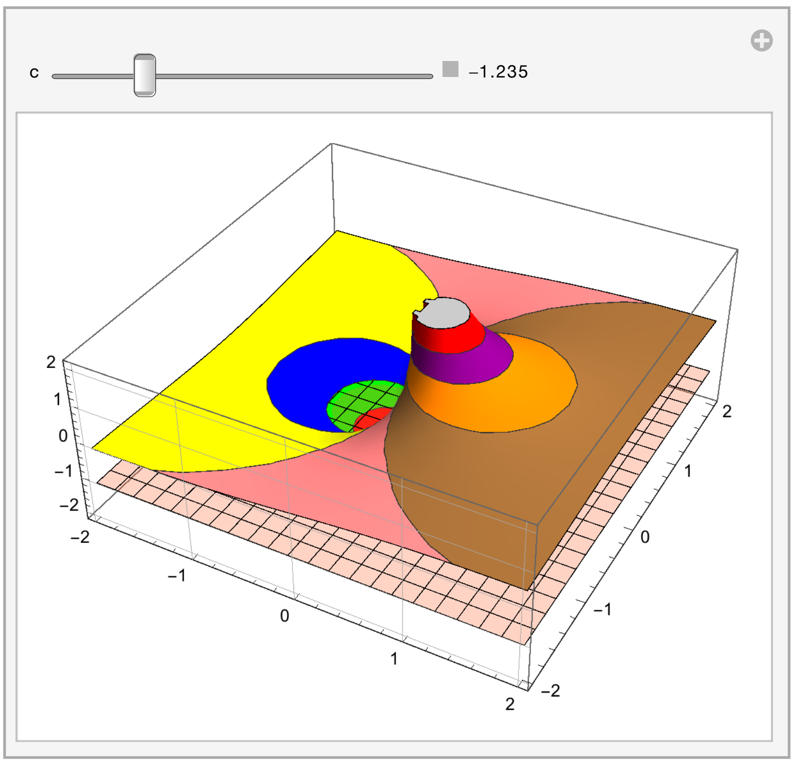
```
Manipulate[v1 = {1, 0, 0};
v2 = {0, 1, 0};
p0 = {0, 0, c};
plane1 = p0 + s * v1 + t * v2;
Show[Plot3D[{y/(y^2 + x^2)}, {x, -2, 2}, {y, -2, 2},
    Mesh → 8, MeshFunctions → {#3 &}, FaceGrids → {{0, -1, 0}},
    ColorFunction → Function[{x, y, z}, Hue[z]],
    ColorFunctionScaling → True, MeshShading →
     {Red, Green, Blue, Yellow, Pink, Brown, Orange, Purple}],
  Plot3D[{plane1}, {x, -2, 2}, {y, -2, 2},
    PlotStyle → Directive[RGBColor[1, 0.5, 0.35, 1], Opacity[0.3]],
    MeshStyle → Black, Mesh → 20]],
 {{c, -1.99}, -1.9, 2.1, Appearance → "Labeled"}]
```



6. $T = \dfrac{x}{(x^2 + y^2)}$

```
Clear["Global`*"]
```

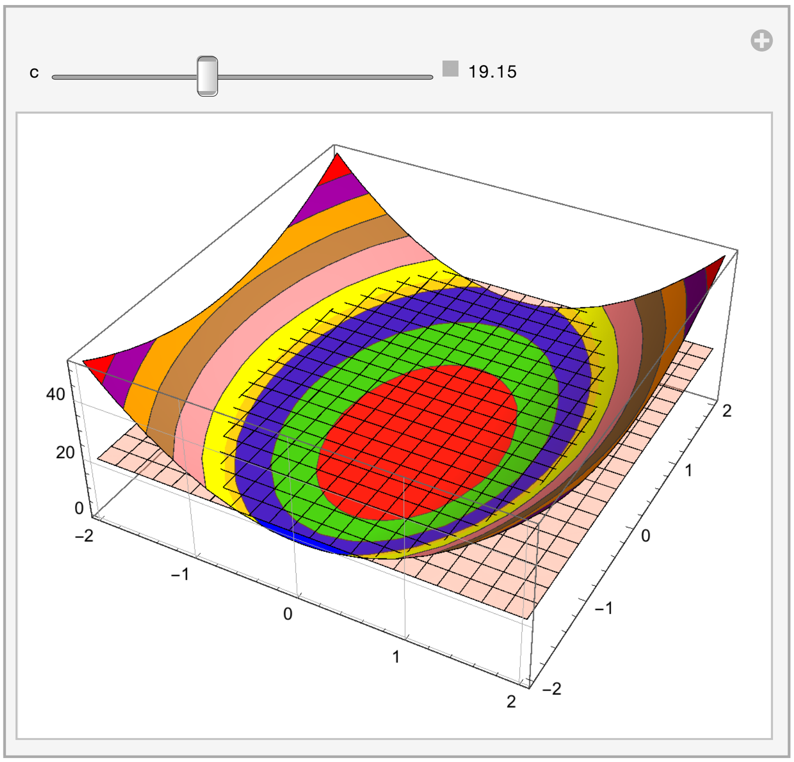
```
Manipulate[v1 = {1, 0, 0};
v2 = {0, 1, 0};
p0 = {0, 0, c};
plane1 = p0 + s * v1 + t * v2;
Show[Plot3D[{x/(y^2 + x^2)}, {x, -2, 2}, {y, -2, 2},
    Mesh → 8, MeshFunctions → {#3 &}, FaceGrids → {{0, -1, 0}},
    ColorFunction → Function[{x, y, z}, Hue[z]],
    ColorFunctionScaling → True, MeshShading →
     {Red, Green, Blue, Yellow, Pink, Brown, Orange, Purple}],
  Plot3D[{plane1}, {x, -2, 2}, {y, -2, 2},
    PlotStyle → Directive[RGBColor[1, 0.5, 0.35, 1], Opacity[0.3]],
    MeshStyle → Black, Mesh → 20]],
 {{c, -2.00}, -2.1, 2, Appearance → "Labeled"}]
```



7. $T = 9x^2 + 4y^2$

```
Clear["Global`*"]
```

```
Manipulate[v1 = {1, 0, 0};
v2 = {0, 1, 0};
p0 = {0, 0, c};
plane1 = p0 + s * v1 + t * v2;
Show[Plot3D[{9 x^2 + 4 y^2}, {x, -2, 2}, {y, -2, 2},
   Mesh → 8, MeshFunctions → {#3 &}, FaceGrids → {{0, -1, 0}},
   ColorFunction → Function[{x, y, z}, Hue[z]],
   ColorFunctionScaling → True, MeshShading →
    {Red, Green, Blue, Yellow, Pink, Brown, Orange, Purple}],
  Plot3D[{plane1}, {x, -2, 2}, {y, -2, 2},
   PlotStyle → Directive[RGBColor[1, 0.5, 0.35, 1], Opacity[0.3]],
   MeshStyle → Black, Mesh → 20]],
 {{c, -1.}, -1, 50, Appearance → "Labeled"}]
```

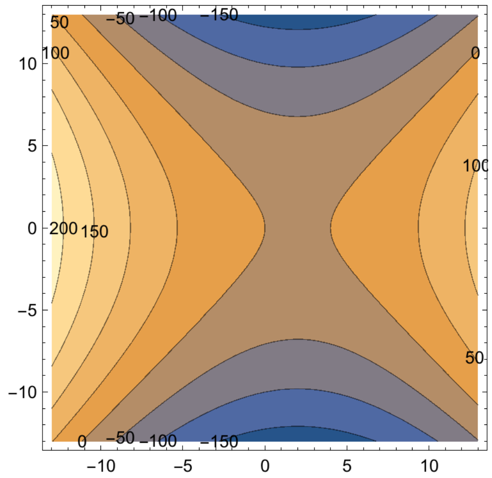


8. CAS project. Scalar fields in the plane. Sketch or graph isotherms of the following fields and describe what they look like.

(a) $x^2 - 4x - y^2$

```
Clear["Global`*"]
```

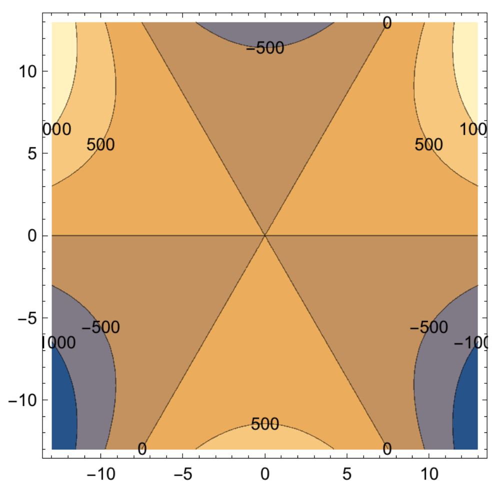
```
ContourPlot[x^2 - 4 x - y^2, {x, -13, 13},
 {y, -13, 13}, ContourLabels → True, ImageSize → 250]
```



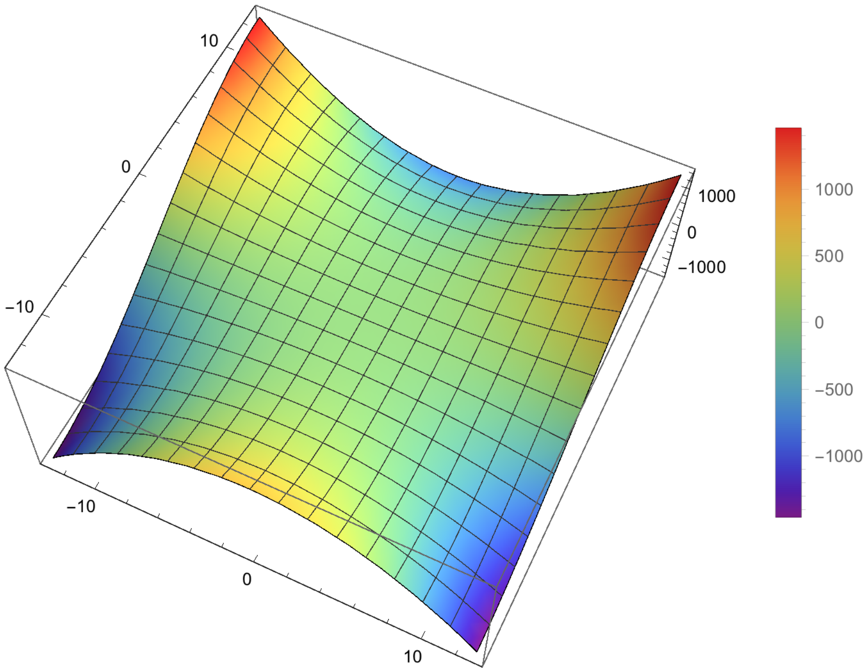
(b) $x^2 y - \frac{y^3}{3}$

```
Clear["Global`*"]
```

```
ContourPlot[x^2 y - y^3/3, {x, -13, 13},
 {y, -13, 13}, ContourLabels → True, ImageSize → 250]
```



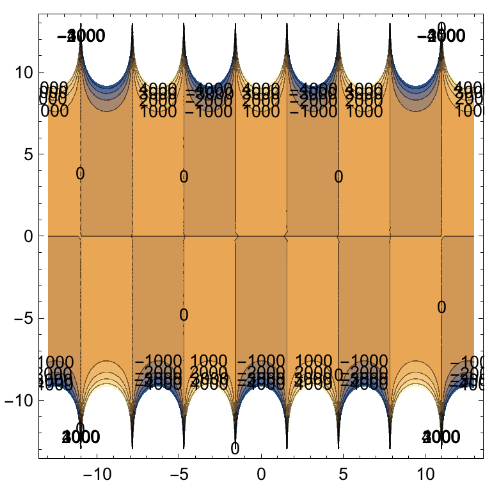
```
Plot3D[x^2 y - y^3/3, {x, -13, 13}, {y, -13, 13},
 ColorFunction → "Rainbow", PlotLegends -> Automatic]
```



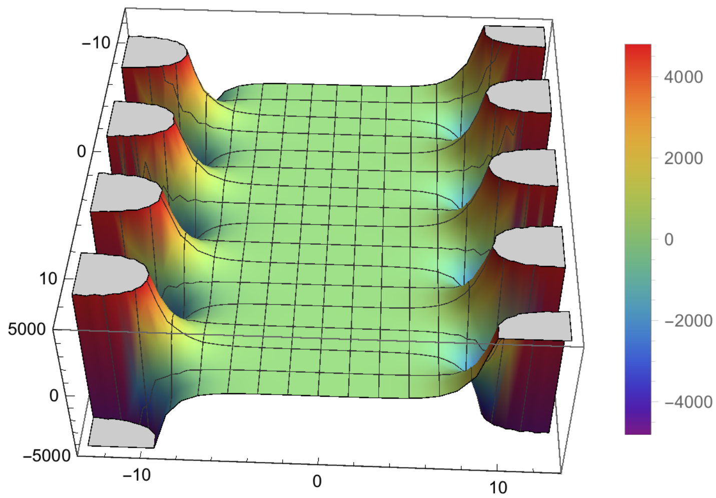
(c) Cos[x] Sinh[y]

```
Clear["Global`*"]
```

```
ContourPlot[Cos[x] Sinh[y], {x, -13, 13},
 {y, -13, 13}, ContourLabels → True, ImageSize → 250]
```



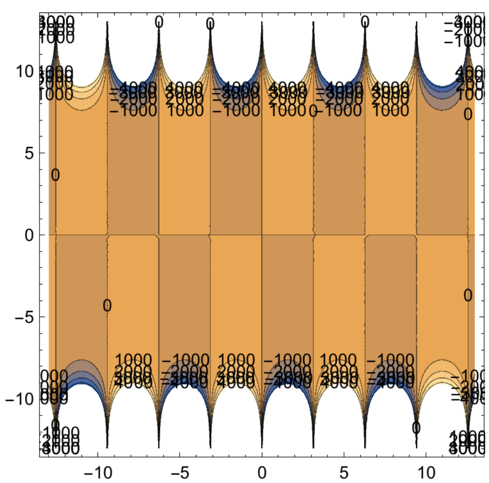
```
Plot3D[Cos[x] Sinh[y], {x, -13, 13}, {y, -13, 13},
 ColorFunction → "Rainbow", PlotLegends -> Automatic]
```



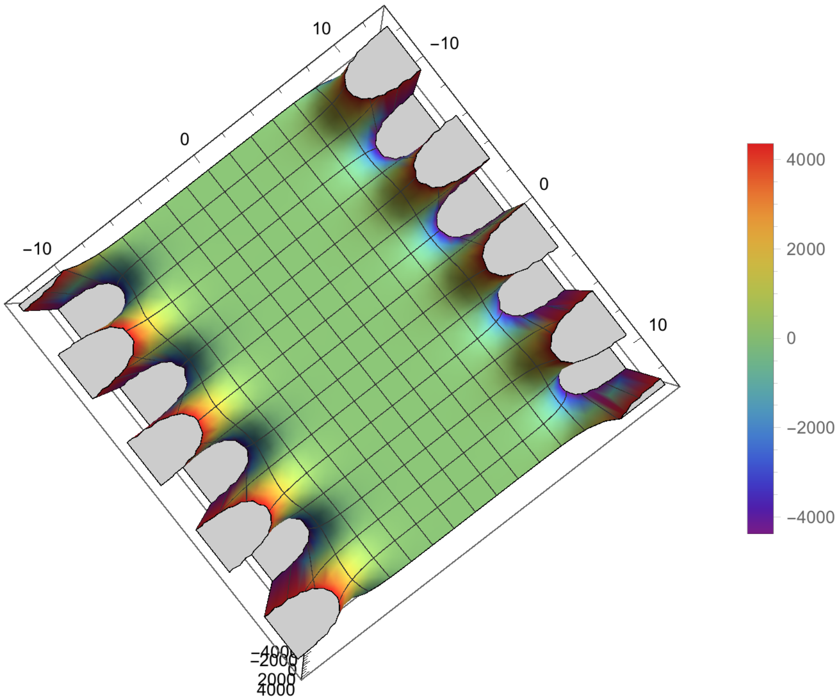
(d) Sin[x] Sinh[y]

```
Clear["Global`*"]
```

```
ContourPlot[Sin[x] Sinh[y], {x, -13, 13},
 {y, -13, 13}, ContourLabels → True, ImageSize → 250]
```

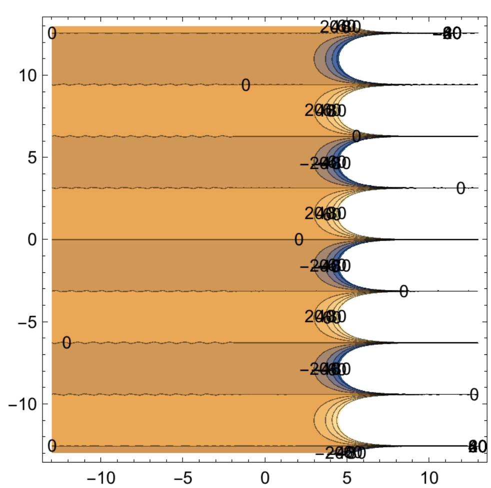


```
Plot3D[Sin[x] Sinh[y], {x, -13, 13}, {y, -13, 13},
 ColorFunction → "Rainbow", PlotLegends -> Automatic]
```

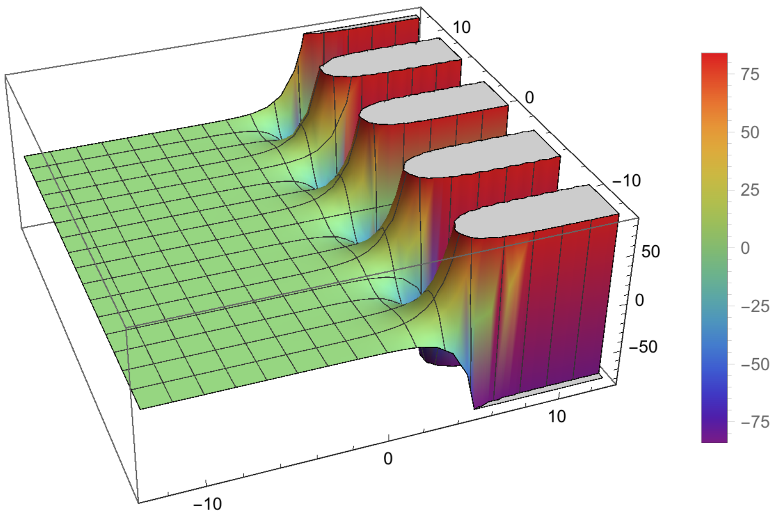


(e) e^x Sin[y]

```
ContourPlot[Exp[x] Sin[y], {x, -13, 13},
 {y, -13, 13}, ContourLabels → True, ImageSize → 250]
```



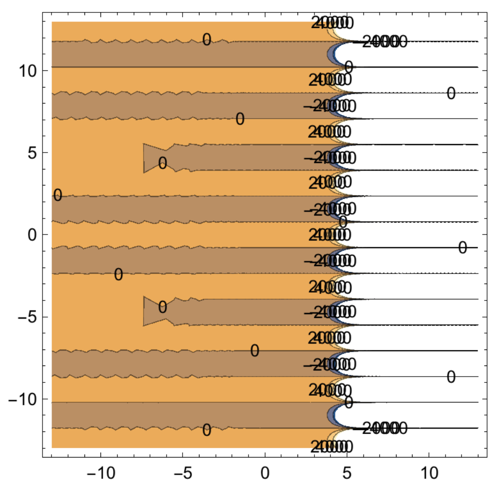
```
Plot3D[Exp[x] Sin[y], {x, -13, 13}, {y, -13, 13},
 ColorFunction → "Rainbow", PlotLegends -> Automatic]
```



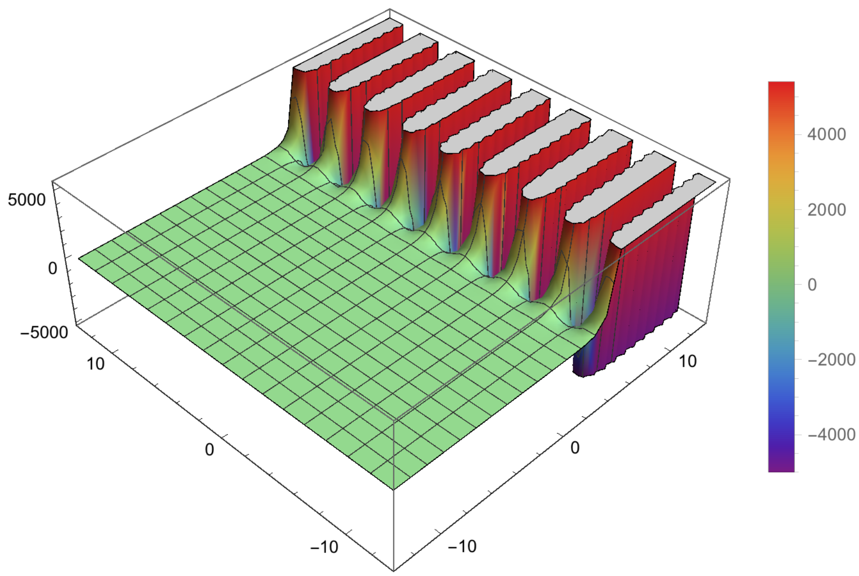

```
Clear["Global`*"]
```

(f) e^{2x} Cos[2 y]

```
ContourPlot[Exp[2 x] Cos[2 y], {x, -13, 13},
 {y, -13, 13}, ContourLabels → True, ImageSize → 250]
```



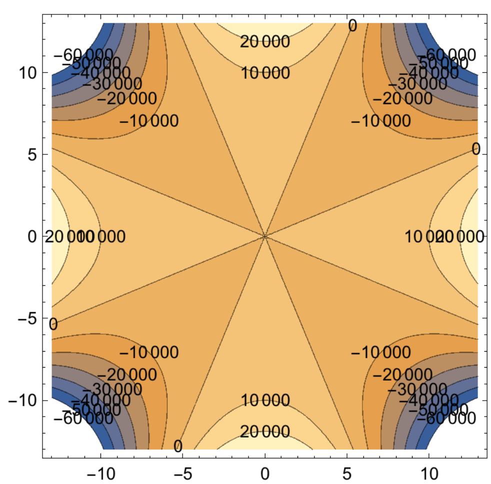
```
Plot3D[Exp[2 x] Cos[2 y], {x, -13, 13}, {y, -13, 13},
 ColorFunction → "Rainbow", PlotLegends -> Automatic]
```



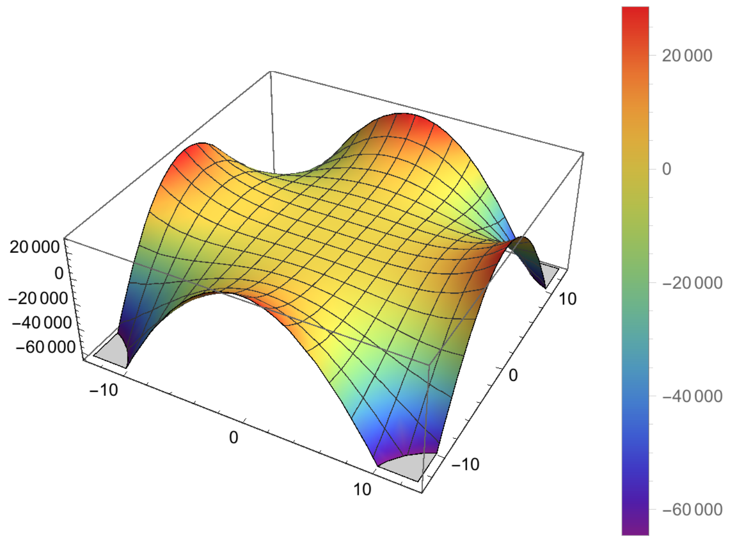
(g) $x^4 - 6x^2y^2 + y^4$

```
Clear["Global`*"]
```

```
ContourPlot[x^4 - 6 x^2 y^2 + y^4, {x, -13, 13},
 {y, -13, 13}, ContourLabels → True, ImageSize → 250]
```



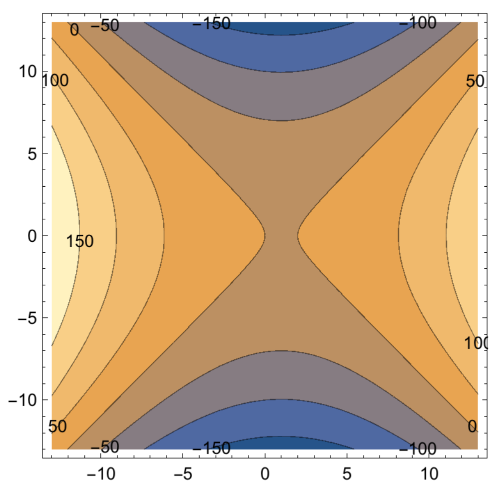
```
Plot3D[x^4 - 6 x^2 y^2 + y^4, {x, -13, 13}, {y, -13, 13},
 ColorFunction → "Rainbow", PlotLegends -> Automatic, ImageSize → 300]
```



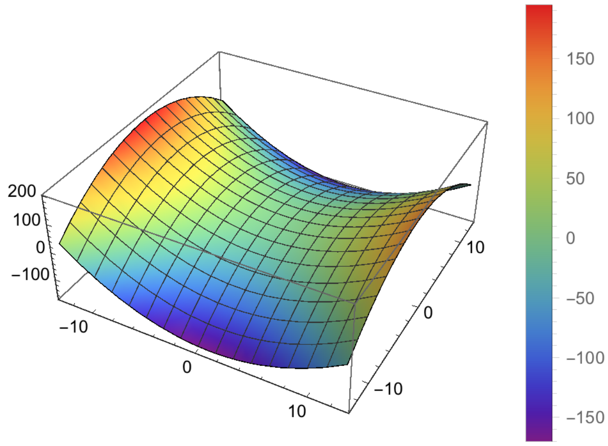
(h) $x^2 - 2x - y^2$

```
Clear["Global`*"]
```

```
ContourPlot[x^2 - 2 x - y^2, {x, -13, 13},
 {y, -13, 13}, ContourLabels → True, ImageSize → 250]
```



```
Plot3D[x^2 - 2 x - y^2, {x, -13, 13}, {y, -13, 13},
 ColorFunction → "Rainbow", PlotLegends -> Automatic, ImageSize → 250]
```



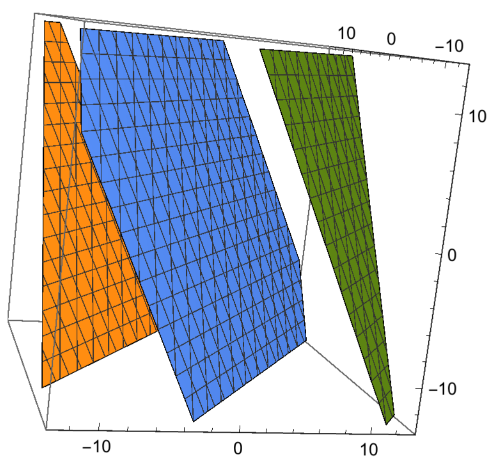
9 - 14 Scalar fields in space

What kind of surfaces are the level surfaces *f(x, y, z)* = const?

9. $f = 4x - 3y + 2z$

```
Clear["Global`*"]
```

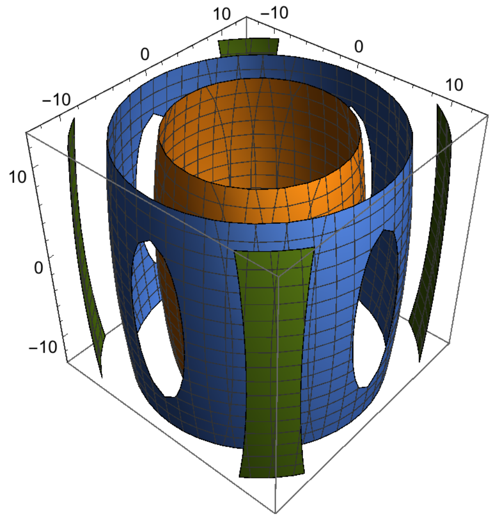
```
ContourPlot3D[4 x - 3 y + 2 z, {x, -13, 13},
 {y, -13, 13}, {z, -13, 13}, ImageSize → 250]
```



10. $f = 9(x^2 + y^2) + z^2$

```
Clear["Global`*"]
```

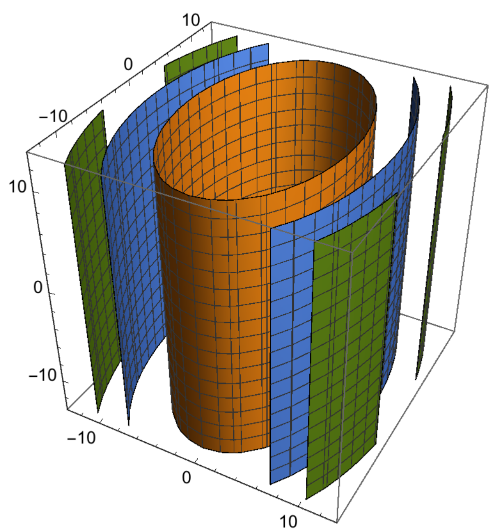
```
ContourPlot3D[9 (x^2 + y^2) + z^2, {x, -13, 13},
 {y, -13, 13}, {z, -13, 13}, ImageSize -> 250]
```



11. $f = 5x^2 + 2y^2$

```
Clear["Global`*"]
```

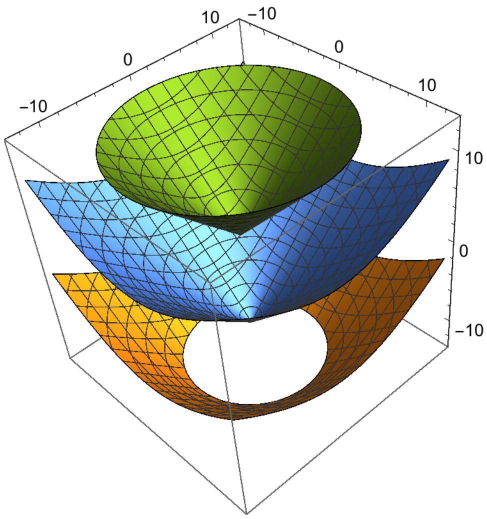
```
ContourPlot3D[5 x^2 + 2 y^2, {x, -13, 13},
 {y, -13, 13}, {z, -13, 13}, ImageSize -> 250]
```



12. $f = z - \sqrt{x^2 + y^2}$

```
Clear["Global`*"]
```

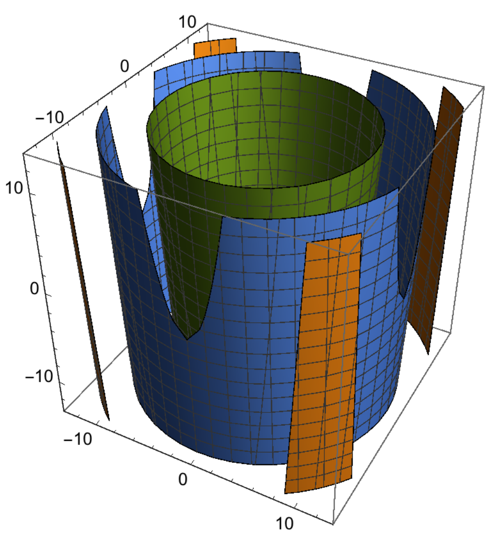
```
ContourPlot3D[z - Sqrt[x^2 + y^2], {x, -13, 13},
 {y, -13, 13}, {z, -13, 13}, ImageSize → 250]
```



13. $f = z - (x^2 + y^2)$

```
Clear["Global`*"]
```

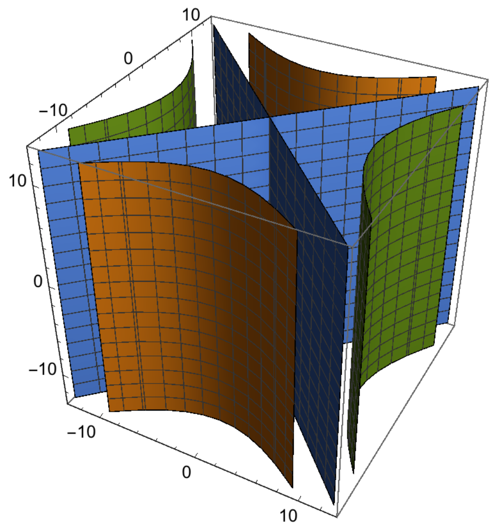
```
ContourPlot3D[z - (x^2 + y^2), {x, -13, 13},
 {y, -13, 13}, {z, -13, 13}, ImageSize → 250]
```



14. $f = x - y^2$

```
Clear["Global`*"]
```

```
ContourPlot3D[x^2 - y^2, {x, -13, 13},
 {y, -13, 13}, {z, -13, 13}, ImageSize → 250]
```



15 - 20 Vector fields

Sketch figures similar to figure 198, p. 378. Try to interpret the field of **v** as a velocity field.

15. v = i + j

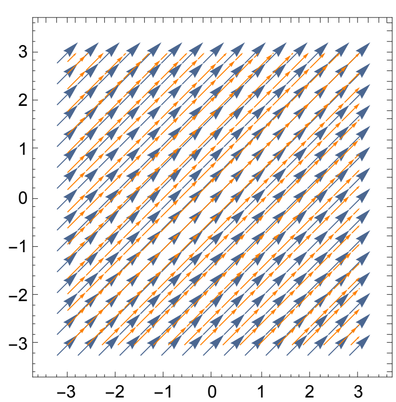
```
Clear["Global`*"]
```

Let me see if I can use the verbiage in problem 19 (which was worked first) to find the right expression for this function.

$$\mathbf{v} = \mathbf{i} + \mathbf{j} = 1[1, 0, 0] + 1[0, 1, 0] + 0[0, 0, 1] = [1, 1, 0]$$

Then to apply the recipe to an input vector. Here z and **k** are ignored, because according to the problem expression, these contribute nothing to the final output. Also, no x or y factors are visible in the problem expression. So again I am operating on an input vector of the form $[x, y, 0]$. What comes out of the function machine is $[x + 1, y + 1, 0]$

```
StreamPlot[{1, 1}, {x, -3, 3}, {y, -3, 3},
 VectorPoints → Automatic, StreamStyle → Orange, ImageSize → 200]
```



17. v = xj

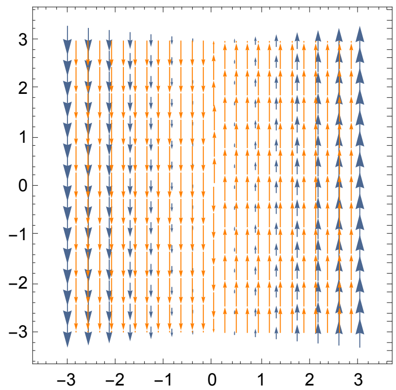
```
Clear["Global`*"]
```

Let me see if I can again use the verbiage in problem 19 to find the right expression for this function.

$$\mathbf{v} = x\mathbf{j} = 0[1, 0, 0] + x[0, 1, 0] + 0[0, 0, 1] = [0, x, 0]$$

Then to apply the recipe to an input vector. Here z and $\mathbf{k}$ are ignored, because according to the problem expression, these contribute nothing to the final output. However, x is visible in the expression and must be used, I believe. So again I am operating on an input vector of the form $[x, y, 0]$. What comes out of the function machine is $[x, x + y, 0]$

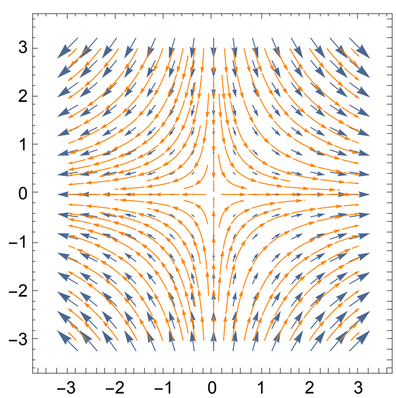
```
StreamPlot[{0, x}, {x, -3, 3}, {y, -3, 3},
 VectorPoints → Automatic, StreamStyle → Orange, ImageSize → 200]
```



19. v = xi - yj

```
Clear["Global`*"]
```

```
StreamPlot[{x, - y}, {x, -3, 3}, {y, -3, 3},
 VectorPoints → Automatic, StreamStyle → Orange, ImageSize → 200]
```



This one is in the s.m.. Maybe I can summarize. First is written the function definition in terms of **i, j, k**. Thus, $\mathbf{v} = \mathbf{v}(x, y, z) = x\mathbf{i} - y\mathbf{j} + 0\,\mathbf{k}$. This makes sense, with regard to the third coordinate. There is no mention of **k**, so it must be treated as zero. Then the function is expanded,

$\mathbf{v} = x\mathbf{i} - y\mathbf{j} = x[1, 0, 0] - y[0, 1, 0] = [x, 0, 0] - [0, y, 0] = [x, -y, 0]$,

ending up with a recipe for the function's operation on an operand *vector*. The s.m. describes what it means for a vector function to operate. A starting vector is transformed into the output vector by the function recipe. In this case, if I take a sample vector $[x, y, 0]$ and feed it to the function, it ***adds to it.*** In effect, I get $[x, y, 0] + [x, -y, 0] = [2x, 0, 0]$.

The thing that comes out is not how the function is shown in printed form, however. It is not what the plot command sees. What the plot command sees is the expression at the end of the sequence of equalities above.

21. CAS project. Vector fields. Plot by arrows:

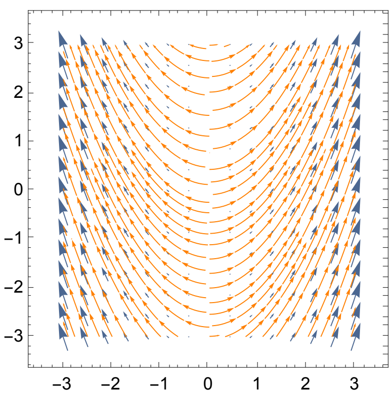
(a) $v = \{x, x^2\}$ (b) $v = \{\frac{1}{y}, \frac{1}{x}\}$

(c) $v = \{\text{Cos}[x], \text{Sin}[x]\}$ (d) $v = e^{-(x^2+y^2)}\{x, -y\}$

```
Clear["Global`*"]
```

For (a):

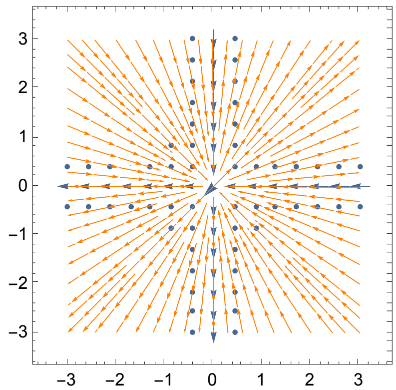
```
StreamPlot[{x, x^2}, {x, -3, 3}, {y, -3, 3},
 VectorPoints → Automatic, StreamStyle → Orange, ImageSize → 200]
```



For (b):

```
Clear["Global`*"]
```

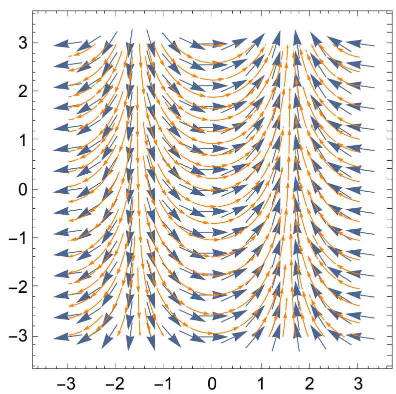
```
StreamPlot[{1/y, 1/x}, {x, -3, 3}, {y, -3, 3},
 VectorPoints → Automatic, StreamStyle → Orange, ImageSize → 200]
```



For (c):

```
Clear["Global`*"]
```

```
StreamPlot[{Cos[x], Sin[x]}, {x, -3, 3}, {y, -3, 3},
 VectorPoints → Automatic, StreamStyle → Orange, ImageSize → 200]
```



For (d):

```
Clear["Global`*"]
```

```
StreamPlot[{Exp[-(x^2 + y^2)] x, Exp[-(x^2 + y^2)]  - y}, {x, -3, 3}, {y, -3, 3},
 VectorPoints → Automatic, StreamStyle → Orange, ImageSize → 200]
```