

I like to know where my files reside:

```
In[31]= NotebookDirectory[]
```

```
Out[31]= C:\Dropbox\Work\myweb\Courses\Math_pages\Math_430\
```

```
In[46]= NotebookFileName[]
```

```
Out[46]= C:\Dropbox\Work\myweb\Courses\Math_pages\Math_430\PiecewiseFunctions3D.nb
```

Piecewise defined functions in Mathematica

General comments

All operating system related commands, pertain to Windows operating system.

This notebook is partly written by ChatGPT. I made some changes, adding things that I like. I will post the original ChatGPT written notebook separately.

Remember, the input cells below, recognized by **Consolas** font used in them.

If you need any help with Mathematica try the menu item Help, or google what you need mentioning Mathematica in your search query, or ask ChatGPT, or post a question in our class Discussions on Canvas.

How to use a piecewise defined function in Mathematica?

How to use a piecewise defined function in Mathematica?

Piecewise-defined functions are functions that have different definitions for different parts of their domain. In Mathematica, the `Piecewise` function is used to define such functions. (*Defining a Piecewise Function*) To define a piecewise function in Mathematica, use the `Piecewise` function. The general syntax is: `Piecewise[{{expr1,cond1},{expr2,cond2},...}]`

For example, the function:

$$f(x)=x^2 \text{ for } x < 0 \text{ and } x \text{ for } x \geq 0$$

is defined in Mathematica as:

```
In[32]= ff[x_] := Piecewise[{{x^2, x < 0}, {x, x ≥ 0}}]
```

Here I want to point out that I like using “strange” names for mathematical objects. In Mathematica, I

never call a function **f**, I always call it something like **ff**, like an abbreviation for “function f”.

To test this definition in Mathematica I evaluate

```
In[33]:= ff[3]
```

```
Out[33]= 3
```

```
In[34]:= ff[-2]
```

```
Out[34]= 4
```

Or even more values

```
In[35]:= Table[ff[x], {x, -4, 4, 1/4}]
```

```
Out[35]= {16,  $\frac{225}{16}$ ,  $\frac{49}{4}$ ,  $\frac{169}{16}$ , 9,  $\frac{121}{16}$ ,  $\frac{25}{4}$ ,  $\frac{81}{16}$ , 4,  $\frac{49}{16}$ ,  $\frac{9}{4}$ ,  $\frac{25}{16}$ , 1,  $\frac{9}{16}$ ,  $\frac{1}{4}$ ,  $\frac{1}{16}$ , 0,  $\frac{1}{4}$ ,  $\frac{1}{2}$ ,  $\frac{3}{4}$ , 1,  $\frac{5}{4}$ ,  $\frac{3}{2}$ ,  $\frac{7}{4}$ , 2,  $\frac{9}{4}$ ,  $\frac{5}{2}$ ,  $\frac{11}{4}$ , 3,  $\frac{13}{4}$ ,  $\frac{7}{2}$ ,  $\frac{15}{4}$ , 4}
```

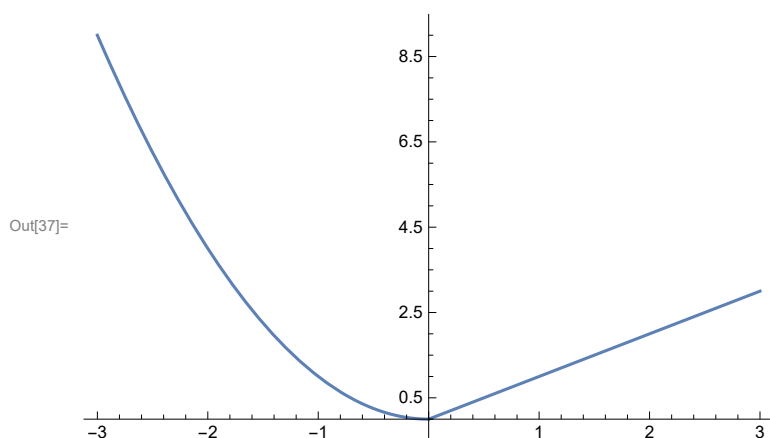
or, the function represented as pairs $\{x, f(x)\}$, which is sometimes useful

```
In[36]:= Table[{x, ff[x]}, {x, -4, 4, 1/4}]
```

```
Out[36]= {{-4, 16}, {- $\frac{15}{4}$ ,  $\frac{225}{16}$ }, {- $\frac{7}{2}$ ,  $\frac{49}{4}$ }, {- $\frac{13}{4}$ ,  $\frac{169}{16}$ }, {-3, 9}, {- $\frac{11}{4}$ ,  $\frac{121}{16}$ }, {- $\frac{5}{2}$ ,  $\frac{25}{4}$ }, {- $\frac{9}{4}$ ,  $\frac{81}{16}$ }, {-2, 4}, {- $\frac{7}{4}$ ,  $\frac{49}{16}$ }, {- $\frac{3}{2}$ ,  $\frac{9}{4}$ }, {- $\frac{5}{4}$ ,  $\frac{25}{16}$ }, {-1, 1}, {- $\frac{3}{4}$ ,  $\frac{9}{16}$ }, {- $\frac{1}{2}$ ,  $\frac{1}{4}$ }, {- $\frac{1}{4}$ ,  $\frac{1}{16}$ }, {0, 0}, { $\frac{1}{4}$ ,  $\frac{1}{16}$ }, { $\frac{1}{2}$ ,  $\frac{1}{4}$ }, { $\frac{3}{4}$ ,  $\frac{9}{16}$ }, {1, 1}, { $\frac{5}{4}$ ,  $\frac{25}{16}$ }, { $\frac{3}{2}$ ,  $\frac{9}{4}$ }, { $\frac{7}{4}$ ,  $\frac{49}{16}$ }, {2, 4}, { $\frac{9}{4}$ ,  $\frac{81}{16}$ }, { $\frac{5}{2}$ ,  $\frac{25}{4}$ }, { $\frac{11}{4}$ ,  $\frac{121}{16}$ }, {3, 9}, { $\frac{13}{4}$ ,  $\frac{169}{16}$ }, { $\frac{7}{2}$ ,  $\frac{49}{4}$ }, { $\frac{15}{4}$ ,  $\frac{225}{16}$ }, {4, 4}}
```

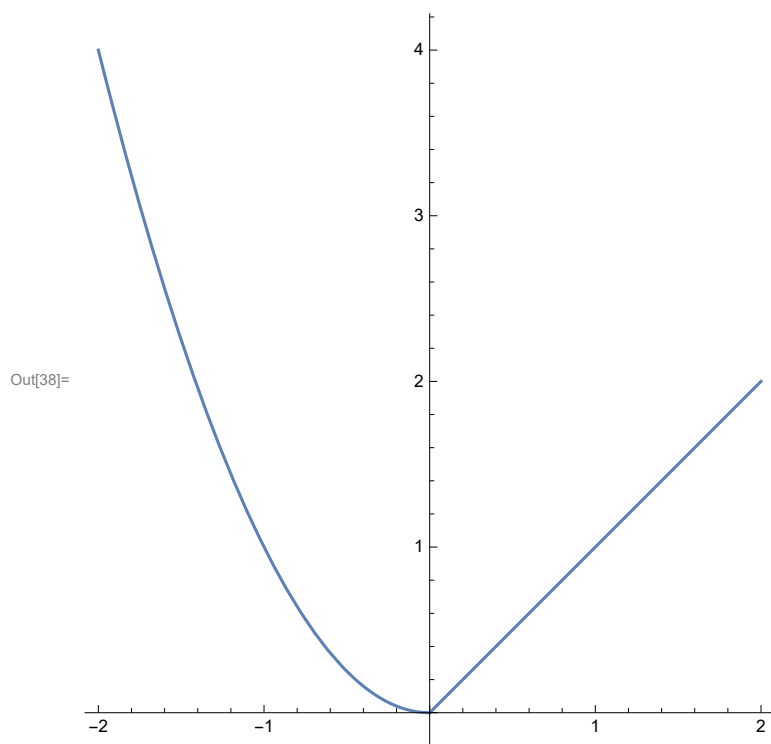
And plotting the function

```
In[37]:= Plot[ff[x], {x, -3, 3}]
```



Sometimes I like to control the **AspectRatio** in a plot. **AspectRatio->Automatic** makes the horizontal units identical in length to the vertical units. To get the arrow as in the code below, one just types “->”

In[38]:= `Plot[ff[x], {x, -2, 2}, AspectRatio -> Automatic]`



Functions of two variables

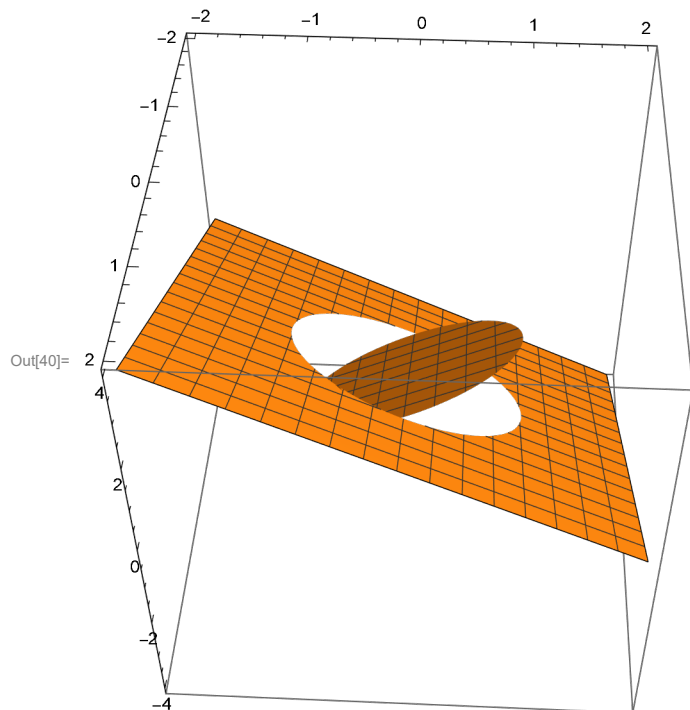
For functions of two variables, the process is similar. Consider the function:

$$g(x,y) = x+y \text{ for } x^2+y^2 < 1 \text{ and } x-y \text{ for } x^2+y^2 \geq 1$$

This can be defined in Mathematica as:

In[39]:= `fg[x_, y_] := Piecewise[{{x + y, x^2 + y^2 < 1}, {x - y, x^2 + y^2 >= 1}}]`

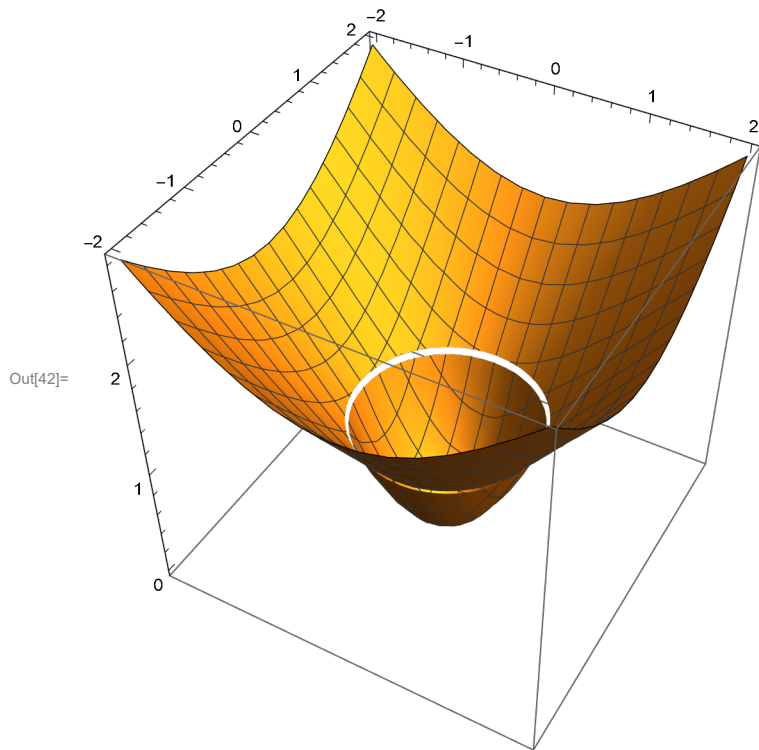
```
In[40]:= Plot3D[fg[x, y], {x, -2, 2}, {y, -2, 2},
  Exclusions -> And[x^2 + y^2 == 1], BoxRatios -> {1, 1, 1}]
```



Ok, the above example was on a more complicated side, needing the **Exclusions** option

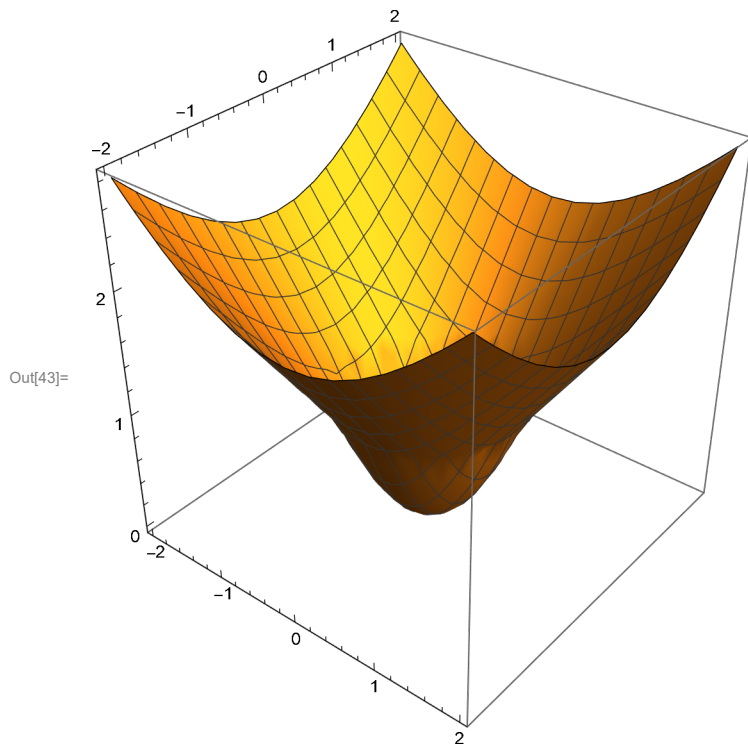
```
In[41]:= fh[x_, y_] := Piecewise[{{(x^2 + y^2), x^2 + y^2 < 1}, {sqrt[x^2 + y^2], x^2 + y^2 >= 1}}]
```

```
In[42]:= Plot3D[fh[x, y], {x, -2, 2}, {y, -2, 2}, BoxRatios -> {1, 1, 1}]
```



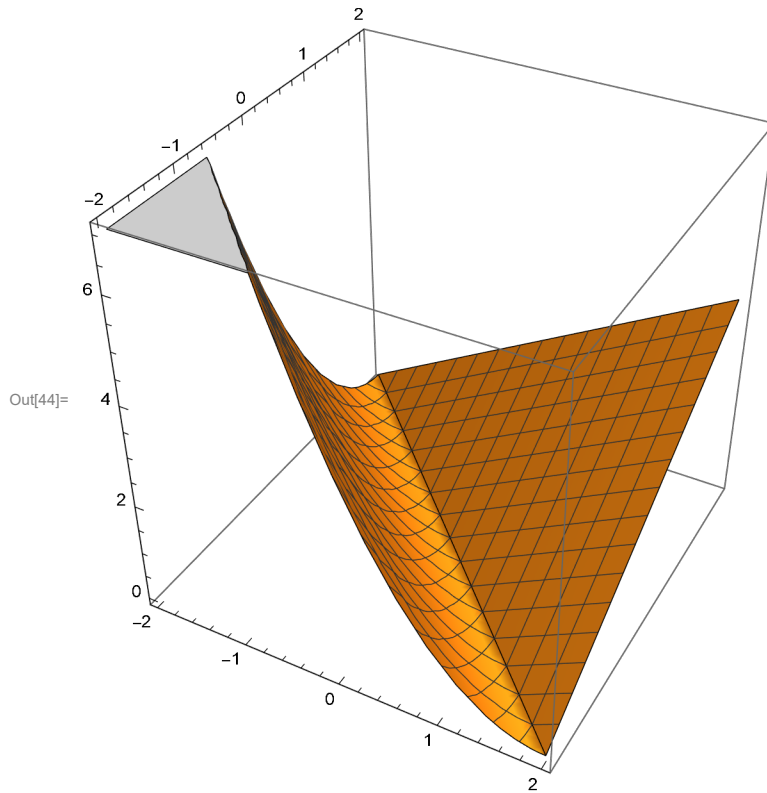
Functions like the one above need the **Exclusions** option as below

```
In[43]:= Plot3D[fh[x, y], {x, -2, 2}, {y, -2, 2}, Exclusions -> None, BoxRatios -> {1, 1, 1}]
```



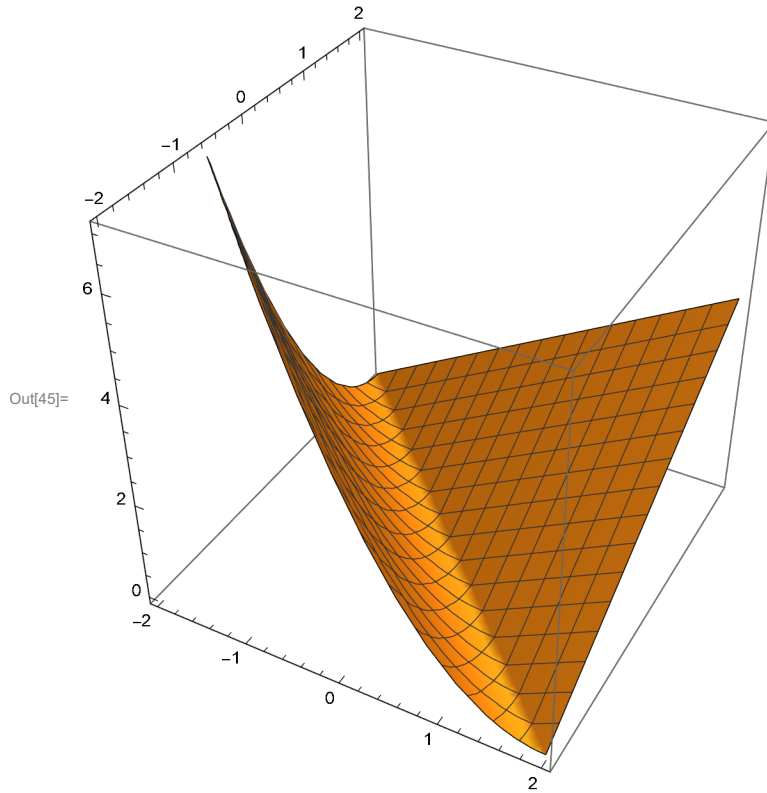
Now we can combine the function **ff** with two variables as follows

```
In[44]:= Plot3D[ff[x + y], {x, -2, 2}, {y, -2, 2}, Exclusions -> None, BoxRatios -> {1, 1, 1}]
```



Sometimes I do not like how Mathematica clips three-dimensional plot. So, I have to use **Plot3D** options to get plots how I like them. In this case it is **ClippingStyle -> None**.

```
In[45]:= Plot3D[ff[x + y], {x, -2, 2}, {y, -2, 2},  
Exclusions -> None, ClippingStyle -> None, BoxRatios -> {1, 1, 1}]
```



Comment

An alternative way to define a piecewise function is to use

```
If[x < 0, x^2, x] (* the logic of this comand is, given a condition,  
if it is True, then the second component evaluates, if the condition is false,  
then the third component evaluates; compare to the definition of ff above *)
```