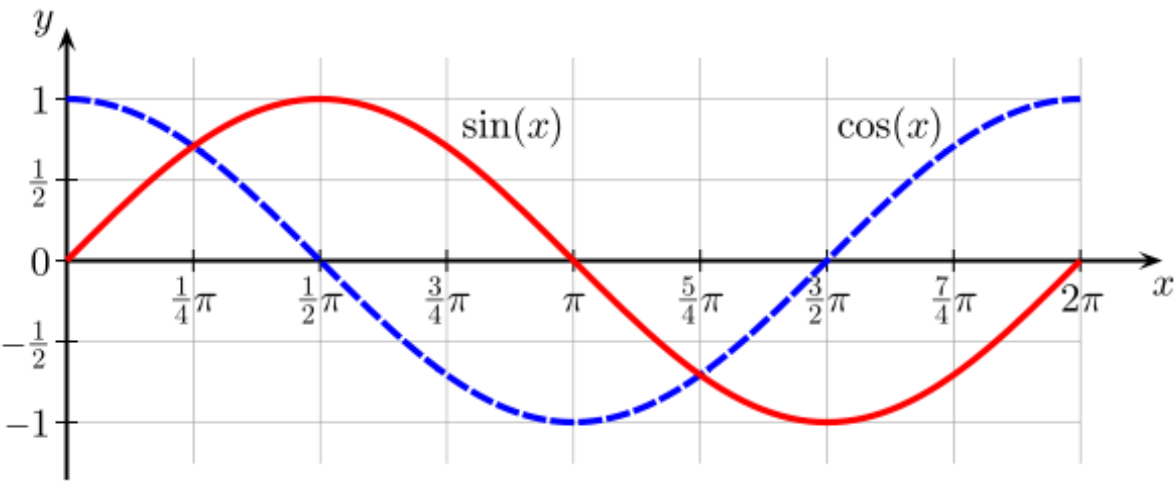


```
using Plots
gr()
default(fmt = :png)
```

```
[2]: using DataFrames
```

Using Plots.jl

Plots.jl outputs plots in different formats. It is written in Julia:



```
f(x) = sin(x)
g(x) = cos(x)
h(x) = tan(x)
```

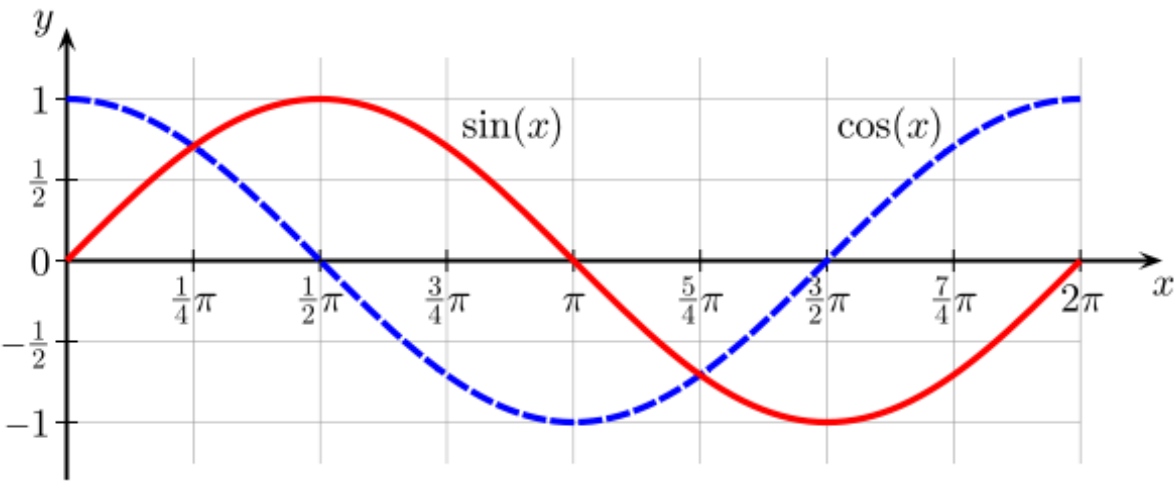
```
[21]:
```

```
h (generic function with 1 method)
```

```
[22]: xs = LinRange(0, 2pi, 100)
```

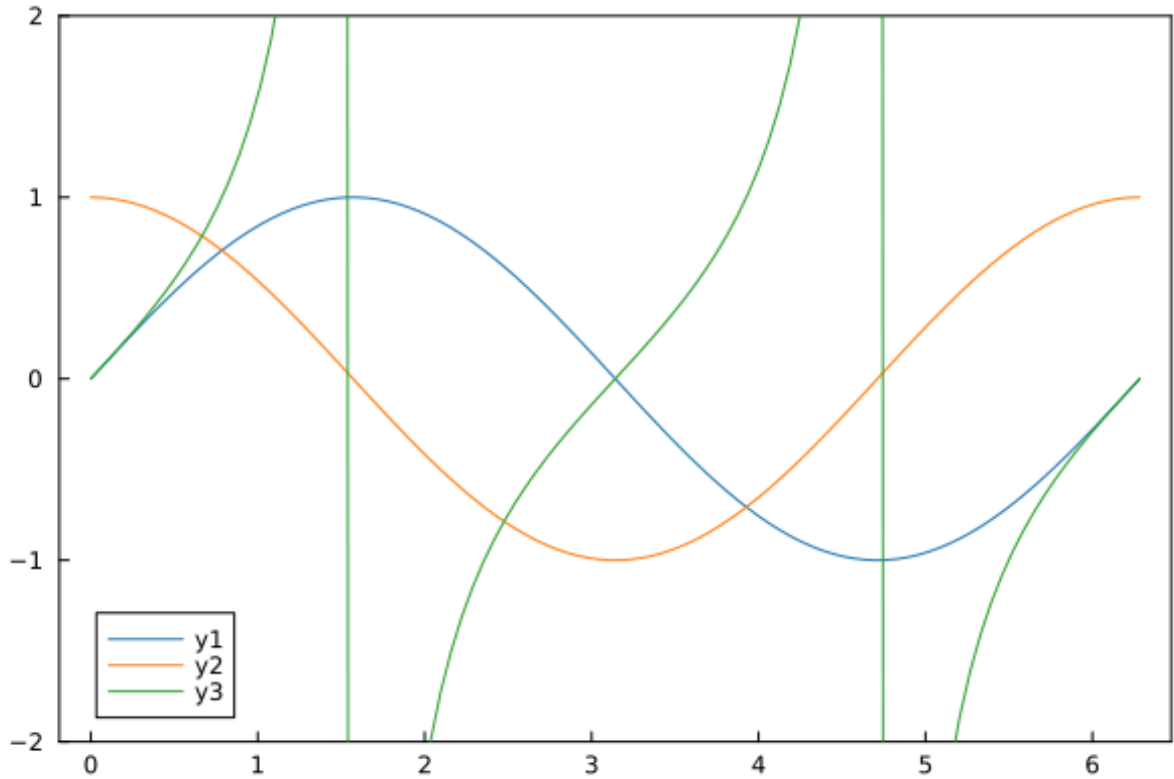
```
100-element LinRange{Float64, Int64}:
 0.0, 0.0634665, 0.126933, 0.1904, ..., 6.09279, 6.15625, 6.21972, 6.28319
```

These are the trigonometric functions,  $\sin(x)$   $\cos(x)$   $\tan(x)$  According to Wikipedia, their graphs look like this:



```
plot(xs, [f, g, h]; ylim = (-2, 2), framestyle = :box, grid = false, palette
= :tab10)
```

```
[23]:
```



Let's produce an error:

```
[24]: i(x)
```

```
UndefVarError: `i` not defined

Stacktrace:
 [1] top-level scope
      @ In[24]:1
```

Rich Outputs

We can try some table outputs, for example:

```
[3]: df = DataFrame((col1 = ["First", "Second", "Third"], col2 = [1, 2, 3]))
```

3×2 DataFrame		
Row	col1 String	col2 Int64
1	First	1
2	Second	2
3	Third	3