trig-graphs

December 7, 2023

[3]:
```
!pip install numpy matplotlib
```

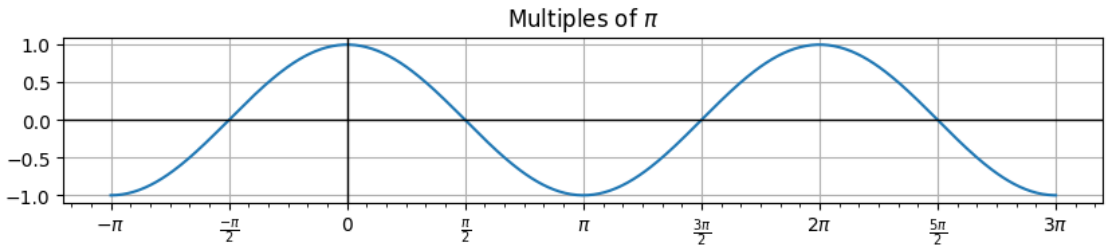
```
Collecting numpy
  Downloading numpy-1.26.2-cp310-cp310-macosx_11_0_arm64.whl (14.0 MB)
                             14.0/14.0 MB
79.5 MB/s eta 0:00:00a 0:00:01
Collecting matplotlib
  Downloading matplotlib-3.8.2-cp310-cp310-macosx_11_0_arm64.whl (7.5 MB)
                             7.5/7.5 MB
77.8 MB/s eta 0:00:00a 0:00:01
Collecting cycler>=0.10
  Downloading cycler-0.12.1-py3-none-any.whl (8.3 kB)
Collecting fonttools>=4.22.0
  Downloading fonttools-4.46.0-cp310-cp310-macosx_10_9_universal2.whl (2.8 MB)
                             2.8/2.8 MB
55.1 MB/s eta 0:00:0000:01
Requirement already satisfied: packaging>=20.0 in
/opt/homebrew/Caskroom/miniconda/base/lib/python3.10/site-packages (from
matplotlib) (23.2)
Collecting pillow>=8
  Downloading Pillow-10.1.0-cp310-cp310-macosx_11_0_arm64.whl (3.3 MB)
                             3.3/3.3 MB
63.9 MB/s eta 0:00:0000:01
Collecting kiwisolver>=1.3.1
  Downloading kiwisolver-1.4.5-cp310-cp310-macosx_11_0_arm64.whl (66 kB)
                             66.2/66.2 kB
8.5 MB/s eta 0:00:00
Collecting contourpy>=1.0.1
  Downloading contourpy-1.2.0-cp310-cp310-macosx_11_0_arm64.whl (242 kB)
                            242.2/242.2 kB
28.3 MB/s eta 0:00:00
Collecting pyparsing>=2.3.1
  Downloading pyparsing-3.1.1-py3-none-any.whl (103 kB)
                             103.1/103.1 kB
11.1 MB/s eta 0:00:00
Requirement already satisfied: python-dateutil>=2.7 in
/opt/homebrew/Caskroom/miniconda/base/lib/python3.10/site-packages (from
matplotlib) (2.8.2)
```

```
Requirement already satisfied: six>=1.5 in
/opt/homebrew/Caskroom/miniconda/base/lib/python3.10/site-packages (from python-
dateutil>=2.7->matplotlib) (1.16.0)
Installing collected packages: pyparsing, pillow, numpy, kiwisolver, fonttools,
cycler, contourpy, matplotlib
Successfully installed contourpy-1.2.0 cycler-0.12.1 fonttools-4.46.0
kiwisolver-1.4.5 matplotlib-3.8.2 numpy-1.26.2 pillow-10.1.0 pyparsing-3.1.1
```

[4]:
```python
import numpy as np
import matplotlib.pyplot as plt

def multiple_formatter(denominator=2, number=np.pi, latex='\pi'):
    def gcd(a, b):
        while b:
            a, b = b, a%b
        return a
    def _multiple_formatter(x, pos):
        den = denominator
        num = np.int32(np.rint(den*x/number))
        com = gcd(num,den)
        (num,den) = (int(num/com),int(den/com))
        if den==1:
            if num==0:
                return r'$0$'
            if num==1:
                return r'$%s$'%latex
            elif num==-1:
                return r'$-%s$'%latex
            else:
                return r'$%s%s$'%(num,latex)
        else:
            if num==1:
                return r'$\frac{%s}{%s}$'%(latex,den)
            elif num==-1:
                return r'$\frac{-%s}{%s}$'%(latex,den)
            else:
                return r'$\frac{%s%s}{%s}$'%(num,latex,den)
    return _multiple_formatter

class Multiple:
    def __init__(self, denominator=2, number=np.pi, latex='\pi'):
        self.denominator = denominator
        self.number = number
        self.latex = latex
    def locator(self):
        return plt.MultipleLocator(self.number / self.denominator)
    def formatter(self):
```

```
        return plt.FuncFormatter(multiple_formatter(self.denominator, self.
↪number, self.latex))
```

[5]:
```
x = np.linspace(-np.pi, 3*np.pi,500)
plt.figure(figsize=(10,6))

plt.plot(x, np.cos(x))
plt.title(r'Multiples of $\pi$')
ax = plt.gca()
ax.grid(True)
ax.set_aspect(1.0)
ax.axhline(0, color='black', lw=1)
ax.axvline(0, color='black', lw=1)
ax.xaxis.set_major_locator(plt.MultipleLocator(np.pi / 2))
ax.xaxis.set_minor_locator(plt.MultipleLocator(np.pi / 12))
ax.yaxis.set_major_locator(plt.MultipleLocator(0.5))
ax.xaxis.set_major_formatter(plt.FuncFormatter(multiple_formatter()))
plt.show()
```

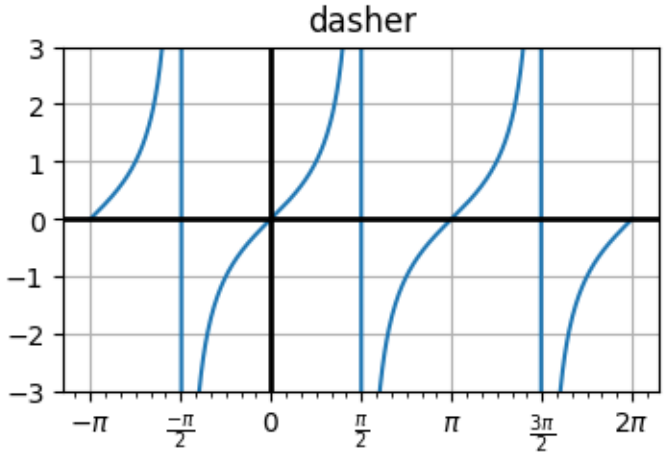


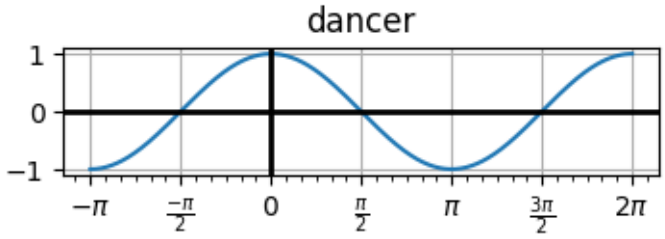
[9]:
```
def trig_plot(f, xmin=-np.pi, xmax=3*np.pi, ymin=-1, ymax=1, title="",␣
↪show_minor_x = False, y_minor=1):
    x = np.linspace(xmin, xmax,500)
    vf = np.vectorize(f)
    plt.figure(figsize=(4,3))
    plt.ylim(top=ymax, bottom=ymin)
    plt.plot(x, vf(x))
    plt.title(title)
    ax = plt.gca()
    ax.grid(True)
    if show_minor_x:
        plt.grid(axis='x', which='both', visible=True)
    else:
        plt.grid(axis='x', which='major', visible=True)
    ax.set_aspect(1.0)
    ax.axhline(0, color='black', lw=2)
    ax.axvline(0, color='black', lw=2)
```

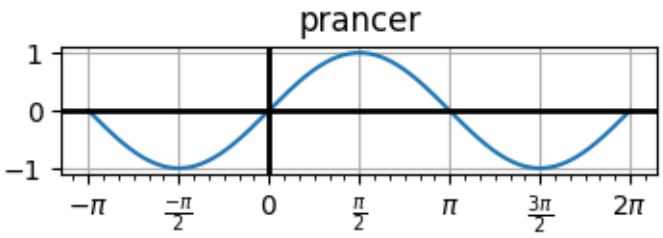
```
    ax.xaxis.set_major_locator(plt.MultipleLocator(np.pi / 2))
    ax.xaxis.set_minor_locator(plt.MultipleLocator(np.pi / 12))
    ax.yaxis.set_major_locator(plt.MultipleLocator(y_minor))
    ax.xaxis.set_major_formatter(plt.FuncFormatter(multiple_formatter()))
    plt.savefig(title+".png",dpi=300,pad_inches=0, bbox_inches='tight' )
```

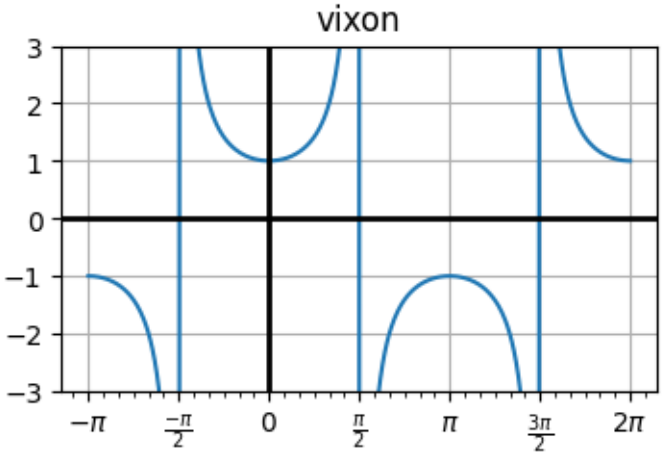
[10]:
```
from math import sin, cos, tan
sec = lambda x:1/cos(x)
csc = lambda x:1/sin(x)
cot = lambda x:1/tan(x)
```

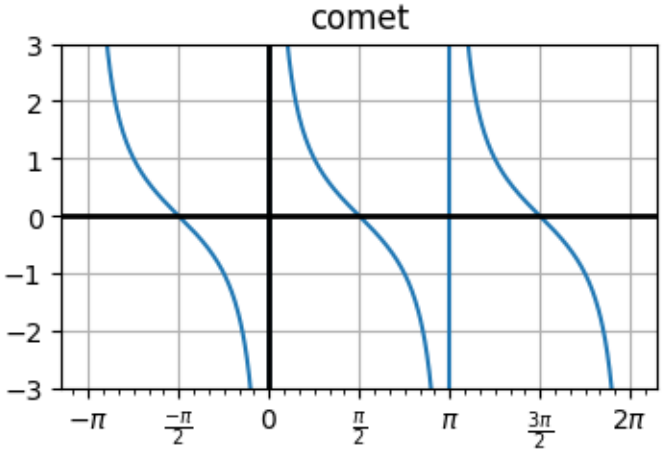
[11]:
```
trig_plot(tan,ymin=-3, ymax=3, xmin=-np.pi, xmax=2*np.pi, title="dasher")
trig_plot(cos,ymin=-1.1, ymax=1.1, xmin=-np.pi, xmax=2*np.pi, title="dancer")
trig_plot(sin,ymin=-1.1, ymax=1.1,  xmin=-np.pi, xmax=2*np.pi, title="prancer")
trig_plot(sec,ymin=-3, ymax=3,  xmin=-np.pi, xmax=2*np.pi, title="vixon")
trig_plot(cot,ymin=-3, ymax=3,  xmin=-np.pi, xmax=2*np.pi, title="comet")
trig_plot(csc,ymin=-3, ymax=3,  xmin=-np.pi, xmax=2*np.pi, title="cupid")
```

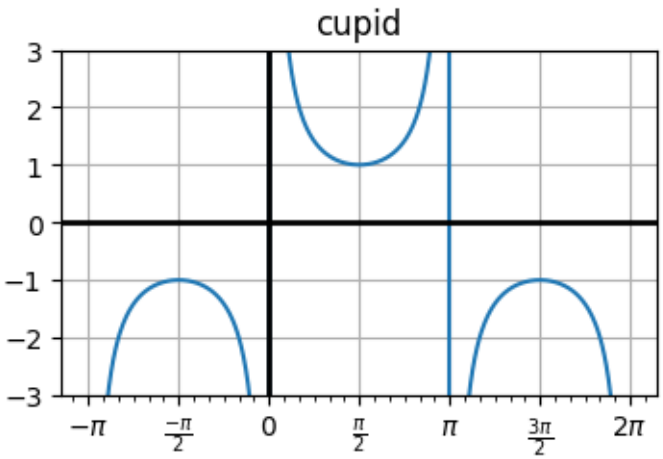




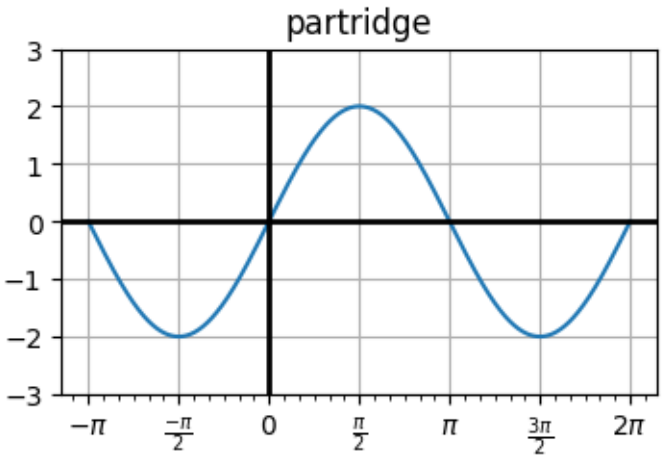

prancer
1
0
−1
−π
−π/2
0
π/2
π
3π/2
2π

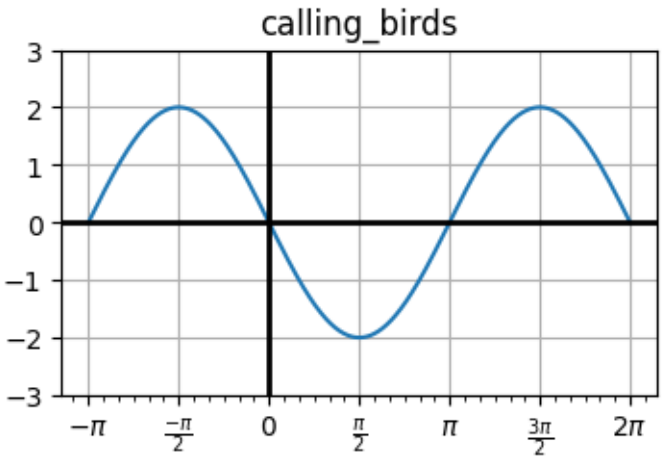

vixon
3
2
1
0
−1
−2
−3
−π
−π/2
0
π/2
π
3π/2
2π

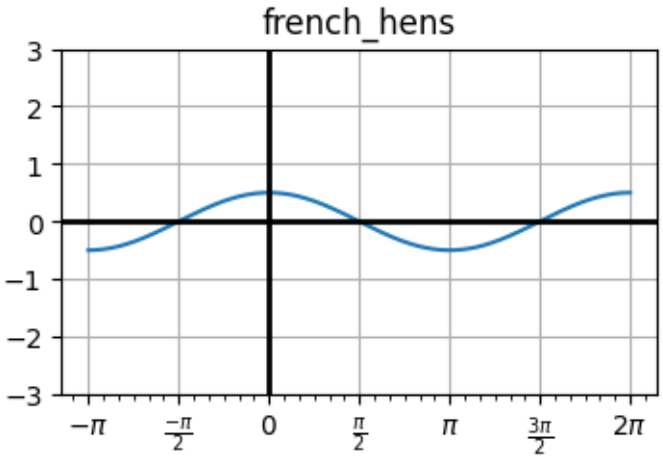

comet
3
2
1
0
−1
−2
−3
−π
−π/2
0
π/2
π
3π/2
2π

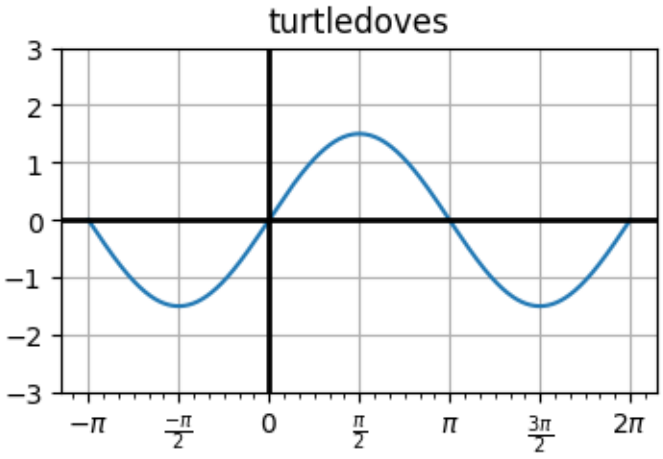


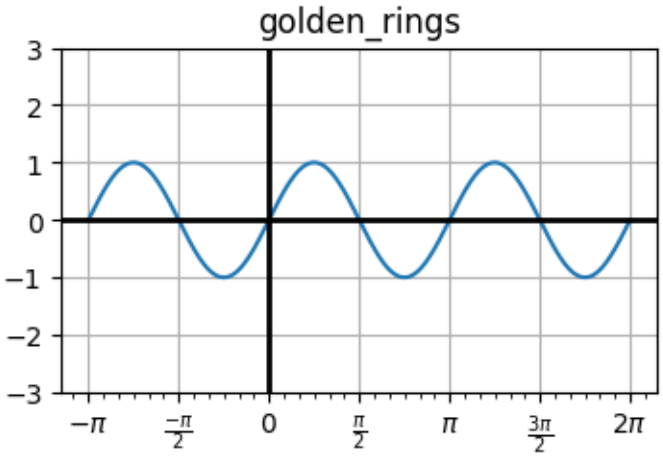
[15]:
```
trig_plot(lambda x:2*sin(x),ymin=-3, ymax=3,  xmin=-np.pi, xmax=2*np.pi,␣
 ↪title="partridge")
trig_plot(lambda x:-2*sin(x),ymin=-3, ymax=3,  xmin=-np.pi, xmax=2*np.pi,␣
 ↪title="calling_birds")
trig_plot(lambda x:0.5*cos(x),ymin=-3, ymax=3,  xmin=-np.pi, xmax=2*np.pi,␣
 ↪title="french_hens")
trig_plot(lambda x:1.5*sin(x),ymin=-3, ymax=3,  xmin=-np.pi, xmax=2*np.pi,␣
 ↪title="turtledoves")
trig_plot(lambda x:sin(2*x),ymin=-3, ymax=3,  xmin=-np.pi, xmax=2*np.pi,␣
 ↪title="golden_rings")
trig_plot(lambda x:cos(x/3),ymin=-3, ymax=3,  xmin=-np.pi, xmax=2*np.pi,␣
 ↪title="geese")
```

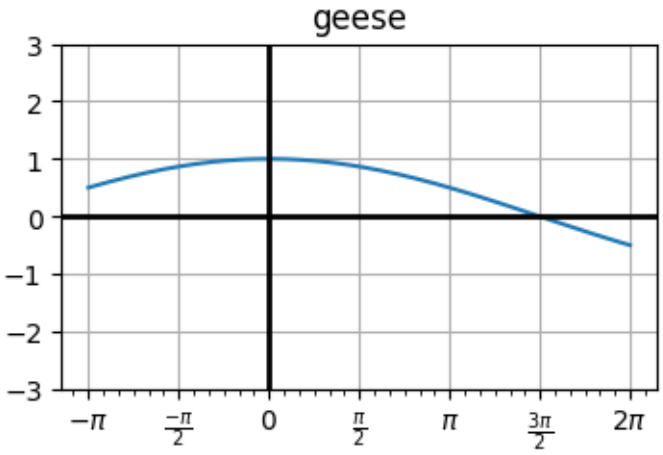



calling_birds
3
2
1
0
−1
−2
−3
−π
−π/2
0
π/2
π
3π/2
2π


french_hens
3
2
1
0
−1
−2
−3
−π
−π/2
0
π/2
π
3π/2
2π

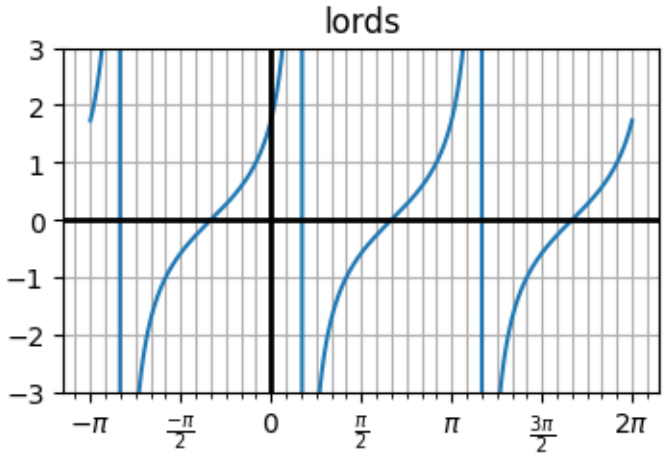

turtledoves
3
2
1
0
−1
−2
−3
−π
−π/2
0
π/2
π
3π/2
2π

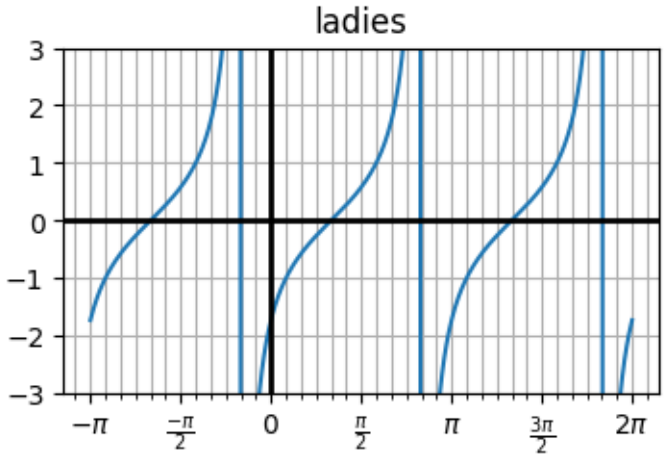


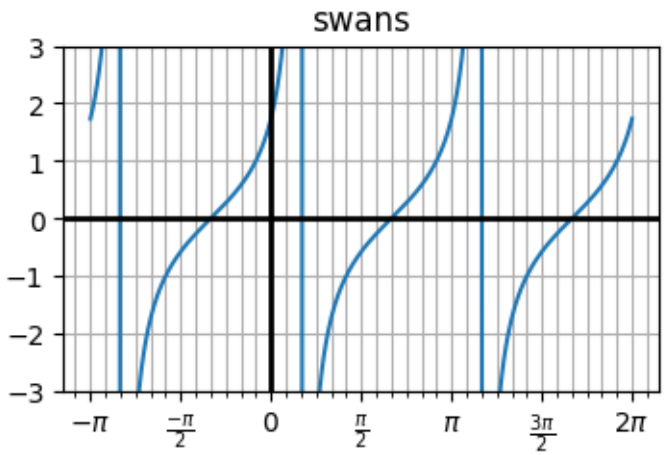


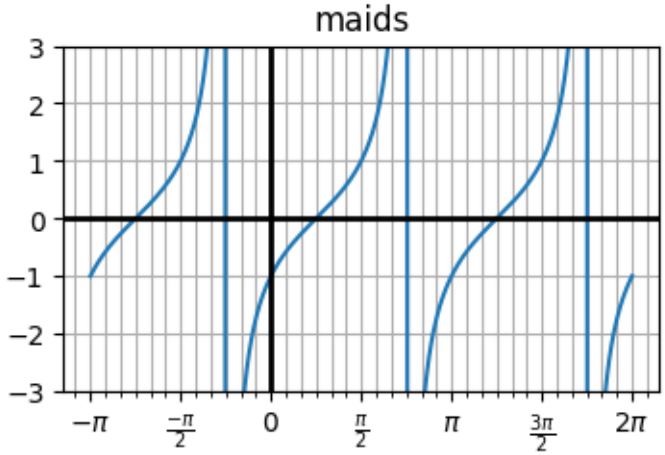
[14]:
```
pi = np.pi
trig_plot(lambda x:tan(x+pi/3),ymin=-3, ymax=3,  xmin=-np.pi, xmax=2*np.pi,␣
 ↪title="lords", show_minor_x = True)
trig_plot(lambda x:tan(x-pi/3),ymin=-3, ymax=3,  xmin=-np.pi, xmax=2*np.pi,␣
 ↪title="ladies", show_minor_x = True)
trig_plot(lambda x:tan(x-2*pi/3),ymin=-3, ymax=3,  xmin=-np.pi, xmax=2*np.pi,␣
 ↪title="swans", show_minor_x = True)
trig_plot(lambda x:tan(x+7*pi/4),ymin=-3, ymax=3,  xmin=-np.pi, xmax=2*np.pi,␣
 ↪title="maids", show_minor_x = True)
trig_plot(lambda x:sin(2*(x-pi/3)),ymin=-1.1, ymax=1.1,  xmin=-np.pi, xmax=2*np.
 ↪pi, title="pipers", show_minor_x = True)
```

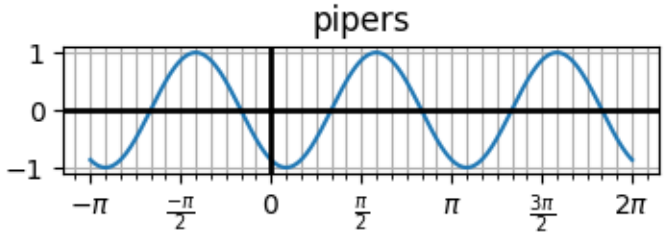
```
trig_plot(lambda x:cos(2*(x-pi/3)),ymin=-1.1, ymax=1.1,  xmin=-np.pi, xmax=2*np.
 ↪pi, title="drummers", show_minor_x = True)
```

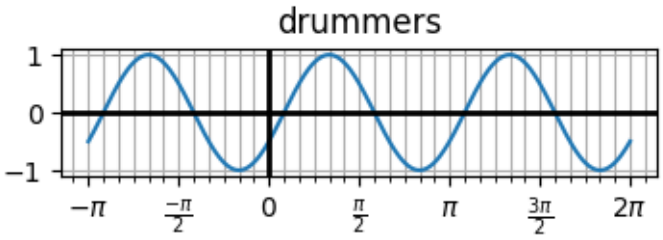





swans
3
2
1
0
−1
−2
−3
−π
−π/2
0
π/2
π
3π/2
2π


maids
3
2
1
0
−1
−2
−3
−π
−π/2
0
π/2
π
3π/2
2π


pipers
1
0
−1
−π
−π/2
0
π/2
π
3π/2
2π


drummers
1
0
−1
$-\pi$
$\frac{-\pi}{2}$
0
$\frac{\pi}{2}$
π
$\frac{3\pi}{2}$
2π

[ ]: