```
def main():
    grapher = CustomPlot(dim = dimension, aspect_ratio = 1)
    x1 = np.linspace(-32, 32, 100000)
    y1 = np.cos(x1)
    grapher.plot(x1, y1, color = 'red', label = r'$y=\cos\,x$')
    grapher.configure(axis_labels = ('$x$', '$y$', '$z$'))
    grapher.axis_fix(axis     = 'x',
                     symbolic = True,
                     s        = r'\pi',
                     v        = np.pi,
                     first    = -2,
                     last     = 2,
                     step     = 1 / 4)
    grapher.axis_fix(axis     = 'y',
                     symbolic = False,
                     s        = r'\pi',
                     v        = np.pi,
                     first    = -3,
                     last     = 3,
                     step     = 1)
    grapher.fig.tight_layout(pad = 2)
    plt.show()
```

Listing 1: Cosine

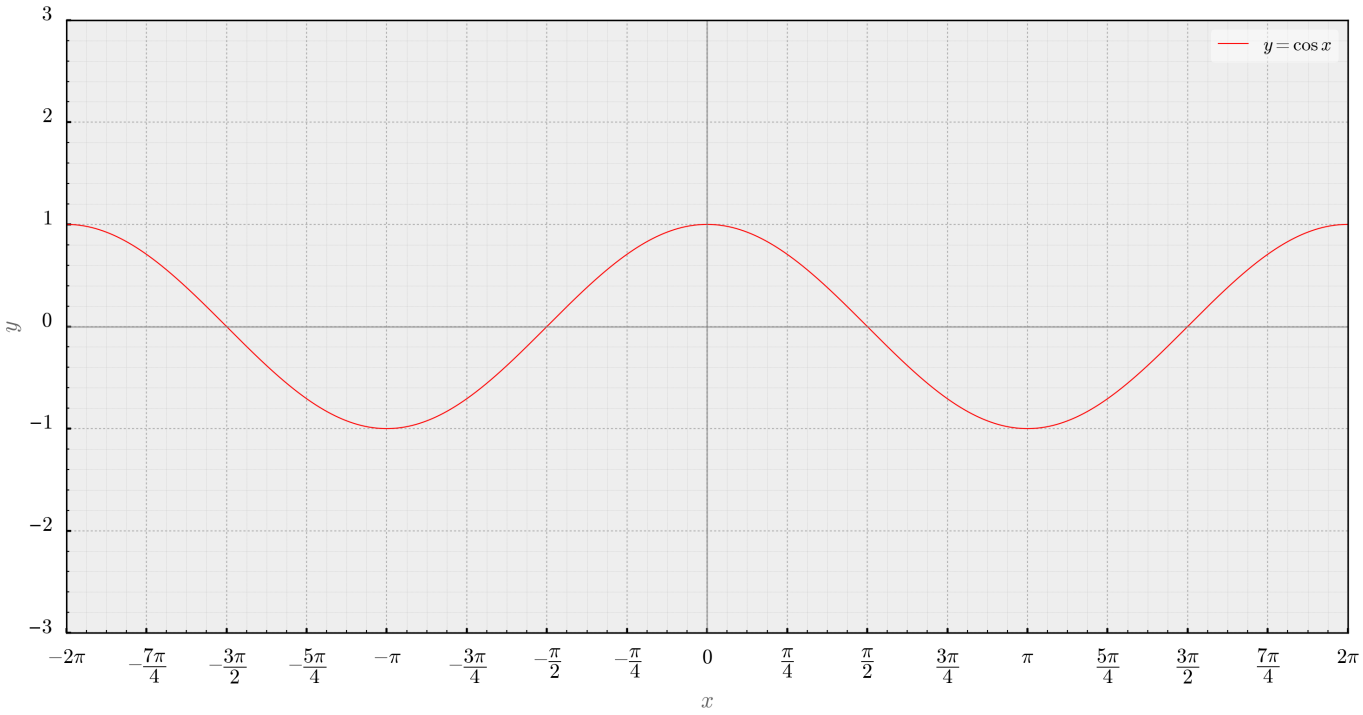

Figure 1: Cosine

```
def main():
    grapher = CustomPlot(dim = dimension, aspect_ratio = 1)
    x1 = np.linspace(-32, 32, 100000)
    y1 = np.sqrt(x1)
    grapher.plot(x1, y1, color = 'red', label = r'$y=\sqrt{x}$')
    grapher.configure(axis_labels = ('$x$', '$y$', '$z$'))
    grapher.axis_fix(axis     = 'x',
                     symbolic = False,
                     s        = r'\pi',
                     v        = np.pi,
                     first    = -2,
                     last     = 6,
                     step     = 1)
    grapher.axis_fix(axis     = 'y',
                     symbolic = False,
                     s        = r'\pi',
                     v        = np.pi,
                     first    = -1,
                     last     = 3,
                     step     = 1)
    grapher.fig.tight_layout(pad = 2)
    plt.show()
```

Listing 2: Square Root

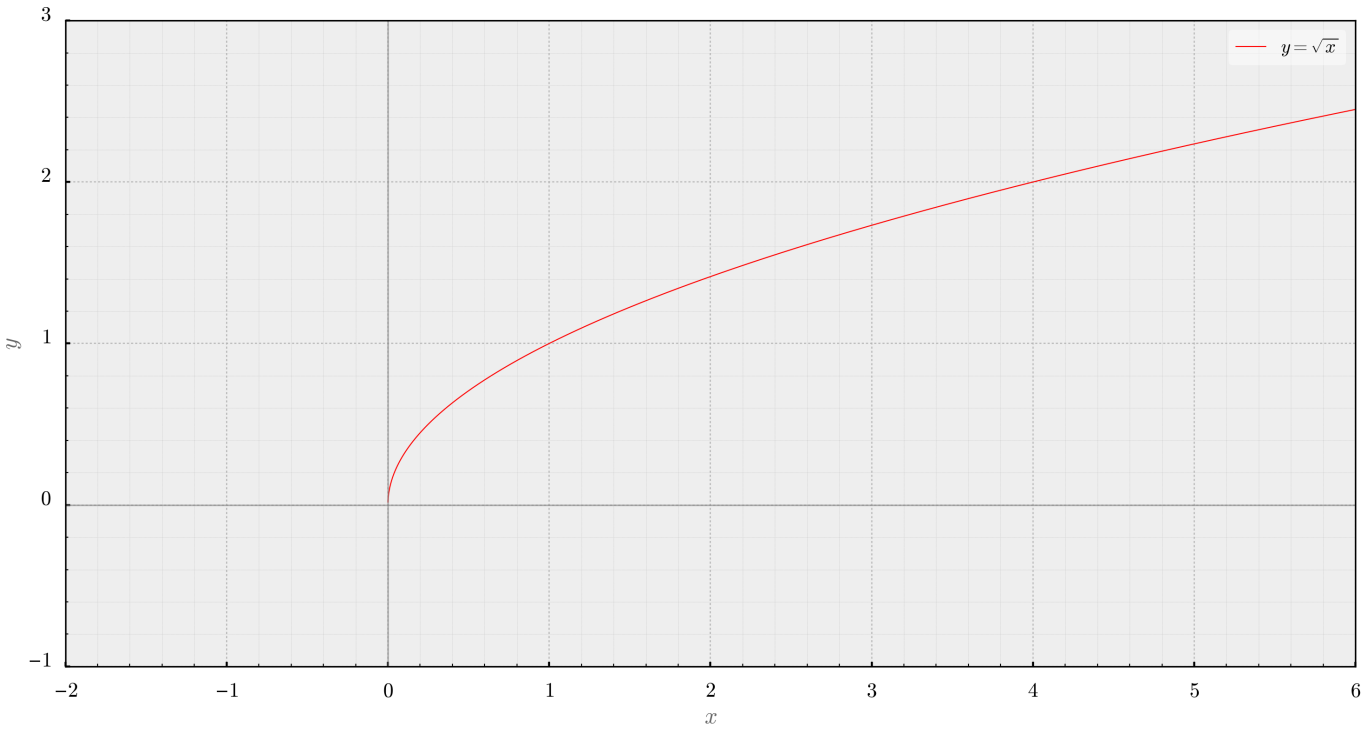

Figure 2: Square Root

You just have to change the `main` function in 'customplot.py' as described in the above examples. Leave everything else unchanged and run the program.