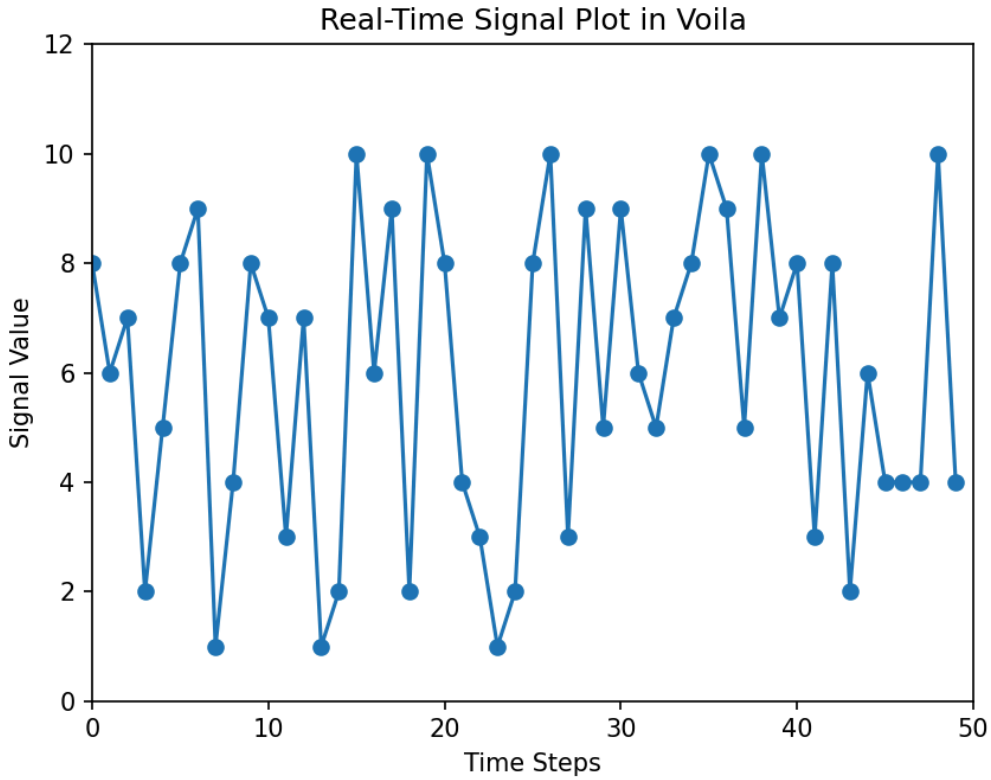
In [11]:
```
import matplotlib.pyplot as plt
import numpy as np
import random
import ipywidgets as widgets
from IPython.display import display
from matplotlib.animation import FuncAnimation
%matplotlib widget # allows interactive plotting




output = widgets.Output()


signal = [0] * 50


fig, ax = plt.subplots()
line, = ax.plot(signal, marker='o', linestyle='-')
ax.set_title("Real-Time Signal Plot in Voila")
ax.set_xlabel("Time Steps")
ax.set_ylabel("Signal Value")
ax.set_ylim(0, 12)


def get_new_signal_value():

    return random.randint(1, 10)  # Simulated input signal

# Function to update the plot
def update(frame):
    """Fetch new signal value and update the existing plot dynamically."""
    new_value = get_new_signal_value()
    signal.append(new_value)  # Append new data point
    signal.pop(0)  # Maintain fixed-length buffer

    # Update the existing plot data instead of creating a new plot
    line.set_ydata(signal)
    ax.set_xlim(0, len(signal))  # Adjust x-axis limits dynamically
    fig.canvas.draw_idle()  # Efficient redraw


ani = FuncAnimation(fig, update, interval=500)


with output:
    display(fig)

display(output)
```

```
Warning: Cannot change to a different GUI toolkit: widget. Using notebook instea
d.
```


Real-Time Signal Plot in Voila
Signal Value
Time Steps
0
2
4
6
8
10
12
0
10
20
30
40
50

```
Output()
```

In [ ]:

In [ ]: