Time buttons on our rainfall graph

Question

The local news station is wanting to update the graphics in the weather section of their website. They have requested a line chart with the ability to filter the data for the last 4 weeks (4WTD), last 48 hours (48HR), and the year to date (YTD). Using the rain DataFrame, create the chart and add buttons for this filtering.

Question Explanation

This task involves creating interactive buttons for a line chart visualization of rainfall data. The chart must enable filtering for specific time periods using Plotly's update_layout() function and the buttons configured with the required specifications.

Answer

Here is the solution to the task:

```python
import plotly.express as px
import pandas as pd

# Sample DataFrame
rain = pd.DataFrame({
    'Date': pd.date_range(start='2023-01-01', periods=365, freq='D'),
    'Rainfall': [i % 50 for i in range(365)]
})

# Create the buttons
date_buttons = [
    {'count': 28, 'label': '4WTD', 'step': 'day', 'stepmode': 'todate'},
    {'count': 48, 'label': '48HR', 'step': 'hour', 'stepmode': 'todate'},
    {'count': 1, 'label': 'YTD', 'step': 'year', 'stepmode': 'todate'}
]

# Create the basic line chart
fig = px.line(data_frame=rain, x='Date', y='Rainfall', title='Rainfall (mm)')

# Add the buttons and show
fig.update_layout(
    {'xaxis': {'rangeselector': {'buttons': date_buttons}}}
)
```

```
fig.show()
```

Answer Explanation

1. The code imports the necessary libraries, Plotly and pandas.
2. A sample DataFrame, `rain`, is created with dates and rainfall data.
3. The `date_buttons` list defines buttons for specific time filters (4 weeks, 48 hours, year-to-date).
4. A line chart is generated using Plotly Express with the rain DataFrame.
5. The `update_layout` function incorporates the buttons into the chart's layout for interactivity, enabling users to filter data based on the specified ranges.