

## A Happier Histogram Plot (Updated)

Learn / Courses / Introduction to Data Visualization w...

← Course Outline →

Daily XP 1295

Exercise

### A happier histogram plot

The stock exchange firm you created the histogram for thinks that all the data and plots being created are too impersonal.

They have requested that a positive message be added to the histogram plot of company revenues you recently created.

You have just the right idea - you can wish the viewer a happy day and use the current day of the week for this!

There is a `fig` histogram available for you, feel free to `show()` or `print()` it to remind yourself what it looks like.

Instructions

100 XP

- Position the annotation halfway along the x-axis and 95% up the y-axis.
- Set the correct values for `xref` and `yref` to allow for relative positioning.
- Use the provided `today` variable in the text, setting the font size to 20 and text color to `'white'`.
- Use the `update_layout()` method to add the annotation.

Take Hint (-30 XP)

script.py

Light Mode

```
1 # Get and format today's date
2 today = datetime.today().strftime('%A')
3
4 # Create the message_annotation
5 message_annotation = {
6     # Set the correct coordinates
7     'x': ____, 'y': ____, 'xref': ____, 'yref': ____,
8     # Set format the text and box
9     'text': f'Have a Happy {today} :)',
10    'font': {'size': ____, 'color': ____},
11    'bgcolor': 'rgb(237, 64, 200)', 'showarrow': False}
12
13 # Update the figure layout and show
14 fig.update_layout({____: [____]})
15 fig.show()
```

↺

Run Code

Submit Answer

IPython Shell

Slides

In [1]:

### Question:

The stock exchange firm you created the histogram for thinks that all the data and plots being created are too impersonal.

They have requested that a positive message be added to the histogram plot of company revenues you recently created.

You have just the right idea - you can wish the viewer a happy day and use the current day of the week for this!

Instructions:

1. Position the annotation halfway along the x-axis and 95% up the y-axis.
2. Set the correct values for ``xref`` and ``yref`` to allow for relative positioning.
3. Use the provided ``today`` variable in the text, setting the font size to 20 and text color to ``white``.
4. Use the ``update_layout()`` method to add the annotation.
5. Display the plot using ``fig.show()``.

### Answer:

```
# Get and format today's date
today = datetime.today().strftime('%A')

# Create the message annotation
message_annotation = {
    'x': 0.5, 'y': 0.95, 'xref': 'paper', 'yref': 'paper',
    # Set format for the text and box
    'text': f'Have a Happy {today}!',
    'font': {'size': 20, 'color': 'white'},
    'bgcolor': 'rgb(237, 64, 200)', 'showarrow': False
}

# Update the figure layout and show
fig.update_layout({'annotations': [message_annotation]})
fig.show()
```

### Explanation of the Answer:

The ``today`` variable is created using the ``datetime.today()`` method and formatted to display the current day of the week. An annotation is created with the ``message_annotation`` dictionary, which specifies the ``x`` and ``y`` positions as relative to the plot using ``xref`` and ``yref`` set to ``paper``. The ``text`` parameter includes the ``today`` variable, the ``font`` size is set to 20, and the ``color`` is set to ``white``. The background color is defined as ``'rgb(237, 64, 200)'` and ``showarrow`` is set to ``False``. The annotation is added to the figure using ``update_layout()``, and the plot is displayed with ``fig.show()``.