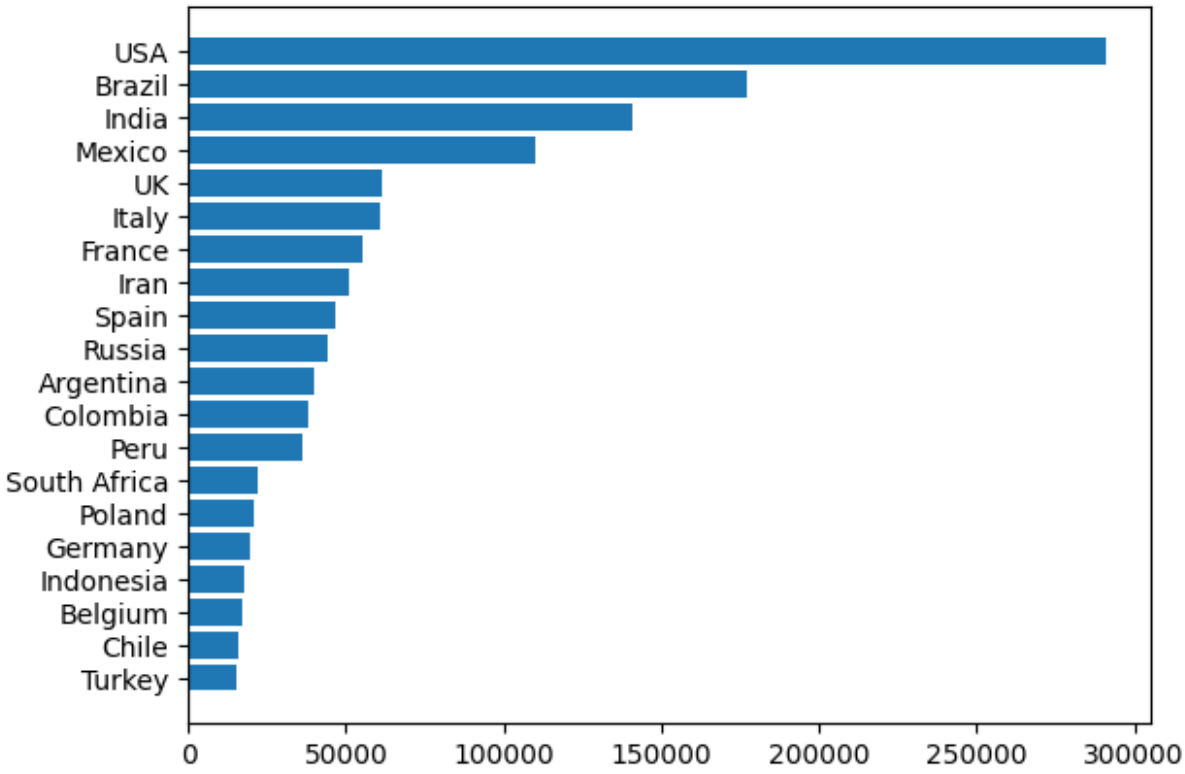
```
import pandas as pd
import matplotlib.pyplot as plt

death = pd.read_csv("deaths.csv")
death
```

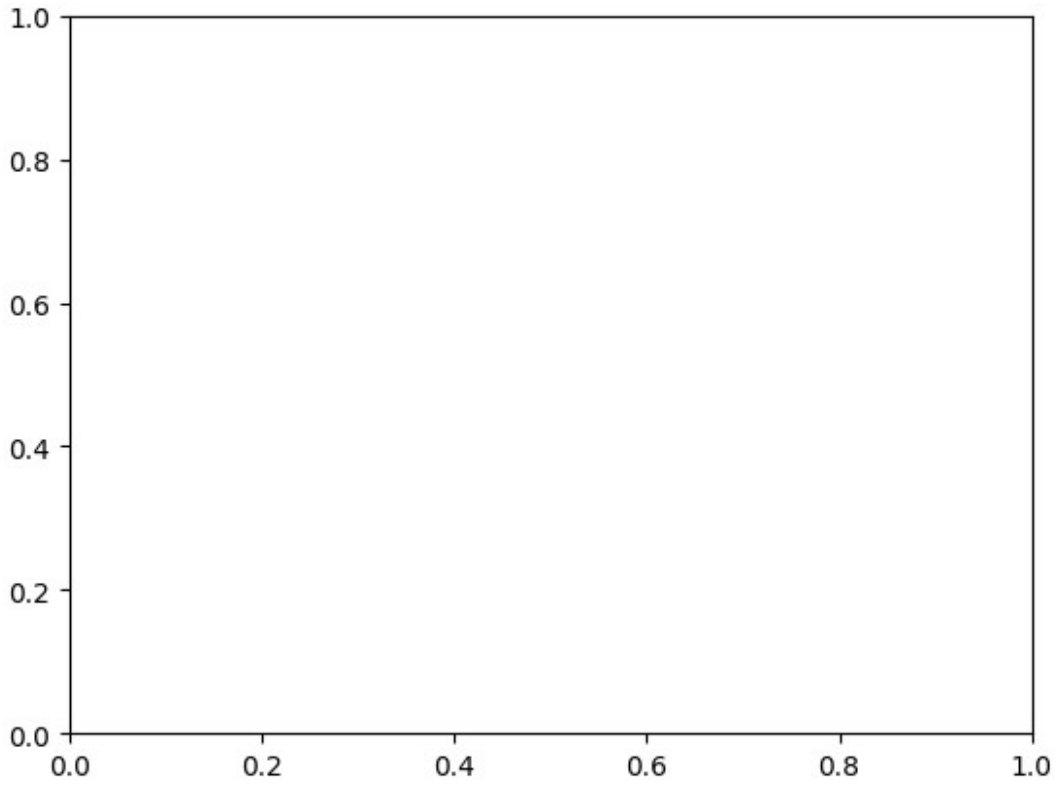
	Country_Other	Total_Deaths
0	Turkey	15103.0
1	Chile	15663.0
2	Belgium	17386.0
3	Indonesia	18000.0
4	Germany	19539.0
5	Poland	20592.0
6	South Africa	22249.0
7	Peru	36324.0
8	Colombia	37995.0
9	Argentina	39888.0
10	Russia	44159.0
11	Spain	46646.0
12	Iran	50917.0
13	France	55521.0
14	Italy	60606.0
15	UK	61434.0
16	Mexico	110074.0
17	India	141005.0
18	Brazil	177388.0
19	USA	290798.0

```
plt.barh(death.Country_Other, death.Total_Deaths) # Functional
interface
plt.show()
```



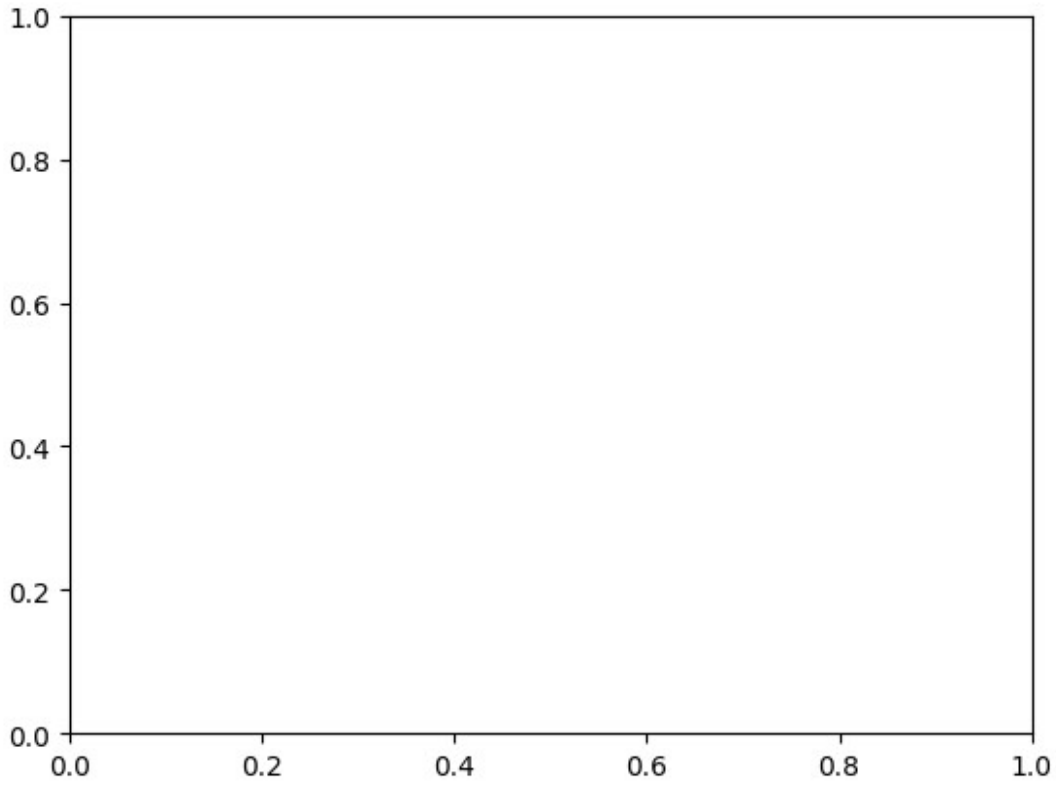

```
type(plt.subplots())
```

```
tuple
```

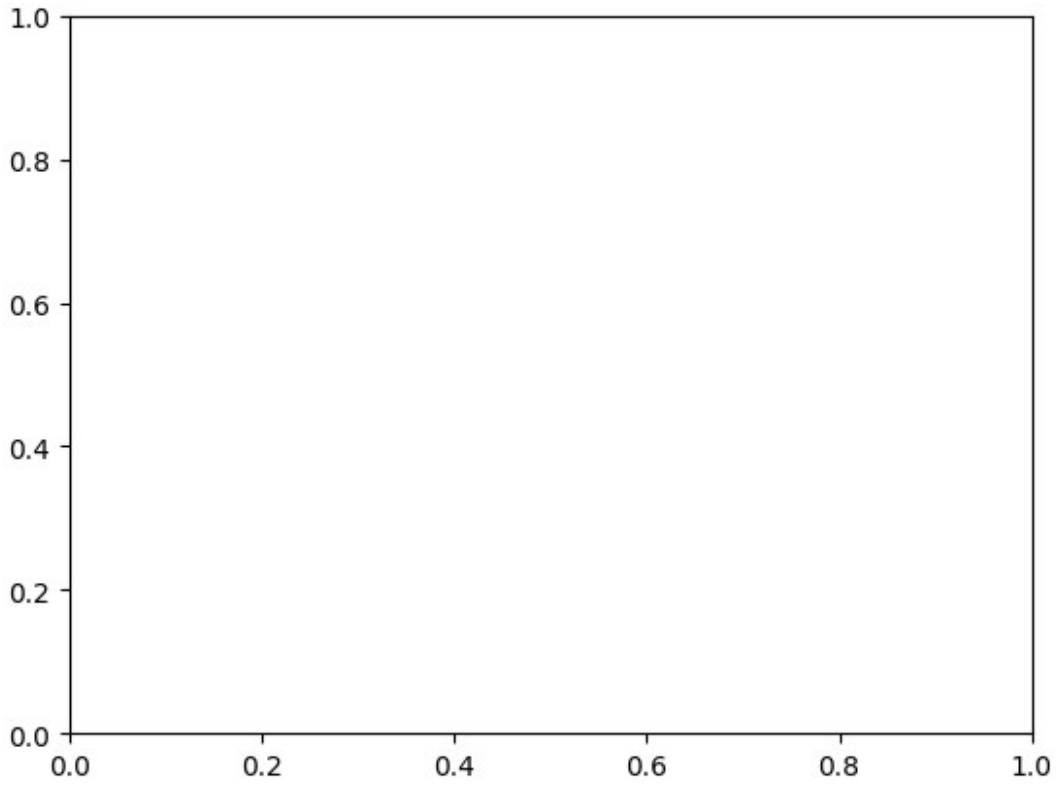



```
plt.subplots()
```

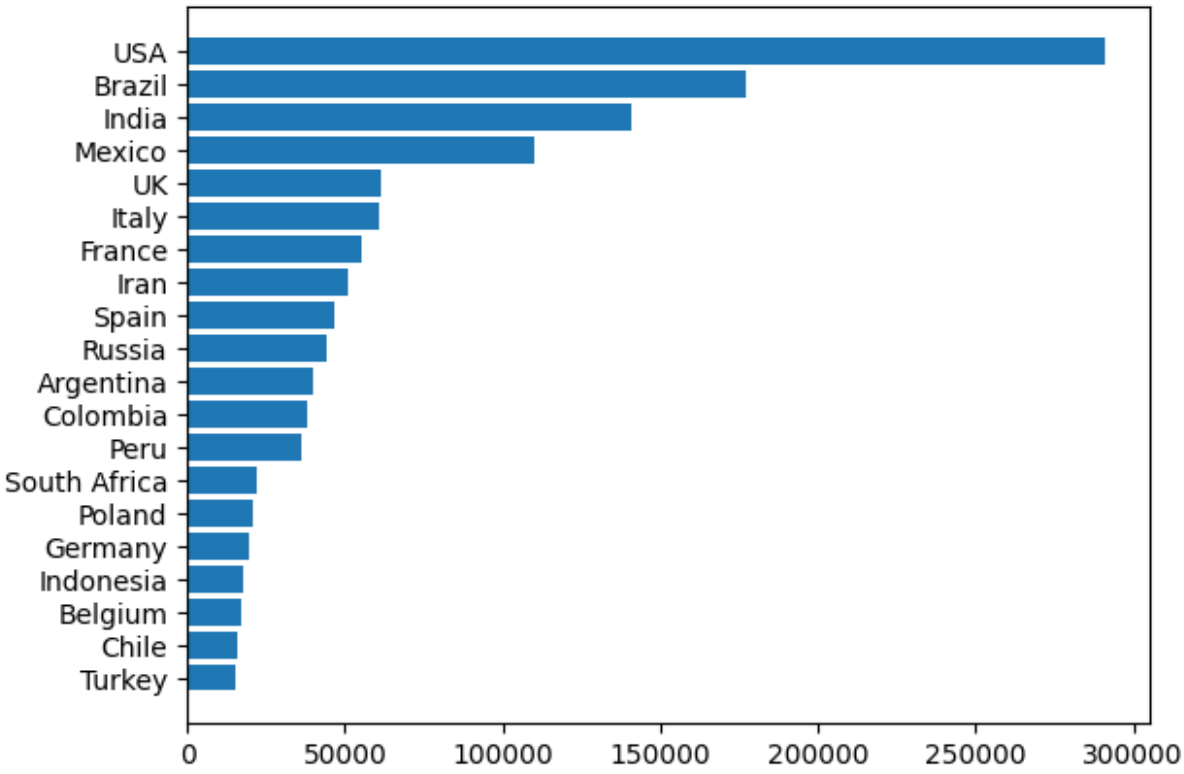
```
(<Figure size 640x480 with 1 Axes>, <Axes: >)
```



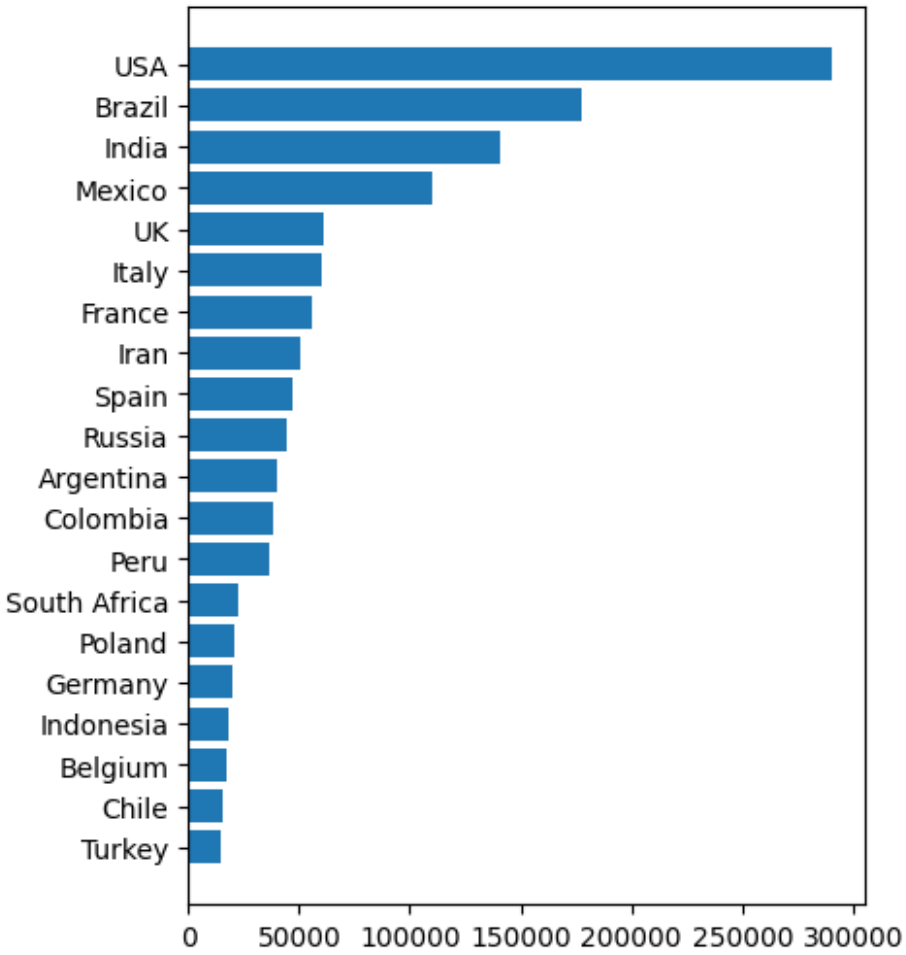

```
plt.subplot()

fig, ax = plt.subplots()
```




```
fig, ax = plt.subplots()

ax.barh(death.Country_Other, death.Total_Deaths) # Object oriented
interface
plt.show()
```




```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths) # Object oriented
interface
plt.show()
```



Design Principle - > help -> Generate a design options, choose among those option

- Familarity Principle
- Maximizing a Data ink ratio -> data ink/total ink (data, structural, decorations) = 0.75

```
5/(5+5+0)
```

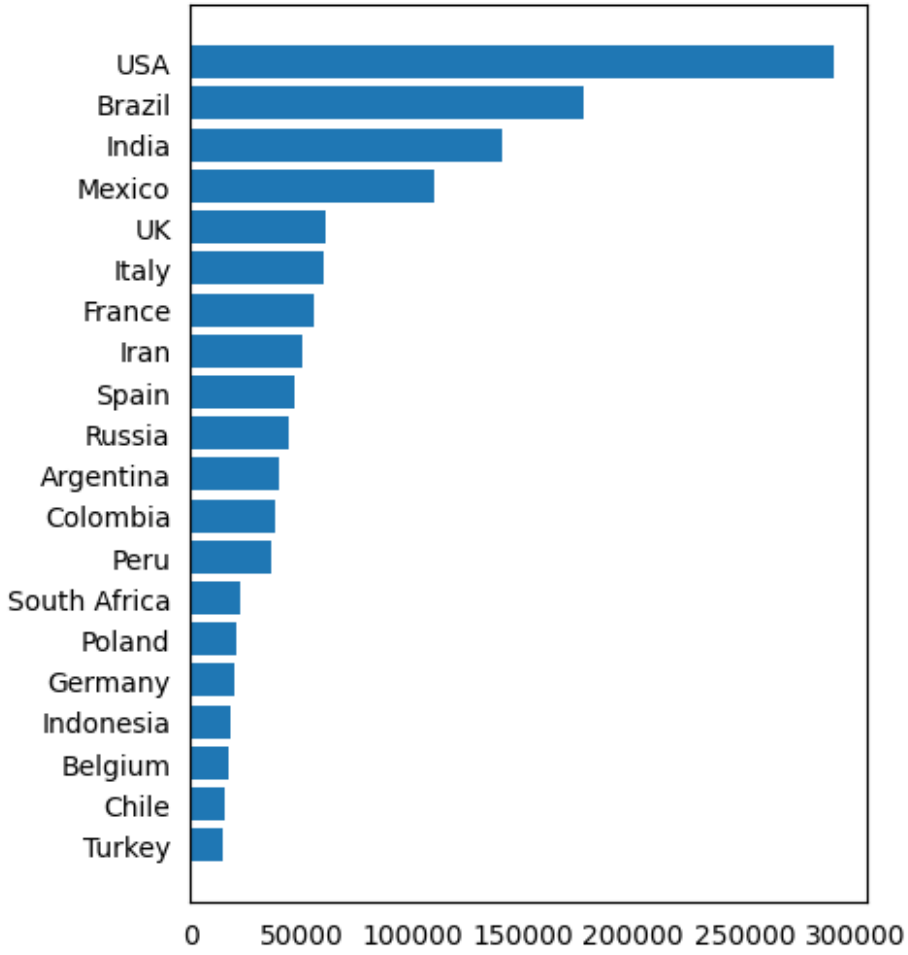
```
0.5
```

```
5/(5+2+0)
```

```
0.7142857142857143
```

```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths) # Object oriented
interface
```

```
ax.tick_params(bottom = False, left = False) # to remove ticks
plt.show()
```




```
dir(ax)
```

```
['ArtistList',
 '_AxesBase__clear',
 '_PROPERTIES_EXCLUDED_FROM_SET',
 '__class__',
 '__delattr__',
 '__dict__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getstate__',
 '__gt__',
 '__hash__',
```

```
 '__init__',
 '__init_subclass__',
 '__le__',
 '__lt__',
 '__module__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__setattr__',
 '__setstate__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 '__weakref__',
 '_add_text',
 '_adjustable',
 '_agg_filter',
 '_alias_map',
 '_alpha',
 '_anchor',
 '_animated',
 '_aspect',
 '_autotitlepos',
 '_axes',
 '_axes_locator',
 '_axis_map',
 '_axis_names',
 '_axisbelow',
 '_box_aspect',
 '_callbacks',
 '_check_no_units',
 '_children',
 '_clipon',
 '_clippath',
 '_cm_set',
 '_colorbars',
 '_convert_dx',
 '_current_image',
 '_different_canvas',
 '_errorevery_to_mask',
 '_facecolor',
 '_fill_between_x_or_y',
 '_frameon',
 '_fully_clipped_to_axes',
 '_gci',
 '_gen_axes_patch',
 '_gen_axes_spines',
```

```
 '_get_aspect_ratio',
 '_get_lines',
 '_get_pan_points',
 '_get_patches_for_fill',
 '_get_view',
 '_gid',
 '_gridOn',
 '_in_layout',
 '_init_axis',
 '_internal_update',
 '_label',
 '_label_outer_xaxis',
 '_label_outer_yaxis',
 '_left_title',
 '_make_twin_axes',
 '_mouseover',
 '_mouseover_set',
 '_navigate',
 '_navigate_mode',
 '_originalPosition',
 '_parse_scatter_color_args',
 '_path_effects',
 '_pcolorargs',
 '_picker',
 '_position',
 '_prepare_view_from_bbox',
 '_process_unit_info',
 '_projection_init',
 '_quiver_units',
 '_rasterization_zorder',
 '_rasterized',
 '_remove_legend',
 '_remove_method',
 '_request_autoscale_view',
 '_right_title',
 '_sci',
 '_set_alpha_for_array',
 '_set_artist_props',
 '_set_gc_clip',
 '_set_lim_and_transforms',
 '_set_position',
 '_set_title_offset_trans',
 '_set_view',
 '_set_view_from_bbox',
 '_shared_axes',
 '_sharex',
 '_sharey',
 '_sketch',
 '_snap',
```

```
 '_stale',
 '_stale_viewlims',
 '_sticky_edges',
 '_subclass_uses_cla',
 '_subplotspec',
 '_tight',
 '_transform',
 '_transformSet',
 '_twinned_axes',
 '_unit_change_handler',
 '_unstale_viewLim',
 '_update_image_limits',
 '_update_line_limits',
 '_update_patch_limits',
 '_update_props',
 '_update_set_signature_and_docstring',
 '_update_title_position',
 '_update_transScale',
 '_url',
 '_use_sticky_edges',
 '_validate_converted_limits',
 '_viewLim',
 '_visible',
 '_xaxis_transform',
 '_xmargin',
 '_yaxis_transform',
 '_ymargin',
 'acorr',
 'add_artist',
 'add_callback',
 'add_child_axes',
 'add_collection',
 'add_container',
 'add_image',
 'add_line',
 'add_patch',
 'add_table',
 'angle_spectrum',
 'annotate',
 'apply_aspect',
 'arrow',
 'artists',
 'autoscale',
 'autoscale_view',
 'axes',
 'axhline',
 'axhspan',
 'axis',
 'axison',
```

```
 'axline',
 'axvline',
 'axvspan',
 'bar',
 'bar_label',
 'barbs',
 'barh',
 'bbox',
 'boxplot',
 'broken_barh',
 'bxp',
 'callbacks',
 'can_pan',
 'can_zoom',
 'child_axes',
 'cla',
 'clabel',
 'clear',
 'clipbox',
 'cohere',
 'collections',
 'containers',
 'contains',
 'contains_point',
 'contour',
 'contourf',
 'convert_xunits',
 'convert_yunits',
 'csd',
 'dataLim',
 'drag_pan',
 'draw',
 'draw_artist',
 'ecdf',
 'end_pan',
 'errorbar',
 'eventplot',
 'figure',
 'fill',
 'fill_between',
 'fill_betweenx',
 'findobj',
 'fmt_xdata',
 'fmt_ydata',
 'format_coord',
 'format_cursor_data',
 'format_xdata',
 'format_ydata',
 'get_adjustable',
```

```
 'get_agg_filter',
 'get_alpha',
 'get_anchor',
 'get_animated',
 'get_aspect',
 'get_autoscale_on',
 'get_autoscalex_on',
 'get_autoscaley_on',
 'get_axes_locator',
 'get_axisbelow',
 'get_box_aspect',
 'get_children',
 'get_clip_box',
 'get_clip_on',
 'get_clip_path',
 'get_cursor_data',
 'get_data_ratio',
 'get_default_bbox_extra_artists',
 'get_facecolor',
 'get_fc',
 'get_figure',
 'get_frame_on',
 'get_gid',
 'get_gridspec',
 'get_images',
 'get_in_layout',
 'get_label',
 'get_legend',
 'get_legend_handles_labels',
 'get_lines',
 'get_mouseover',
 'get_navigate',
 'get_navigate_mode',
 'get_path_effects',
 'get_picker',
 'get_position',
 'get_rasterization_zorder',
 'get_rasterized',
 'get_shared_x_axes',
 'get_shared_y_axes',
 'get_sketch_params',
 'get_snap',
 'get_subplotspec',
 'get_tightbbox',
 'get_title',
 'get_transform',
 'get_transformed_clip_path_and_affine',
 'get_url',
 'get_visible',
```

```
 'get_window_extent',
 'get_xaxis',
 'get_xaxis_text1_transform',
 'get_xaxis_text2_transform',
 'get_xaxis_transform',
 'get_xbound',
 'get_xgridlines',
 'get_xlabel',
 'get_xlim',
 'get_xmajorticklabels',
 'get_xminorticklabels',
 'get_xscale',
 'get_xticklabels',
 'get_xticklines',
 'get_xticks',
 'get_yaxis',
 'get_yaxis_text1_transform',
 'get_yaxis_text2_transform',
 'get_yaxis_transform',
 'get_ybound',
 'get_ygridlines',
 'get_ylabel',
 'get_ylim',
 'get_ymajorticklabels',
 'get_yminorticklabels',
 'get_yscale',
 'get_yticklabels',
 'get_yticklines',
 'get_yticks',
 'get_zorder',
 'grid',
 'has_data',
 'have_units',
 'hexbin',
 'hist',
 'hist2d',
 'hlines',
 'ignore_existing_data_limits',
 'images',
 'imshow',
 'in_axes',
 'indicate_inset',
 'indicate_inset_zoom',
 'inset_axes',
 'invert_xaxis',
 'invert_yaxis',
 'is_transform_set',
 'label_outer',
 'legend',
```

```
 'legend_',
 'lines',
 'locator_params',
 'loglog',
 'magnitude_spectrum',
 'margins',
 'matshow',
 'minorticks_off',
 'minorticks_on',
 'mouseover',
 'name',
 'patch',
 'patches',
 'pchanged',
 'pcolor',
 'pcolorfast',
 'pcolormesh',
 'phase_spectrum',
 'pick',
 'pickable',
 'pie',
 'plot',
 'plot_date',
 'properties',
 'psd',
 'quiver',
 'quiverkey',
 'redraw_in_frame',
 'relim',
 'remove',
 'remove_callback',
 'reset_position',
 'scatter',
 'secondary_xaxis',
 'secondary_yaxis',
 'semilogx',
 'semilogy',
 'set',
 'set_adjustable',
 'set_agg_filter',
 'set_alpha',
 'set_anchor',
 'set_animated',
 'set_aspect',
 'set_autoscale_on',
 'set_autoscalex_on',
 'set_autoscaley_on',
 'set_axes_locator',
 'set_axis_off',
```

```
 'set_axis_on',
 'set_axisbelow',
 'set_box_aspect',
 'set_clip_box',
 'set_clip_on',
 'set_clip_path',
 'set_facecolor',
 'set_fc',
 'set_figure',
 'set_frame_on',
 'set_gid',
 'set_in_layout',
 'set_label',
 'set_mouseover',
 'set_navigate',
 'set_navigate_mode',
 'set_path_effects',
 'set_picker',
 'set_position',
 'set_prop_cycle',
 'set_rasterization_zorder',
 'set_rasterized',
 'set_sketch_params',
 'set_snap',
 'set_subplotspec',
 'set_title',
 'set_transform',
 'set_url',
 'set_visible',
 'set_xbound',
 'set_xlabel',
 'set_xlim',
 'set_xmargin',
 'set_xscale',
 'set_xticklabels',
 'set_xticks',
 'set_ybound',
 'set_ylabel',
 'set_ylim',
 'set_ymargin',
 'set_yscale',
 'set_yticklabels',
 'set_yticks',
 'set_zorder',
 'sharex',
 'sharey',
 'specgram',
 'spines',
 'spy',
```

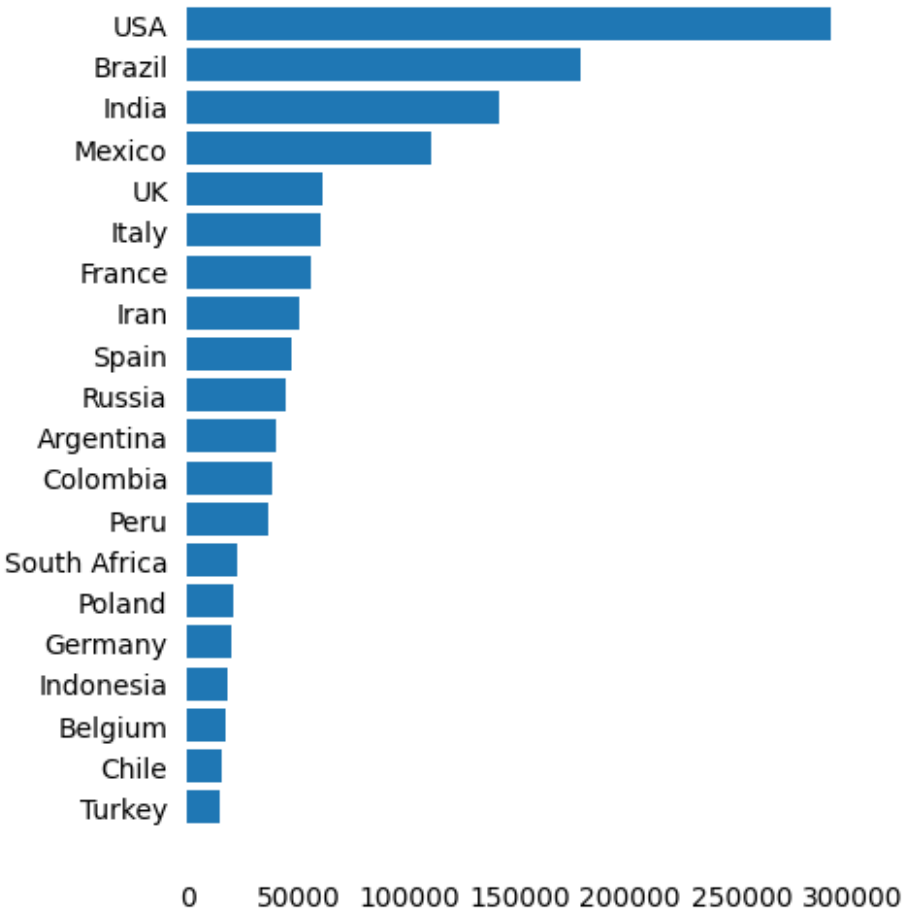
```
 'stackplot',
 'stairs',
 'stale',
 'stale_callback',
 'start_pan',
 'stem',
 'step',
 'sticky_edges',
 'streamplot',
 'table',
 'tables',
 'text',
 'texts',
 'tick_params',
 'ticklabel_format',
 'title',
 'titleOffsetTrans',
 'transAxes',
 'transData',
 'transLimits',
 'transScale',
 'tricontour',
 'tricontourf',
 'tripcolor',
 'triplot',
 'twinx',
 'twiny',
 'update',
 'update_datalim',
 'update_from',
 'use_sticky_edges',
 'viewLim',
 'violin',
 'violinplot',
 'vlines',
 'xaxis',
 'xaxis_date',
 'xaxis_inverted',
 'xcorr',
 'yaxis',
 'yaxis_date',
 'yaxis_inverted',
 'zorder']
```

```
ax.spines["right"]
```

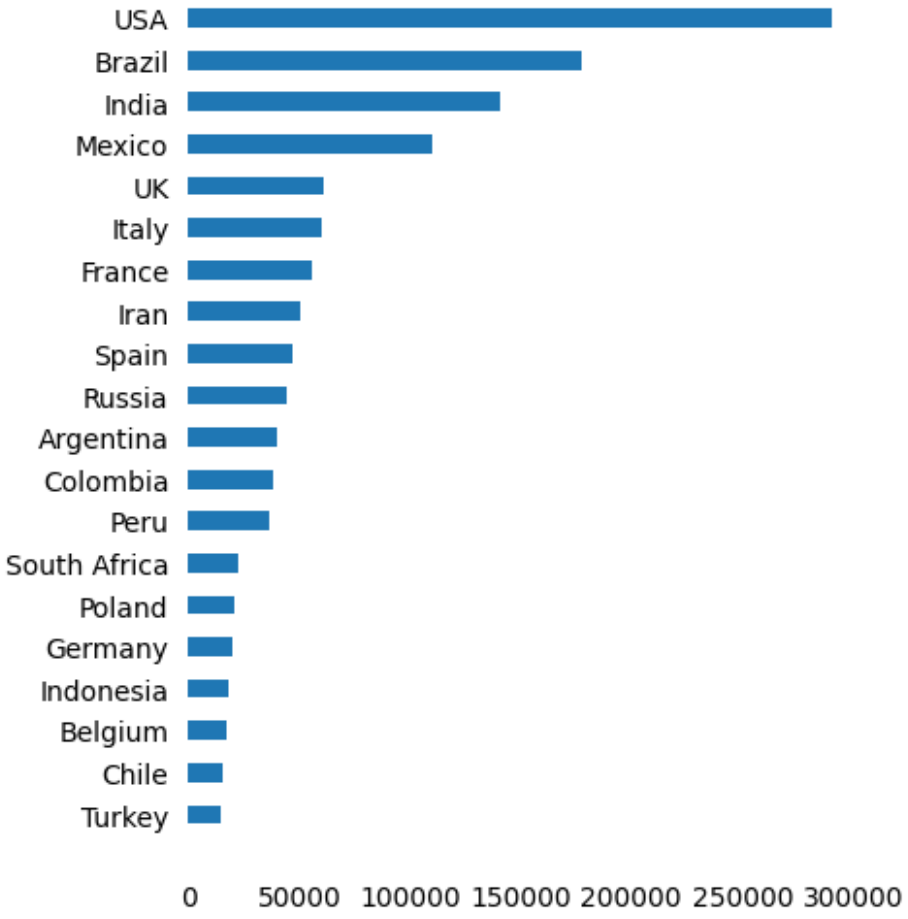
```
<matplotlib.spines.Spine at 0x11cf70fef90>
```

```
for spine in ax.spines:
    print(spine)
```

```
left
right
bottom
top

for sp_value in ax.spines.values():
    print(sp_value)

Spine
Spine
Spine
Spine

fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths) # Object oriented
interface
ax.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax.spines.values():
    sp_value.set_visible(False)
plt.show()
```



```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45) #
Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax.spines.values():
    sp_value.set_visible(False)
plt.show()
```

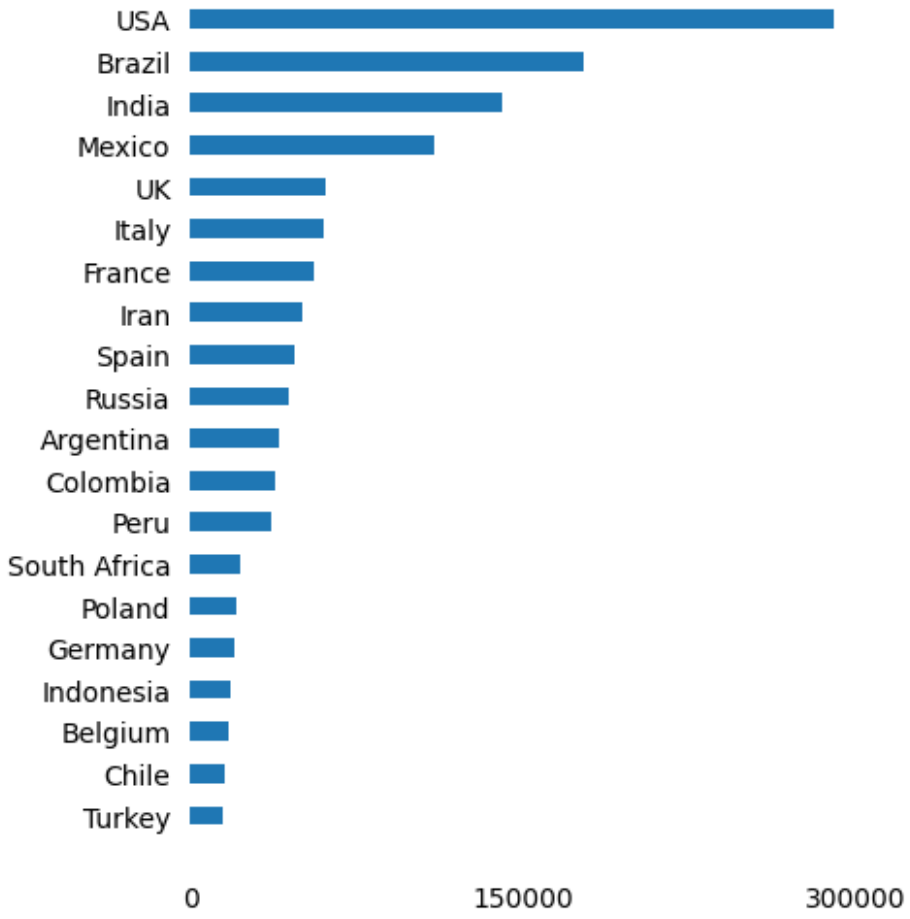



```
import numpy as np

np.linspace(0,300000,3)
```

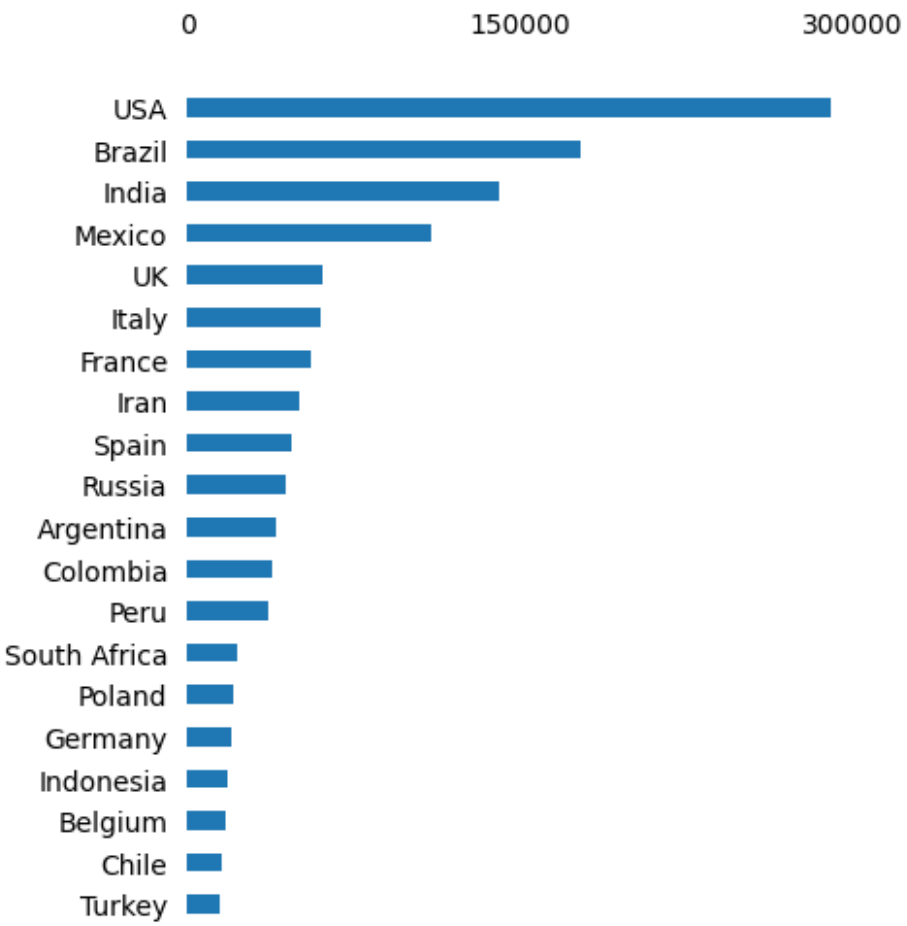
```
array([     0., 150000., 300000.])
```

```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45) #
Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
```

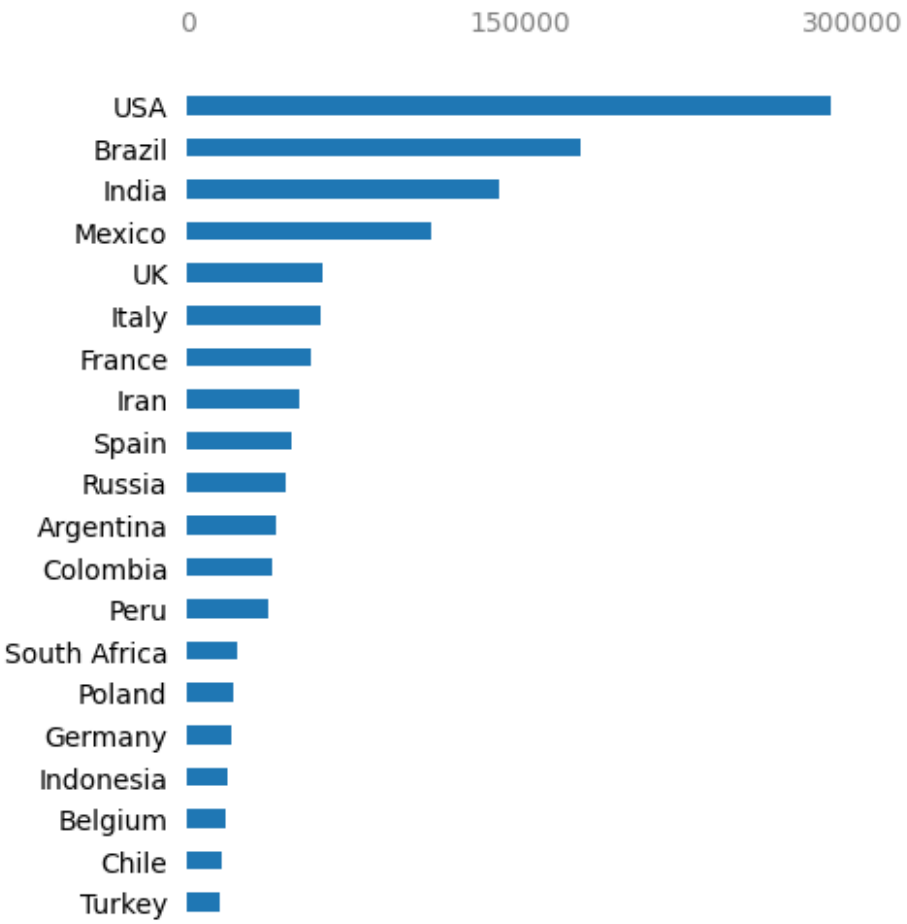
```
for sp_value in ax.spines.values():
    sp_value.set_visible(False)

ax.set_xticks([0,150000,300000])
plt.show()
```



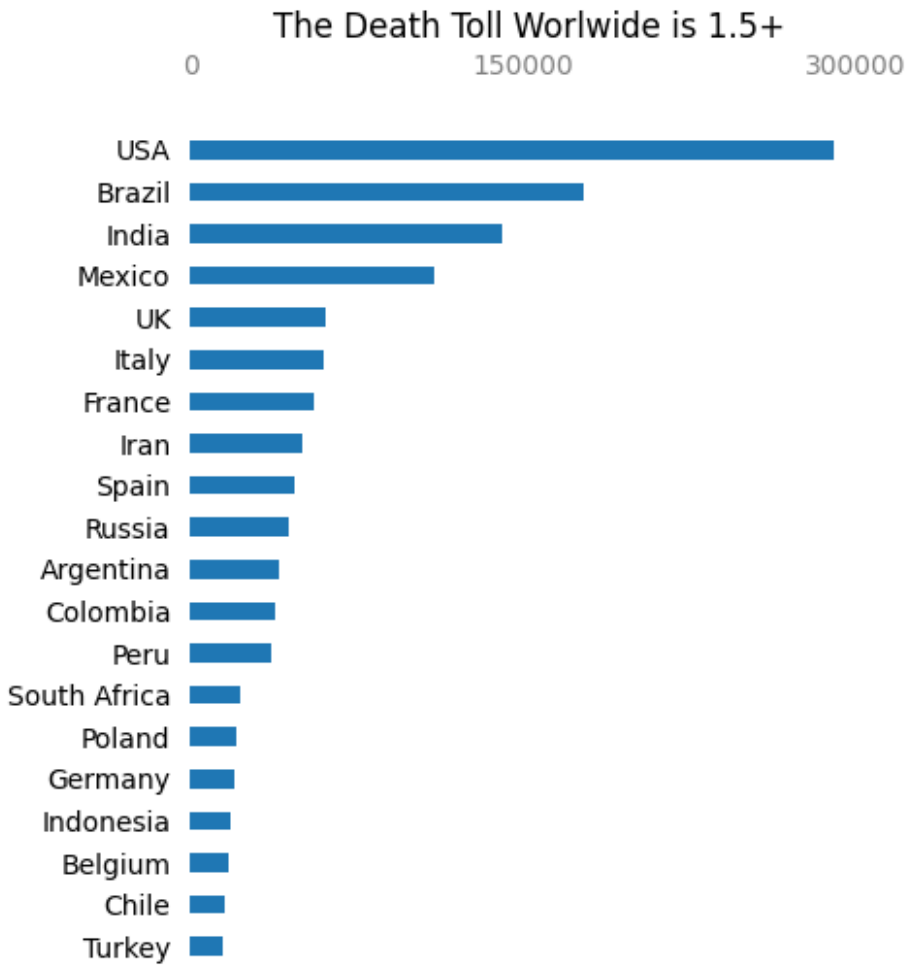

```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45) #
Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax.spines.values():
    sp_value.set_visible(False)

ax.set_xticks([0,150000,300000])
ax.xaxis.tick_top()

ax.tick_params(top = False)
plt.show()
```



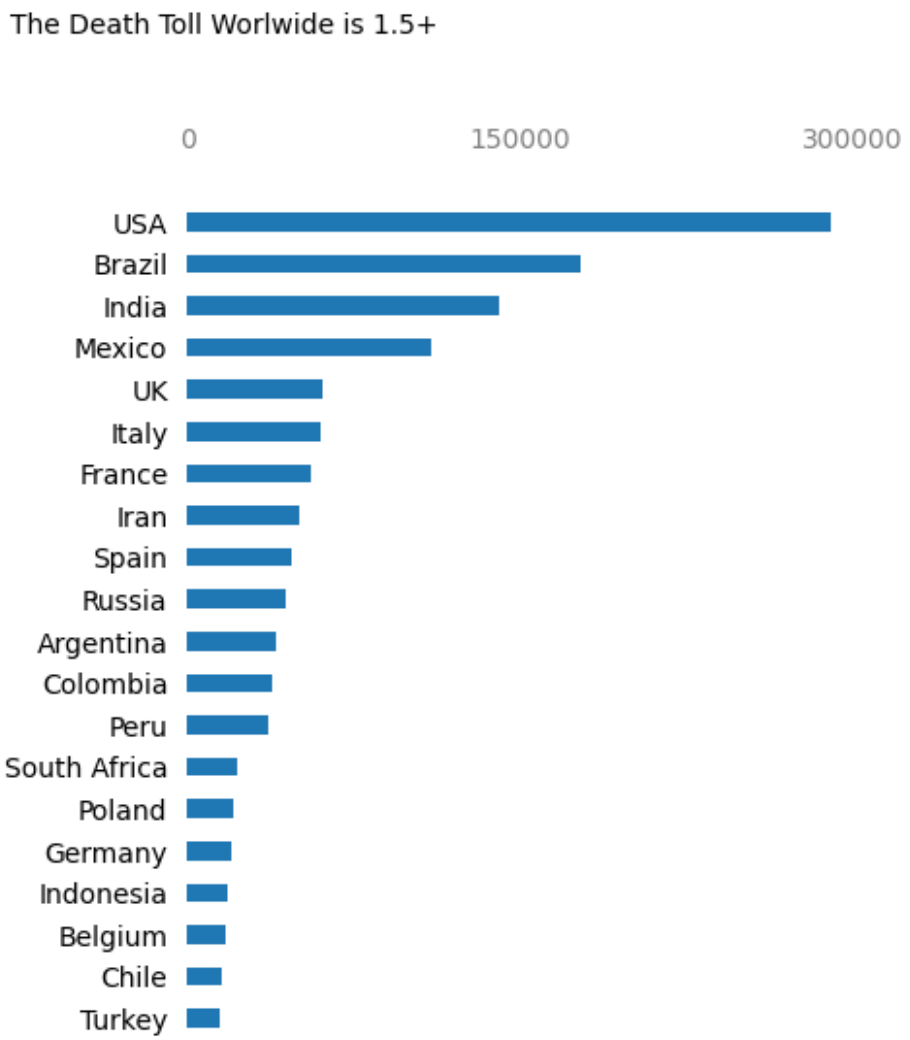

```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45) #
Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax.spines.values():
    sp_value.set_visible(False)

ax.set_xticks([0,150000,300000])
ax.xaxis.tick_top()

ax.tick_params(top = False)
ax.tick_params(axis = "x", colors ="grey")
plt.show()
```




```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45) #
Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax.spines.values():
    sp_value.set_visible(False)

ax.set_xticks([0,150000,300000])
ax.xaxis.tick_top()

ax.tick_params(top = False)
ax.tick_params(axis = "x", colors ="grey")
plt.title("The Death Toll Worlwide is 1.5+")
plt.show()
```

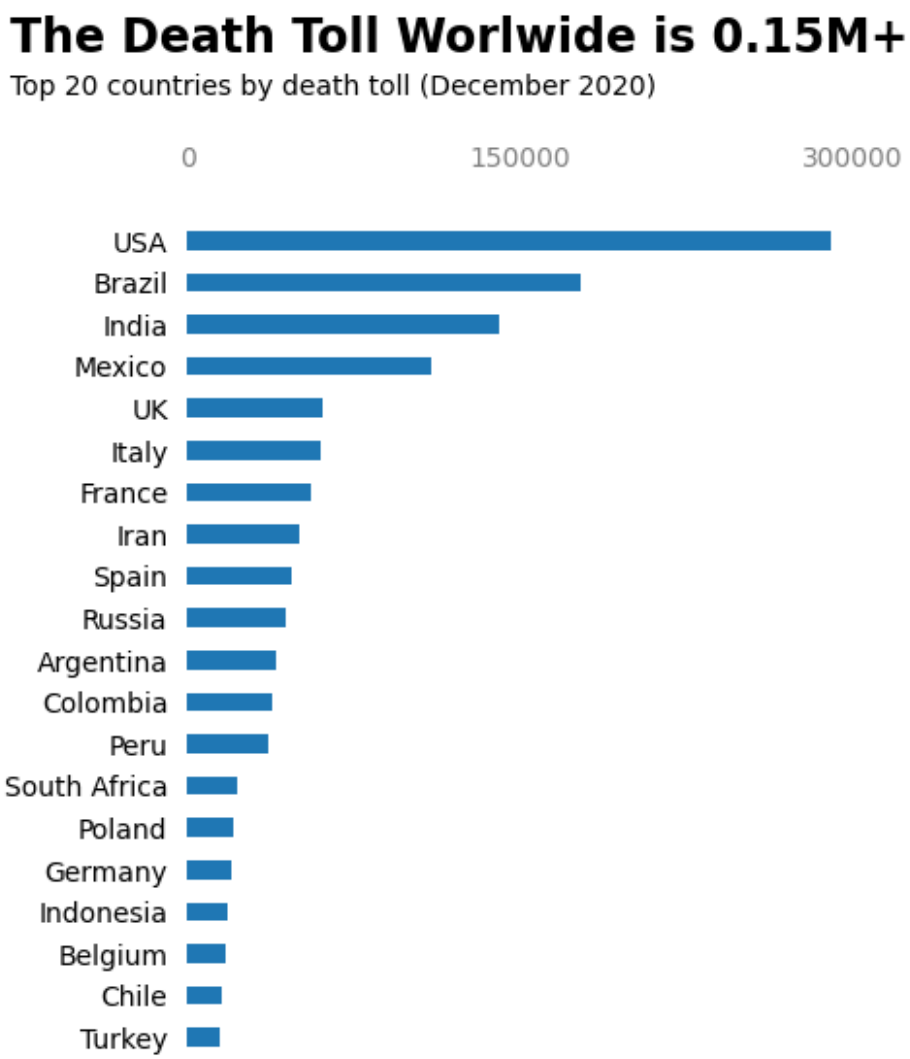



```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45) #
Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax.spines.values():
    sp_value.set_visible(False)

ax.set_xticks([0,150000,300000])
ax.xaxis.tick_top()

ax.tick_params(top = False)
ax.tick_params(axis = "x", colors ="grey")
ax.text(x = -80000, y = 23.5, s ="The Death Toll Worlwide is 1.5+")  #
plt.title("The Death Toll Worlwide is 1.5+")
plt.show()
```



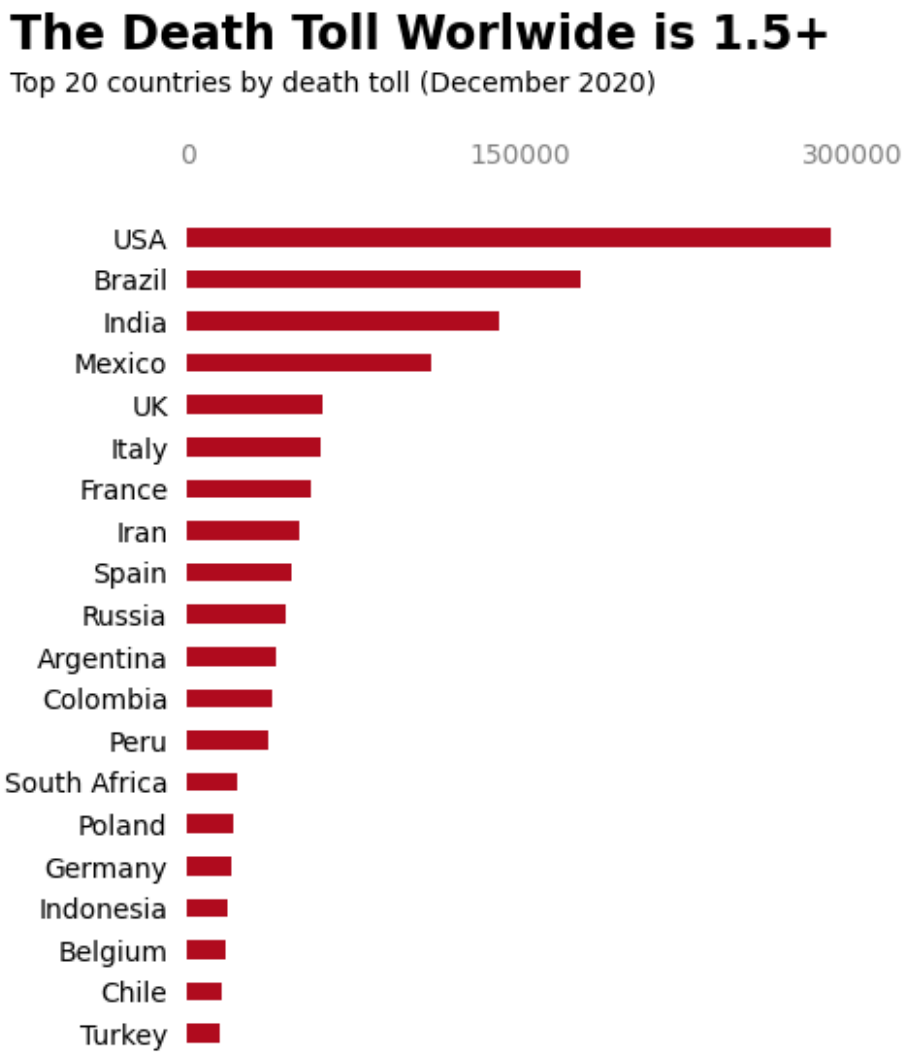

```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45) #
Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax.spines.values():
    sp_value.set_visible(False)

ax.set_xticks([0,150000,300000])
ax.xaxis.tick_top()

ax.tick_params(top = False)
ax.tick_params(axis = "x", colors ="grey")
ax.text(x = -80000, y = 23.5, s ="The Death Toll Worlwide is 0.15M+",
size = 17, weight = "bold")  # plt.title("The Death Toll Worlwide is
1.5+")
ax.text(x = -80000, y = 22.5, s ="Top 20 countries by death toll
```

```
(December 2020)", size = 10)
plt.show()
```




```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45, color
= "#b00b1e") # Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax.spines.values():
    sp_value.set_visible(False)

ax.set_xticks([0,150000,300000])
ax.xaxis.tick_top()

ax.tick_params(top = False)
ax.tick_params(axis = "x", colors ="grey")
```

```
ax.text(x = -80000, y = 23.5, s ="The Death Toll Worlwide is 1.5+",
size = 17, weight = "bold")  # plt.title("The Death Toll Worlwide is
1.5+")
ax.text(x = -80000, y = 22.5, s ="Top 20 countries by death toll
(December 2020)", size = 10)
plt.show()
```

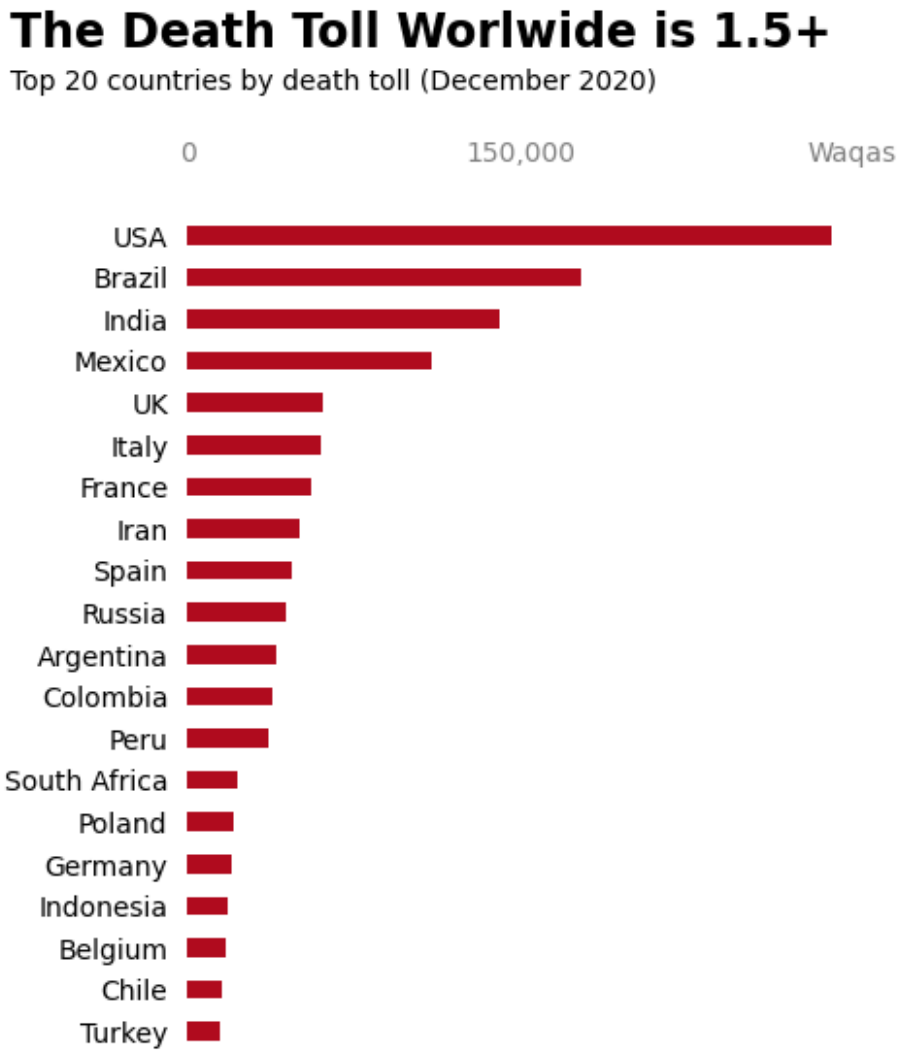



```
300,100
```

```
(300, 100)
```

```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45, color
= "#b00b1e") # Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
```

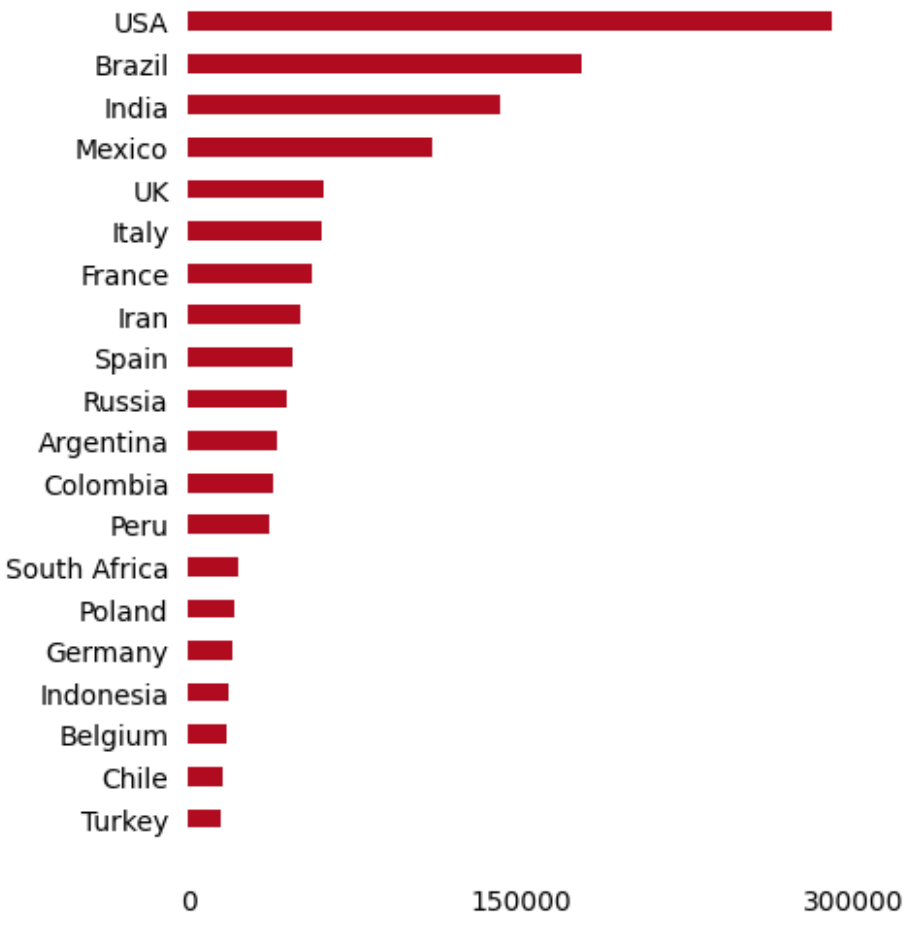
```
for sp_value in ax.spines.values():
    sp_value.set_visible(False)

ax.set_xticks([0,150000,300000])
ax.set_xticklabels(["0","150,000","Waqas"])
ax.xaxis.tick_top()

ax.tick_params(top = False)
ax.tick_params(axis = "x", colors ="grey")
ax.text(x = -80000, y = 23.5, s ="The Death Toll Worlwide is 1.5+",
size = 17, weight = "bold")  # plt.title("The Death Toll Worlwide is
1.5+")
ax.text(x = -80000, y = 22.5, s ="Top 20 countries by death toll
(December 2020)", size = 10)
plt.show()
```



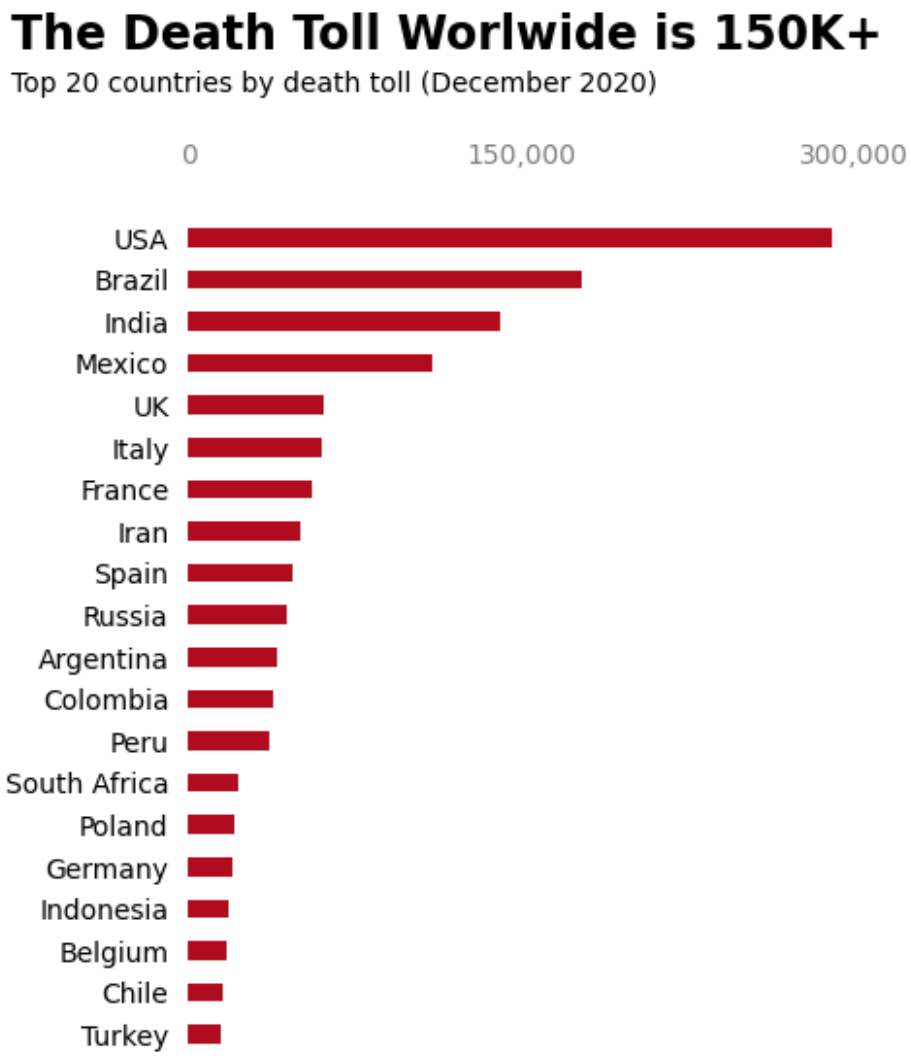
```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45, color
= "#b00b1e") # Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax.spines.values():
    sp_value.set_visible(False)

ax.set_xticks([0,150000,300000])
ax.set_xticklabels(["0","150,000","300,000","Waqas"])
ax.xaxis.tick_top()

ax.tick_params(top = False)
ax.tick_params(axis = "x", colors ="grey")
ax.text(x = -80000, y = 23.5, s ="The Death Toll Worlwide is 1.5+",
size = 17, weight = "bold")  # plt.title("The Death Toll Worlwide is
1.5+")
ax.text(x = -80000, y = 22.5, s ="Top 20 countries by death toll
(December 2020)", size = 10)
plt.show()
```

```
---------------------------------------------------------------------
-----
ValueError                                Traceback (most recent call
last)
Cell In[44], line 9
      6     sp_value.set_visible(False)
      8 ax.set_xticks([0,150000,300000])
----> 9 ax.set_xticklabels(["0","150,000","300,000","Waqas"])
     10 ax.xaxis.tick_top()
     12 ax.tick_params(top = False)

File ~\AppData\Roaming\Python\Python311\site-packages\matplotlib\axes\
_base.py:73, in
_axis_method_wrapper.__set_name__.<locals>.wrapper(self, *args,
**kwargs)
     72 def wrapper(self, *args, **kwargs):
---> 73     return get_method(self)(*args, **kwargs)

File ~\AppData\Roaming\Python\Python311\site-packages\matplotlib\_api\
deprecation.py:297, in rename_parameter.<locals>.wrapper(*args,
**kwargs)
    292     warn_deprecated(
    293         since, message=f"The {old!r} parameter of
{func.__name__}() "
    294         f"has been renamed {new!r} since Matplotlib {since};
support "
    295         f"for the old name will be dropped %(removal)s.")
    296     kwargs[new] = kwargs.pop(old)
--> 297 return func(*args, **kwargs)
```

```
File ~\AppData\Roaming\Python\Python311\site-packages\matplotlib\
axis.py:2025, in Axis.set_ticklabels(self, labels, minor, fontdict,
**kwargs)
   2021 elif isinstance(locator, mticker.FixedLocator):
   2022     # Passing [] as a list of labels is often used as a way to
   2023     # remove all tick labels, so only error for > 0 labels
   2024     if len(locator.locs) != len(labels) and len(labels) != 0:
-> 2025         raise ValueError(
   2026             "The number of FixedLocator locations"
   2027             f" ({len(locator.locs)}), usually from a call to"
   2028             " set_ticks, does not match"
   2029             f" the number of labels ({len(labels)}).")
   2030     tickd = {loc: lab for loc, lab in zip(locator.locs,
labels)}
   2031     func = functools.partial(self._format_with_dict, tickd)

ValueError: The number of FixedLocator locations (3), usually from a
call to set_ticks, does not match the number of labels (4).
```

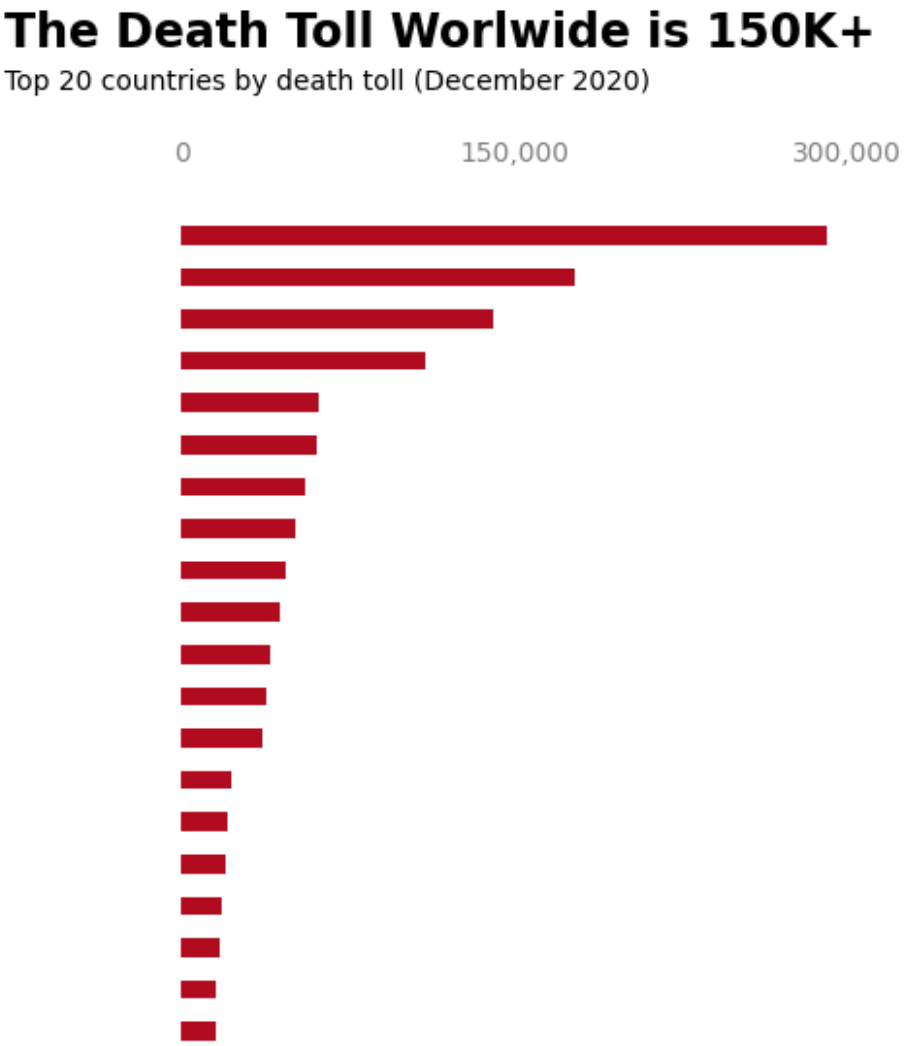


```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45, color
= "#b00b1e") # Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax.spines.values():
    sp_value.set_visible(False)

ax.set_xticks([0,150000,300000])
ax.set_xticklabels(["0","150,000","300,000",])
ax.xaxis.tick_top()

ax.tick_params(top = False)
ax.tick_params(axis = "x", colors ="grey")
ax.text(x = -80000, y = 23.5, s ="The Death Toll Worlwide is 150K+",
size = 17, weight = "bold")  # plt.title("The Death Toll Worlwide is
1.5+")
ax.text(x = -80000, y = 22.5, s ="Top 20 countries by death toll
(December 2020)", size = 10)
plt.show()
```




```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45, color
= "#b00b1e") # Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax.spines.values():
    sp_value.set_visible(False)

ax.set_xticks([0,150000,300000])
ax.set_xticklabels(["0","150,000","300,000",])
ax.set_yticklabels([])
ax.xaxis.tick_top()

ax.tick_params(top = False)
ax.tick_params(axis = "x", colors ="grey")
ax.text(x = -80000, y = 23.5, s ="The Death Toll Worlwide is 150K+",
```

```
size = 17, weight = "bold")  # plt.title("The Death Toll Worlwide is 1.5+")
ax.text(x = -80000, y = 22.5, s ="Top 20 countries by death toll (December 2020)", size = 10)
plt.show()
```




```
death["Country_Other"]
```

```
0            Turkey
1             Chile
2           Belgium
3         Indonesia
4           Germany
5            Poland
6      South Africa
7              Peru
```

```
8          Colombia
9         Argentina
10           Russia
11            Spain
12             Iran
13           France
14            Italy
15               UK
16           Mexico
17            India
18           Brazil
19              USA
Name: Country_Other, dtype: object
```

```
countries = death["Country_Other"].to_list()
```

```
for c in countries:
    print(c)
```

```
Turkey
Chile
Belgium
Indonesia
Germany
Poland
South Africa
Peru
Colombia
Argentina
Russia
Spain
Iran
France
Italy
UK
Mexico
India
Brazil
USA
```

```
a,b = (1,2)
```

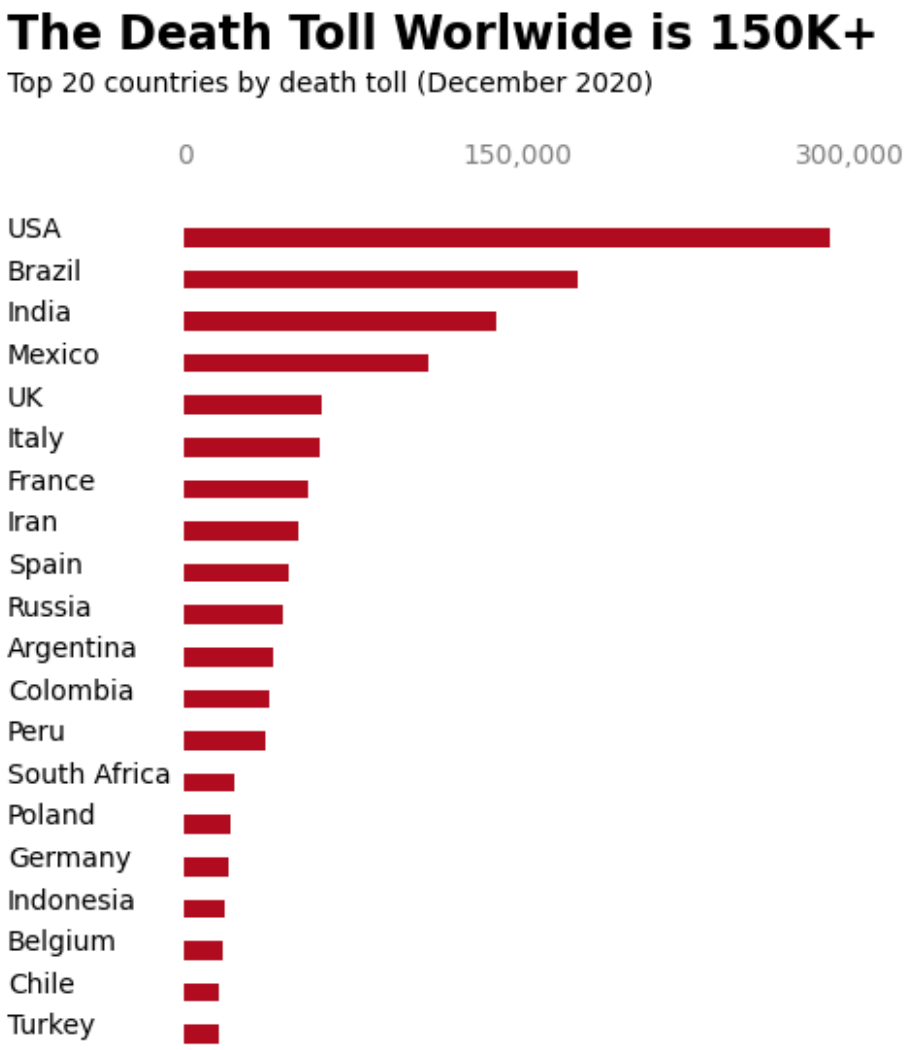
```
a
```

```
(1, 2)
```

```
for i,c in enumerate(countries):
    print(i,c)
```

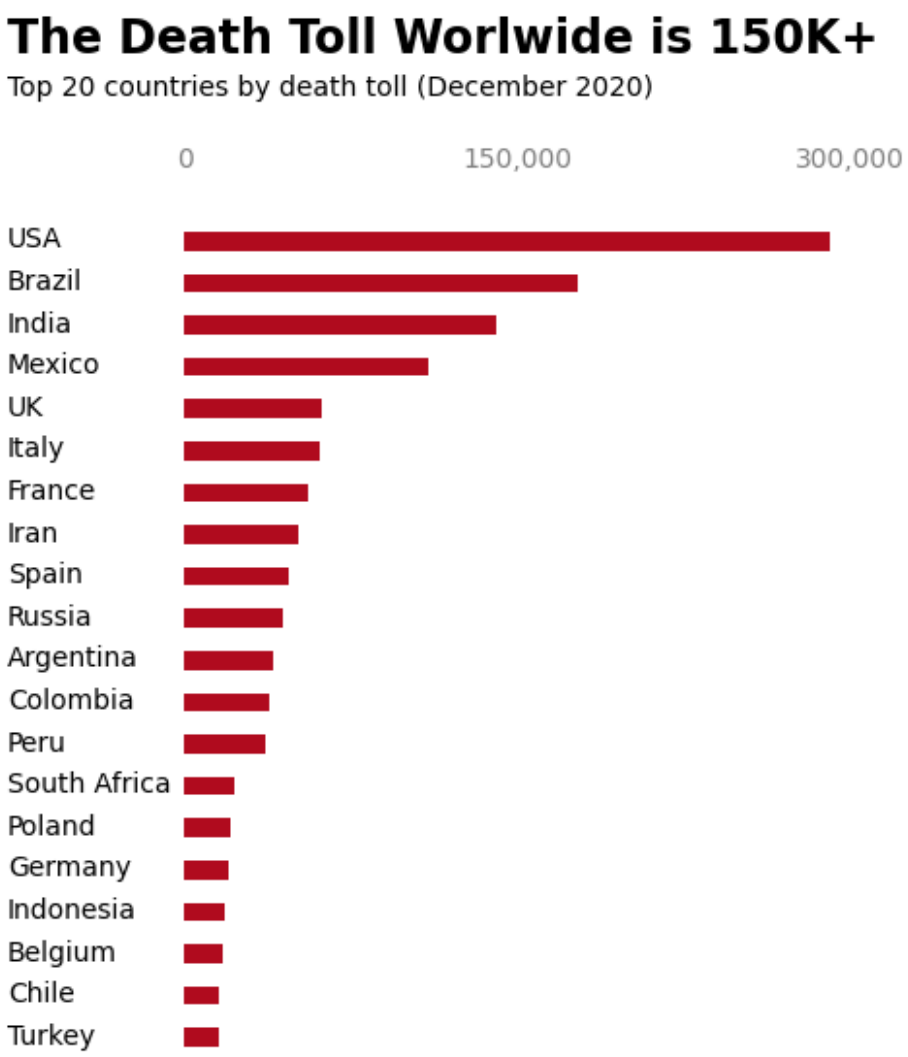
```
0 Turkey
1 Chile
```

```
2 Belgium
3 Indonesia
4 Germany
5 Poland
6 South Africa
7 Peru
8 Colombia
9 Argentina
10 Russia
11 Spain
12 Iran
13 France
14 Italy
15 UK
16 Mexico
17 India
18 Brazil
19 USA

fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45, color
= "#b00b1e") # Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax.spines.values():
    sp_value.set_visible(False)

ax.set_xticks([0,150000,300000])
ax.set_xticklabels(["0","150,000","300,000",])
ax.set_yticklabels([])
ax.xaxis.tick_top()

ax.tick_params(top = False)
ax.tick_params(axis = "x", colors ="grey")
ax.text(x = -80000, y = 23.5, s ="The Death Toll Worlwide is 150K+",
size = 17, weight = "bold")  # plt.title("The Death Toll Worlwide is
1.5+")
ax.text(x = -80000, y = 22.5, s ="Top 20 countries by death toll
(December 2020)", size = 10)

for i,c in enumerate(countries):
    ax.text(x = -80000, y = i, s = c)
plt.show()
```




```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45, color
= "#b00b1e") # Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax.spines.values():
    sp_value.set_visible(False)

ax.set_xticks([0,150000,300000])
ax.set_xticklabels(["0","150,000","300,000",])
ax.set_yticklabels([])
ax.xaxis.tick_top()

ax.tick_params(top = False)
ax.tick_params(axis = "x", colors ="grey")
ax.text(x = -80000, y = 23.5, s ="The Death Toll Worlwide is 150K+",
```

```
size = 17, weight = "bold")  # plt.title("The Death Toll Worlwide is 1.5+")
ax.text(x = -80000, y = 22.5, s ="Top 20 countries by death toll (December 2020)", size = 10)

for i,c in enumerate(countries):
    ax.text(x = -80000, y = i-0.15, s = c)
plt.show()
```




```
dir(ax)
```

```
['ArtistList',
 '_AxesBase__clear',
 '_PROPERTIES_EXCLUDED_FROM_SET',
 '__class__',
 '__delattr__',
```

```
 '__dict__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getstate__',
 '__gt__',
 '__hash__',
 '__init__',
 '__init_subclass__',
 '__le__',
 '__lt__',
 '__module__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__setattr__',
 '__setstate__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 '__weakref__',
 '_add_text',
 '_adjustable',
 '_agg_filter',
 '_alias_map',
 '_alpha',
 '_anchor',
 '_animated',
 '_aspect',
 '_autotitlepos',
 '_axes',
 '_axes_locator',
 '_axis_map',
 '_axis_names',
 '_axisbelow',
 '_box_aspect',
 '_callbacks',
 '_check_no_units',
 '_children',
 '_clipon',
 '_clippath',
 '_cm_set',
 '_colorbars',
 '_convert_dx',
```

```
 '_current_image',
 '_different_canvas',
 '_errorevery_to_mask',
 '_facecolor',
 '_fill_between_x_or_y',
 '_frameon',
 '_fully_clipped_to_axes',
 '_gci',
 '_gen_axes_patch',
 '_gen_axes_spines',
 '_get_aspect_ratio',
 '_get_lines',
 '_get_pan_points',
 '_get_patches_for_fill',
 '_get_view',
 '_gid',
 '_gridOn',
 '_in_layout',
 '_init_axis',
 '_internal_update',
 '_label',
 '_label_outer_xaxis',
 '_label_outer_yaxis',
 '_left_title',
 '_make_twin_axes',
 '_mouseover',
 '_mouseover_set',
 '_navigate',
 '_navigate_mode',
 '_originalPosition',
 '_parse_scatter_color_args',
 '_path_effects',
 '_pcolorargs',
 '_picker',
 '_position',
 '_prepare_view_from_bbox',
 '_process_unit_info',
 '_projection_init',
 '_quiver_units',
 '_rasterization_zorder',
 '_rasterized',
 '_remove_legend',
 '_remove_method',
 '_request_autoscale_view',
 '_right_title',
 '_sci',
 '_set_alpha_for_array',
 '_set_artist_props',
 '_set_gc_clip',
```

```
 '_set_lim_and_transforms',
 '_set_position',
 '_set_title_offset_trans',
 '_set_view',
 '_set_view_from_bbox',
 '_shared_axes',
 '_sharex',
 '_sharey',
 '_sketch',
 '_snap',
 '_stale',
 '_stale_viewlims',
 '_sticky_edges',
 '_subclass_uses_cla',
 '_subplotspec',
 '_tight',
 '_transform',
 '_transformSet',
 '_twinned_axes',
 '_unit_change_handler',
 '_unstale_viewLim',
 '_update_image_limits',
 '_update_line_limits',
 '_update_patch_limits',
 '_update_props',
 '_update_set_signature_and_docstring',
 '_update_title_position',
 '_update_transScale',
 '_url',
 '_use_sticky_edges',
 '_validate_converted_limits',
 '_viewLim',
 '_visible',
 '_xaxis_transform',
 '_xmargin',
 '_yaxis_transform',
 '_ymargin',
 'acorr',
 'add_artist',
 'add_callback',
 'add_child_axes',
 'add_collection',
 'add_container',
 'add_image',
 'add_line',
 'add_patch',
 'add_table',
 'angle_spectrum',
 'annotate',
```

```
 'apply_aspect',
 'arrow',
 'artists',
 'autoscale',
 'autoscale_view',
 'axes',
 'axhline',
 'axhspan',
 'axis',
 'axison',
 'axline',
 'axvline',
 'axvspan',
 'bar',
 'bar_label',
 'barbs',
 'barh',
 'bbox',
 'boxplot',
 'broken_barh',
 'bxp',
 'callbacks',
 'can_pan',
 'can_zoom',
 'child_axes',
 'cla',
 'clabel',
 'clear',
 'clipbox',
 'cohere',
 'collections',
 'containers',
 'contains',
 'contains_point',
 'contour',
 'contourf',
 'convert_xunits',
 'convert_yunits',
 'csd',
 'dataLim',
 'drag_pan',
 'draw',
 'draw_artist',
 'ecdf',
 'end_pan',
 'errorbar',
 'eventplot',
 'figure',
 'fill',
```

```
 'fill_between',
 'fill_betweenx',
 'findobj',
 'fmt_xdata',
 'fmt_ydata',
 'format_coord',
 'format_cursor_data',
 'format_xdata',
 'format_ydata',
 'get_adjustable',
 'get_agg_filter',
 'get_alpha',
 'get_anchor',
 'get_animated',
 'get_aspect',
 'get_autoscale_on',
 'get_autoscalex_on',
 'get_autoscaley_on',
 'get_axes_locator',
 'get_axisbelow',
 'get_box_aspect',
 'get_children',
 'get_clip_box',
 'get_clip_on',
 'get_clip_path',
 'get_cursor_data',
 'get_data_ratio',
 'get_default_bbox_extra_artists',
 'get_facecolor',
 'get_fc',
 'get_figure',
 'get_frame_on',
 'get_gid',
 'get_gridspec',
 'get_images',
 'get_in_layout',
 'get_label',
 'get_legend',
 'get_legend_handles_labels',
 'get_lines',
 'get_mouseover',
 'get_navigate',
 'get_navigate_mode',
 'get_path_effects',
 'get_picker',
 'get_position',
 'get_rasterization_zorder',
 'get_rasterized',
 'get_shared_x_axes',
```

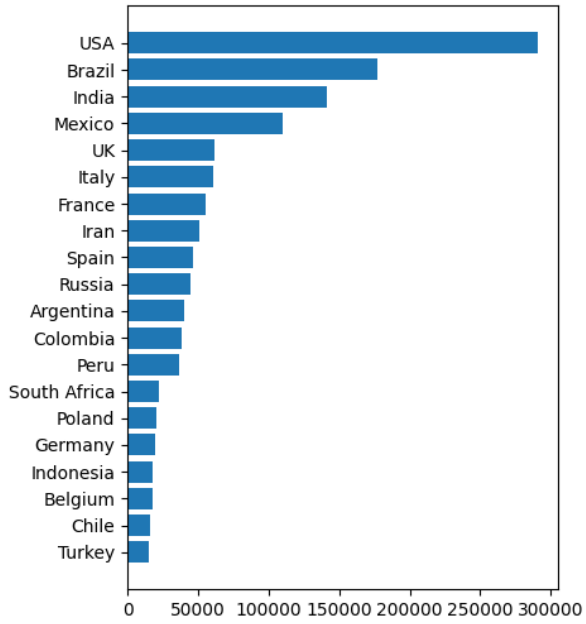
```
 'get_shared_y_axes',
 'get_sketch_params',
 'get_snap',
 'get_subplotspec',
 'get_tightbbox',
 'get_title',
 'get_transform',
 'get_transformed_clip_path_and_affine',
 'get_url',
 'get_visible',
 'get_window_extent',
 'get_xaxis',
 'get_xaxis_text1_transform',
 'get_xaxis_text2_transform',
 'get_xaxis_transform',
 'get_xbound',
 'get_xgridlines',
 'get_xlabel',
 'get_xlim',
 'get_xmajorticklabels',
 'get_xminorticklabels',
 'get_xscale',
 'get_xticklabels',
 'get_xticklines',
 'get_xticks',
 'get_yaxis',
 'get_yaxis_text1_transform',
 'get_yaxis_text2_transform',
 'get_yaxis_transform',
 'get_ybound',
 'get_ygridlines',
 'get_ylabel',
 'get_ylim',
 'get_ymajorticklabels',
 'get_yminorticklabels',
 'get_yscale',
 'get_yticklabels',
 'get_yticklines',
 'get_yticks',
 'get_zorder',
 'grid',
 'has_data',
 'have_units',
 'hexbin',
 'hist',
 'hist2d',
 'hlines',
 'ignore_existing_data_limits',
 'images',
```

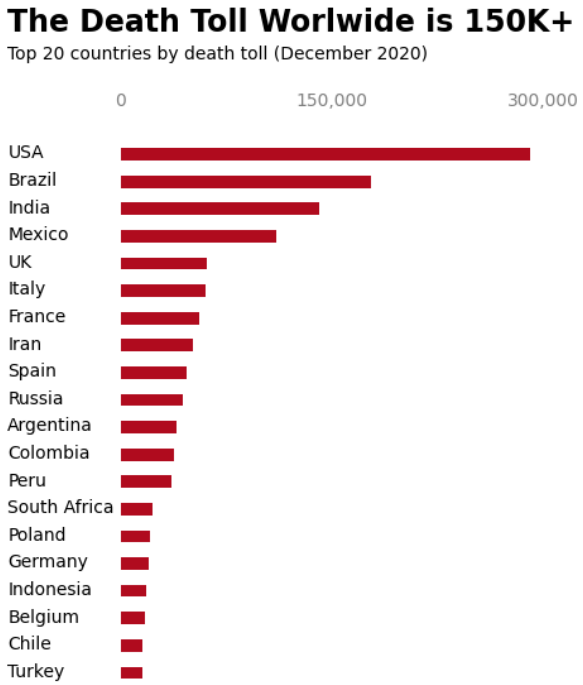
```
 'imshow',
 'in_axes',
 'indicate_inset',
 'indicate_inset_zoom',
 'inset_axes',
 'invert_xaxis',
 'invert_yaxis',
 'is_transform_set',
 'label_outer',
 'legend',
 'legend_',
 'lines',
 'locator_params',
 'loglog',
 'magnitude_spectrum',
 'margins',
 'matshow',
 'minorticks_off',
 'minorticks_on',
 'mouseover',
 'name',
 'patch',
 'patches',
 'pchanged',
 'pcolor',
 'pcolorfast',
 'pcolormesh',
 'phase_spectrum',
 'pick',
 'pickable',
 'pie',
 'plot',
 'plot_date',
 'properties',
 'psd',
 'quiver',
 'quiverkey',
 'redraw_in_frame',
 'relim',
 'remove',
 'remove_callback',
 'reset_position',
 'scatter',
 'secondary_xaxis',
 'secondary_yaxis',
 'semilogx',
 'semilogy',
 'set',
 'set_adjustable',
```

```
 'set_agg_filter',
 'set_alpha',
 'set_anchor',
 'set_animated',
 'set_aspect',
 'set_autoscale_on',
 'set_autoscalex_on',
 'set_autoscaley_on',
 'set_axes_locator',
 'set_axis_off',
 'set_axis_on',
 'set_axisbelow',
 'set_box_aspect',
 'set_clip_box',
 'set_clip_on',
 'set_clip_path',
 'set_facecolor',
 'set_fc',
 'set_figure',
 'set_frame_on',
 'set_gid',
 'set_in_layout',
 'set_label',
 'set_mouseover',
 'set_navigate',
 'set_navigate_mode',
 'set_path_effects',
 'set_picker',
 'set_position',
 'set_prop_cycle',
 'set_rasterization_zorder',
 'set_rasterized',
 'set_sketch_params',
 'set_snap',
 'set_subplotspec',
 'set_title',
 'set_transform',
 'set_url',
 'set_visible',
 'set_xbound',
 'set_xlabel',
 'set_xlim',
 'set_xmargin',
 'set_xscale',
 'set_xticklabels',
 'set_xticks',
 'set_ybound',
 'set_ylabel',
 'set_ylim',
```

```
 'set_ymargin',
 'set_yscale',
 'set_yticklabels',
 'set_yticks',
 'set_zorder',
 'sharex',
 'sharey',
 'specgram',
 'spines',
 'spy',
 'stackplot',
 'stairs',
 'stale',
 'stale_callback',
 'start_pan',
 'stem',
 'step',
 'sticky_edges',
 'streamplot',
 'table',
 'tables',
 'text',
 'texts',
 'tick_params',
 'ticklabel_format',
 'title',
 'titleOffsetTrans',
 'transAxes',
 'transData',
 'transLimits',
 'transScale',
 'tricontour',
 'tricontourf',
 'tripcolor',
 'triplot',
 'twinx',
 'twiny',
 'update',
 'update_datalim',
 'update_from',
 'use_sticky_edges',
 'viewLim',
 'violin',
 'violinplot',
 'vlines',
 'xaxis',
 'xaxis_date',
 'xaxis_inverted',
 'xcorr',
 'yaxis',
```

```
 'yaxis_date',
 'yaxis_inverted',
 'zorder']

fig, (ax1, ax2) = plt.subplots(nrows =1, ncols = 2, figsize = (11,6))

ax1.barh(death.Country_Other, death.Total_Deaths) # Object oriented
interface


ax2.barh(death.Country_Other, death.Total_Deaths, height = 0.45, color
= "#b00b1e") # Object oriented interface
ax2.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax2.spines.values():
    sp_value.set_visible(False)

ax2.set_xticks([0,150000,300000])
ax2.set_xticklabels(["0","150,000","300,000",])
ax2.set_yticklabels([])
ax2.xaxis.tick_top()

ax2.tick_params(top = False)
ax2.tick_params(axis = "x", colors ="grey")
ax2.text(x = -80000, y = 23.5, s ="The Death Toll Worlwide is 150K+",
size = 17, weight = "bold")  # plt.title("The Death Toll Worlwide is
1.5+")
ax2.text(x = -80000, y = 22.5, s ="Top 20 countries by death toll
(December 2020)", size = 10)

for i,c in enumerate(countries):
    ax2.text(x = -80000, y = i-0.15, s = c)
plt.subplots_adjust(wspace = 0.5)
plt.show()
```






```
fig, ax = plt.subplots(figsize = (4.5,6))

ax.barh(death.Country_Other, death.Total_Deaths, height = 0.45, color
= "#b00b1e") # Object oriented interface
ax.tick_params(bottom = False, left = False) # to remove ticks
for sp_value in ax.spines.values():
    sp_value.set_visible(False)

ax.set_xticks([0,150000,300000])
ax.set_xticklabels(["0","150,000","300,000",])
ax.set_yticklabels([])
ax.xaxis.tick_top()

ax.tick_params(top = False)
ax.tick_params(axis = "x", colors ="grey")
ax.text(x = -80000, y = 23.5, s ="The Death Toll Worlwide is 150K+",
size = 17, weight = "bold")  # plt.title("The Death Toll Worlwide is
1.5+")
ax.text(x = -80000, y = 22.5, s ="Top 20 countries by death toll
(December 2020)", size = 10)

for i,c in enumerate(countries):
    ax.text(x = -80000, y = i-0.15, s = c)
ax.axvline(150000, ymin = 0.045, ymax = 1, alpha = 0.5, c = "grey")
plt.show()
```

The Death Toll Worlwide is 150K+

Top 20 countries by death toll (December 2020)