In [75]:
```
!pip install pytrends matplotlib pandas seaborn plotly
```

```
Requirement already satisfied: pytrends in c:\users\admin\lib\site-packages
(4.9.2)
Requirement already satisfied: matplotlib in c:\users\admin\lib\site-packages
(3.10.3)
Requirement already satisfied: pandas in c:\users\admin\lib\site-packages
(2.3.1)
Requirement already satisfied: seaborn in c:\users\admin\lib\site-packages (0.1
3.2)
Requirement already satisfied: plotly in c:\users\admin\lib\site-packages
(6.2.0)
Requirement already satisfied: requests>=2.0 in c:\users\admin\lib\site-package
s (from pytrends) (2.32.4)
Requirement already satisfied: lxml in c:\users\admin\lib\site-packages (from p
ytrends) (6.0.0)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\admin\lib\site-pack
ages (from matplotlib) (1.3.2)
Requirement already satisfied: cycler>=0.10 in c:\users\admin\lib\site-packages
(from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\admin\lib\site-pac
kages (from matplotlib) (4.58.5)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\admin\lib\site-pac
kages (from matplotlib) (1.4.8)
Requirement already satisfied: numpy>=1.23 in c:\users\admin\lib\site-packages
(from matplotlib) (2.3.1)
Requirement already satisfied: packaging>=20.0 in c:\users\admin\lib\site-packa
ges (from matplotlib) (25.0)
Requirement already satisfied: pillow>=8 in c:\users\admin\lib\site-packages (f
rom matplotlib) (11.3.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\admin\lib\site-pack
ages (from matplotlib) (3.2.3)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\admin\lib\site-
packages (from matplotlib) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\admin\lib\site-packages
(from pandas) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in c:\users\admin\lib\site-packag
es (from pandas) (2025.2)
Requirement already satisfied: narwhals>=1.15.1 in c:\users\admin\lib\site-pack
ages (from plotly) (1.46.0)
Requirement already satisfied: six>=1.5 in c:\users\admin\lib\site-packages (fr
om python-dateutil>=2.7->matplotlib) (1.17.0)
Requirement already satisfied: charset_normalizer<4,>=2 in c:\users\admin\lib\s
ite-packages (from requests>=2.0->pytrends) (3.4.2)
Requirement already satisfied: idna<4,>=2.5 in c:\users\admin\lib\site-packages
(from requests>=2.0->pytrends) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\admin\lib\site-pa
ckages (from requests>=2.0->pytrends) (2.5.0)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\admin\lib\site-pa
ckages (from requests>=2.0->pytrends) (2025.7.9)
[notice] A new release of pip is available: 24.0 -> 25.1.1
[notice] To update, run: python.exe -m pip install --upgrade pip
```

In [76]:
```
import pandas as pd
```

```
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.io as pio
pio.renderers.default = 'notebook'  # or 'iframe' or 'svg'
import plotly.express as px
from pytrends.request import TrendReq
```

# 1) First, write a code where, by changing just the keyword, we can search for multiple things? - SET UP PYTREND LIBRARY AND DEFINE KEYWORD

In [77]:
```
pytrends = TrendReq(hl='en-US', tz=360)
keyword = "Data science"
```

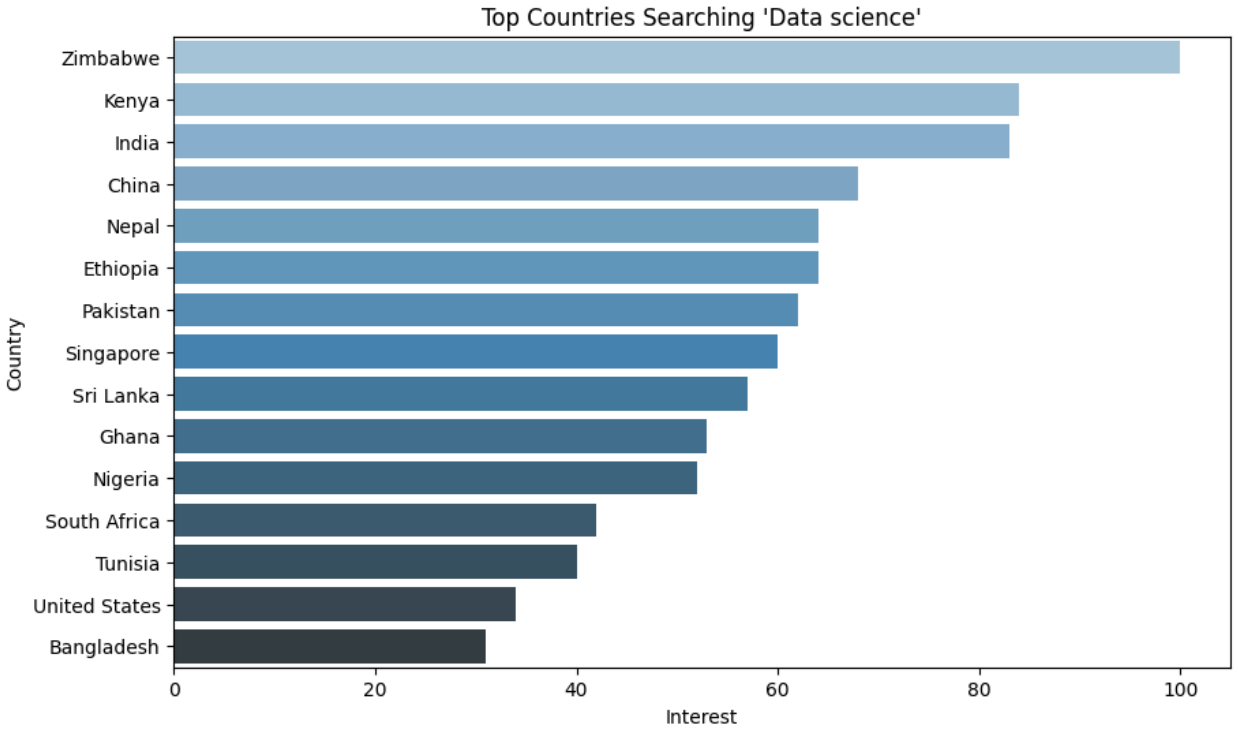
In [86]:
```
pytrends.build_payload([keyword], cat= 0, timeframe = 'today 12-m',geo='',gprc
```

# 2)Top 15 countries where the keywords are searched the most, and also create visual representation of it ?

In [79]:
```
region_data = pytrends.interest_by_region()
region_data = region_data.sort_values(by = keyword,ascending = False).head(15)
```

In [80]:
```
plot_df = pd.DataFrame({
    'Country': region_data.index,
    'Interest': region_data[keyword]
})

plt.figure(figsize=(10, 6))
sns.barplot(data=plot_df, x='Interest', y='Country', hue='Country', palette='B
plt.title(f"Top Countries Searching '{keyword}'")
plt.xlabel("Interest")
plt.ylabel("Country")
plt.show()
```



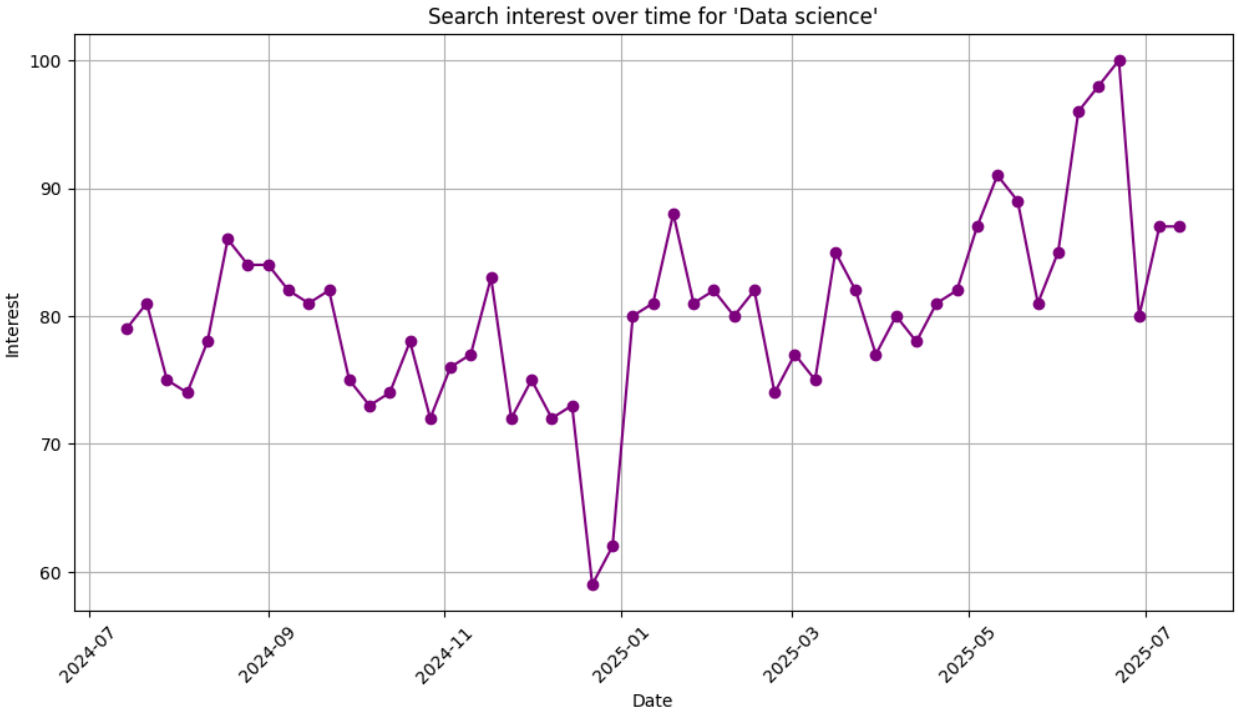

# 3) We need to extract the time-wise interest of the keyword — how it trended in different years ? -TIME WISE INTEREST

In [81]:
```
time_df =pytrends.interest_over_time()
```

```
C:\Users\Admin\Lib\site-packages\pytrends\request.py:260: FutureWarning:

Downcasting object dtype arrays on .fillna, .ffill, .bfill is deprecated and wi
ll change in a future version. Call result.infer_objects(copy=False) instead. T
o opt-in to the future behavior, set `pd.set_option('future.no_silent_downcasti
ng', True)`
```

In [82]:
```
plt.figure(figsize=(12,6))
plt.plot(time_df.index, time_df[keyword], marker='o', color='purple')
plt.title(f"Search interest over time for '{keyword}'")
plt.xlabel("Date")
plt.ylabel("Interest")
plt.grid(True)
plt.xticks(rotation=45)
plt.show()
```




# 4) Compare related keywords and plot the graph ? - MULTIPLE KEYWORDS COMPARISION

In [83]:
```
kw_list = ["cloud computing","Data science","Machine Learning"]
pytrends.build_payload(kw_list, cat=0 , timeframe = 'today 12-m',geo ='',gprop
```

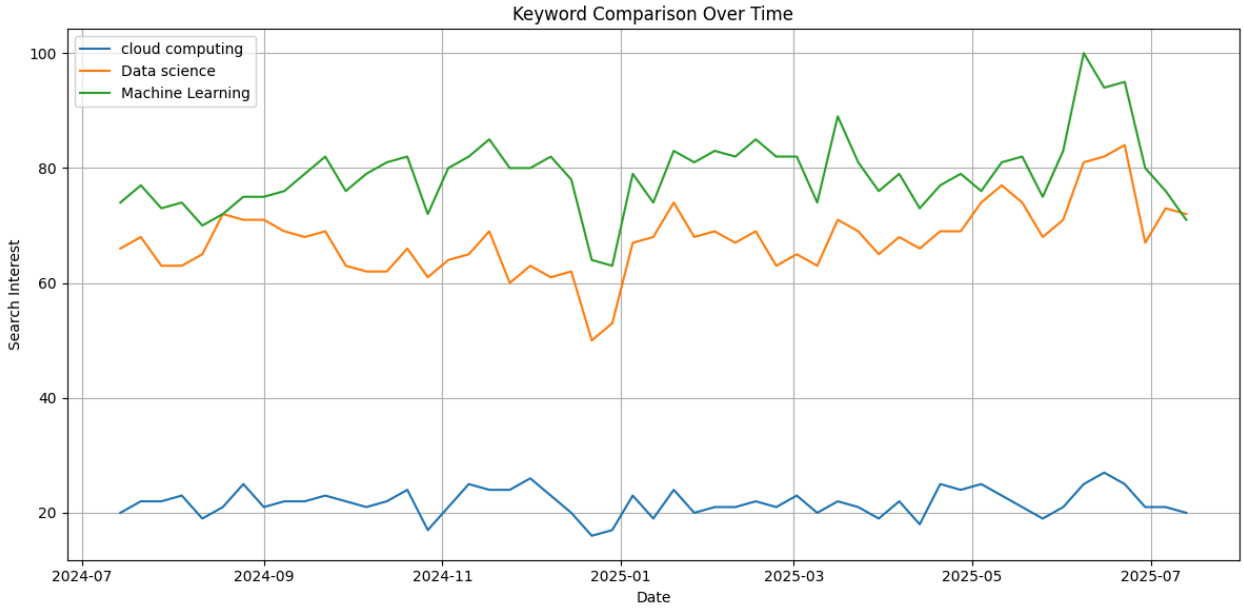
In [84]:
```
compare_df = pytrends.interest_over_time()

plt.figure(figsize=(12,6))
for kw in kw_list:
    plt.plot(compare_df.index, compare_df[kw], label=kw)

plt.title("Keyword Comparison Over Time")
plt.xlabel("Date")
plt.ylabel("Search Interest")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

```
C:\Users\Admin\Lib\site-packages\pytrends\request.py:260: FutureWarning:

Downcasting object dtype arrays on .fillna, .ffill, .bfill is deprecated and wi
ll change in a future version. Call result.infer_objects(copy=False) instead. T
o opt-in to the future behavior, set `pd.set_option('future.no_silent_downcasti
ng', True)`
```




# 5) A world map needs to be plotted showing the countries that search the keyword the most? - WORLD MAP SHOWING MAJOR COUNTRIES SEARCHING

In [85]:
```
region_data = region_data.reset_index()

fig = px.choropleth(region_data,
                    locations='geoName',
                    locationmode='country names',
                    color=keyword,
                    title=f"Search Interest for '{keyword}' by Country",
                    color_continuous_scale='Blues')
fig.show()
```

In [ ]:

In [ ]:

In [ ]: