In [6]:
```
pip install pytrends matplotlib pandas seaborn plotly
```

```
Requirement already satisfied: pytrends in c:\users\lenovo\anaconda3\lib\site-packages (4.9.2)Note: you may need to restart the kernel to use updated
packages.

Requirement already satisfied: matplotlib in c:\users\lenovo\anaconda3\lib\site-packages (3.7.1)
Requirement already satisfied: pandas in c:\users\lenovo\anaconda3\lib\site-packages (1.5.3)
Requirement already satisfied: seaborn in c:\users\lenovo\anaconda3\lib\site-packages (0.12.2)
Requirement already satisfied: plotly in c:\users\lenovo\anaconda3\lib\site-packages (5.9.0)
Requirement already satisfied: requests>=2.0 in c:\users\lenovo\anaconda3\lib\site-packages (from pytrends) (2.29.0)
Requirement already satisfied: lxml in c:\users\lenovo\anaconda3\lib\site-packages (from pytrends) (4.9.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\lenovo\anaconda3\lib\site-packages (from matplotlib) (1.0.5)
Requirement already satisfied: cycler>=0.10 in c:\users\lenovo\anaconda3\lib\site-packages (from matplotlib) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\lenovo\anaconda3\lib\site-packages (from matplotlib) (4.25.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\lenovo\anaconda3\lib\site-packages (from matplotlib) (1.4.4)
Requirement already satisfied: numpy>=1.20 in c:\users\lenovo\anaconda3\lib\site-packages (from matplotlib) (1.24.3)
Requirement already satisfied: packaging>=20.0 in c:\users\lenovo\anaconda3\lib\site-packages (from matplotlib) (23.0)
Requirement already satisfied: pillow>=6.2.0 in c:\users\lenovo\anaconda3\lib\site-packages (from matplotlib) (9.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\lenovo\anaconda3\lib\site-packages (from matplotlib) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\lenovo\anaconda3\lib\site-packages (from matplotlib) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\lenovo\anaconda3\lib\site-packages (from pandas) (2022.7)
Requirement already satisfied: tenacity>=6.2.0 in c:\users\lenovo\anaconda3\lib\site-packages (from plotly) (8.2.2)
Requirement already satisfied: six>=1.5 in c:\users\lenovo\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\lenovo\anaconda3\lib\site-packages (from requests>=2.0->pytrends) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in c:\users\lenovo\anaconda3\lib\site-packages (from requests>=2.0->pytrends) (3.4)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\lenovo\anaconda3\lib\site-packages (from requests>=2.0->pytrends) (1.26.16)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\lenovo\anaconda3\lib\site-packages (from requests>=2.0->pytrends) (2023.5.7)
```

In [13]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import plotly.express as px
import seaborn as sns
from pytrends.request import TrendReq
```

setup pytrend library and keyword define

In [14]:
```
pytrends = TrendReq(hl = 'en-us', tz = 360)
keyword = 'artificial intelligence'
```

Data Request

In [15]:
```
pytrends.build_payload([keyword], cat = 0, timeframe = 'today 12-m', geo ='', gprop ='')
```

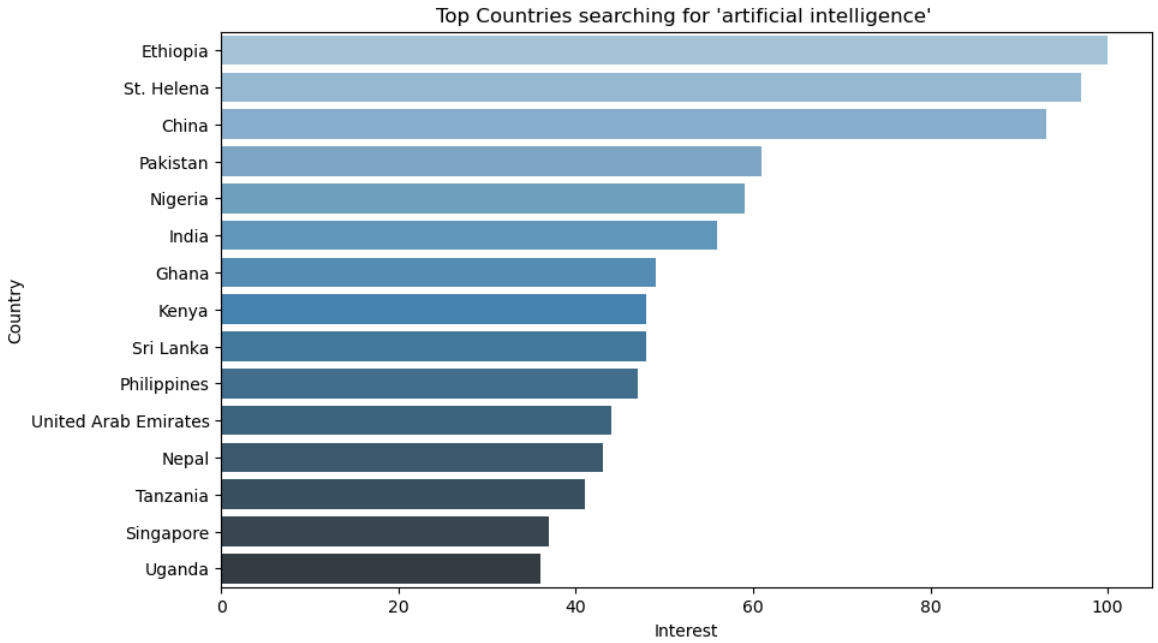
country wise interest

In [16]:
```
# 🔄 Handle Google Trends rate limiting
# Google may return a "TooManyRequestsError (HTTP 429)" if too many requests are made quickly.
# This loop safely retries the request after waiting 60 seconds,
# ensuring the notebook doesn't crash and data is fetched successfully.


import time
from pytrends.exceptions import TooManyRequestsError

while True:
    try:
        region_data = pytrends.interest_by_region()
        break
    except TooManyRequestsError:
        print("Too many requests — waiting 1 minute...")
        time.sleep(60)
```

```
Too many requests — waiting 1 minute...
Too many requests — waiting 1 minute...
```

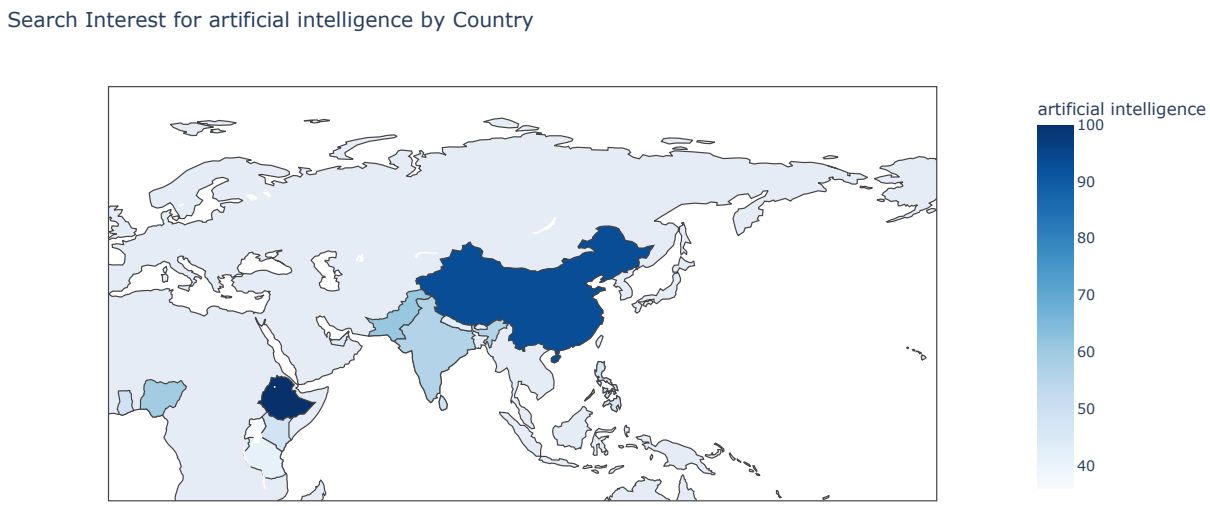
In [17]:
```
region_data = pytrends.interest_by_region()
region_data = region_data.sort_values(by = keyword,ascending = False).head(15)
```

In [18]:
```
plt.figure(figsize = (10,6))
#sns.barplot(x = region_data[keyword], y = region_data.index, palette = "Blues_d")
sns.barplot(x=region_data[keyword], y=region_data.index, palette="Blues_d", dodge=False)
plt.title(f"Top Countries searching for '{keyword}' ")
plt.xlabel("Interest")
plt.ylabel("Country")
plt.show()
```



World Map

In [19]:
```
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()
```



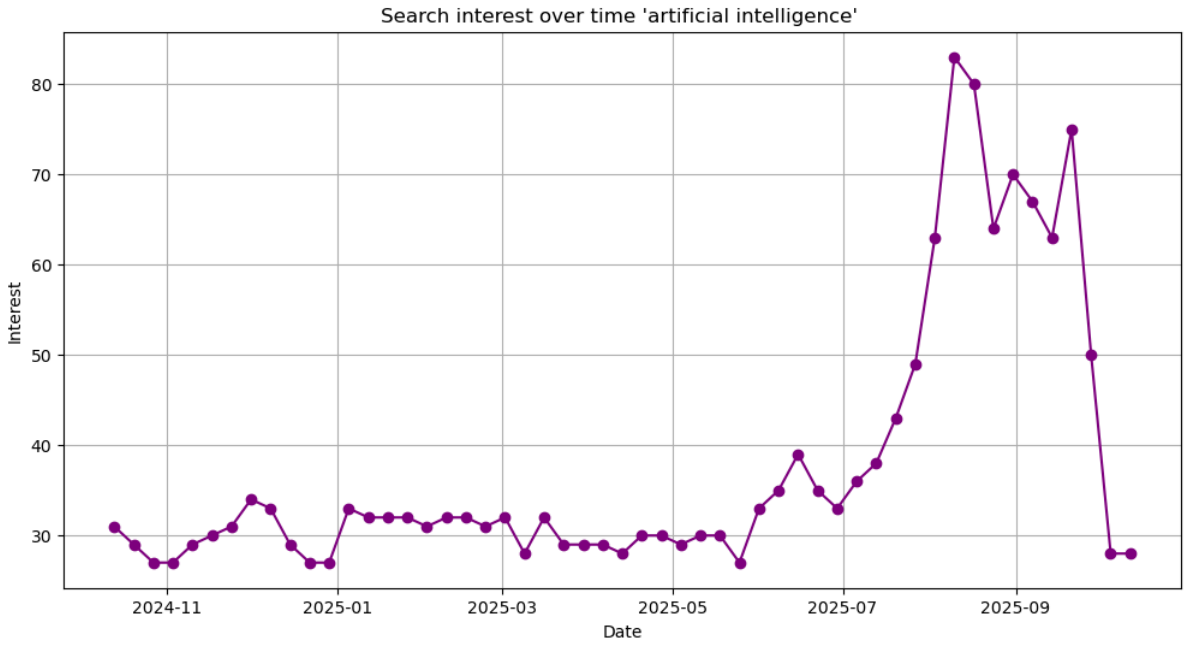
Time wise interest

In [31]:
```
import time
from pytrends.exceptions import TooManyRequestsError

while True:
    try:
        time_df = pytrends.interest_over_time()
        break
    except TooManyRequestsError:
        print("Too many requests — waiting 1 minute...")
        time.sleep(60)
```

In [32]:
```
time_df = pytrends.interest_over_time()
```

In [33]:
```
plt.figure(figsize=(12,6))
plt.plot(time_df.index, time_df[keyword], marker='o', color='purple')
plt.title(f"Search interest over time '{keyword}' ")
plt.xlabel("Date")
plt.ylabel("Interest")
```

```
plt.grid(True)
plt.show()
```



Multiple keywords compare

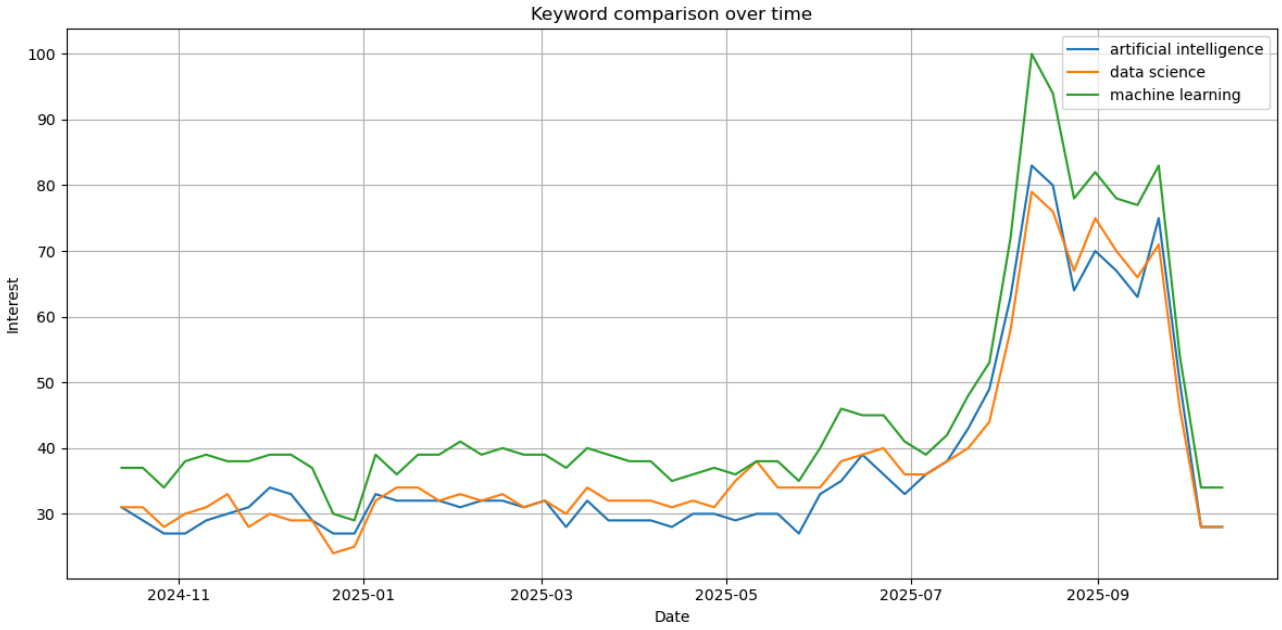
In [42]:
```
kw_list = ["artificial intelligence", "data science", "machine learning"]
pytrends.build_payload(kw_list, cat=0, timeframe='today 12-m', geo='', gprop ='')
```

In [43]:
```
from pytrends.exceptions import TooManyRequestsError
import requests
import time

while True:
    try:
        compare_df = pytrends.interest_over_time()
        break
    except TooManyRequestsError:
        print("Too many requests — waiting 60 seconds...")
        time.sleep(60)
    except requests.exceptions.ReadTimeout:
        print("Read timeout — waiting 30 seconds...")
        time.sleep(30)
    except requests.exceptions.ConnectionError:
        print("Connection error — retrying after 30 seconds...")
        time.sleep(30)
```

```
Read timeout — waiting 30 seconds...
Too many requests — waiting 60 seconds...
Too many requests — waiting 60 seconds...
Too many requests — waiting 60 seconds...
Too many requests — waiting 60 seconds...
```

In [44]:
```
compare_df = pytrends.interest_over_time()
```

In [45]:
```
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("Interest")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```


Keyword comparison over time
artificial intelligence
data science
machine learning
Interest
Date
100
90
80
70
60
50
40
30
2024-11
2025-01
2025-03
2025-05
2025-07
2025-09