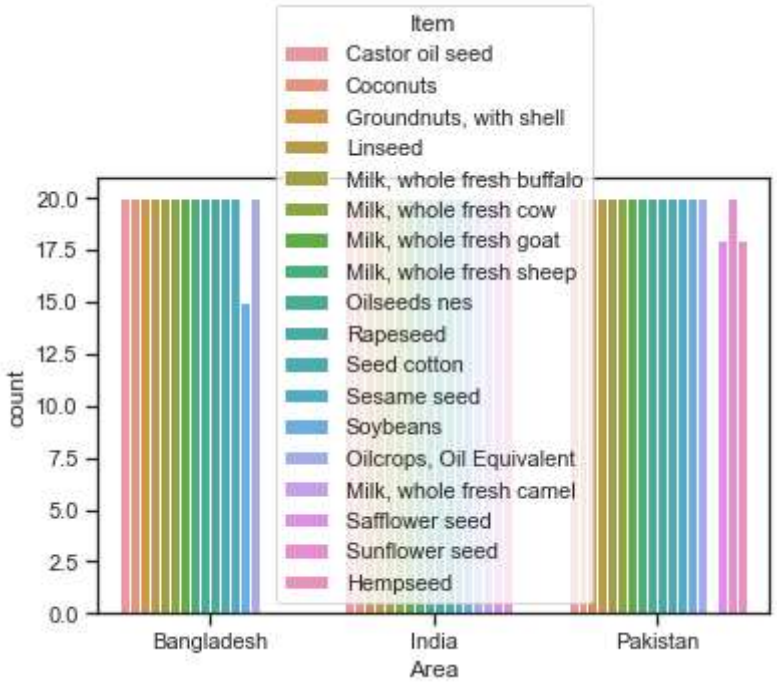
In [1]:
```
#Import librairies
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [35]:
```
#import data from file
chilla=pd.read_csv("FnAgri_data.csv")
chilla.head()
```

Out[35]:

	Domain Code	Domain	Area Code (FAO)	Area	Element Code	Element	Item Code (FAO)	Item	Year Code	Year	Unit	Valu
0	QI	Production Indices	16	Bangladesh	434	Gross per capita Production Index Number (2014...	265	Castor oil seed	2000	2000	index	130.
1	QI	Production Indices	16	Bangladesh	434	Gross per capita Production Index Number (2014...	265	Castor oil seed	2001	2001	index	123.
2	QI	Production Indices	16	Bangladesh	434	Gross per capita Production Index Number (2014...	265	Castor oil seed	2002	2002	index	120.
3	QI	Production Indices	16	Bangladesh	434	Gross per capita Production Index Number (2014...	265	Castor oil seed	2003	2003	index	127.
4	QI	Production Indices	16	Bangladesh	434	Gross per capita Production Index Number (2014...	265	Castor oil seed	2004	2004	index	129.

In [27]:
```
sns.set_theme(style="ticks", color_codes=True)
#p=sns.lineplot(x="Gender",hue="Age",data=chilla)
#p=sns.countplot(x="Qualification_completed",hue="field_of_study",data=chilla)
p=sns.countplot(x="Area",hue="Item",data=chilla)
plt.show()
```

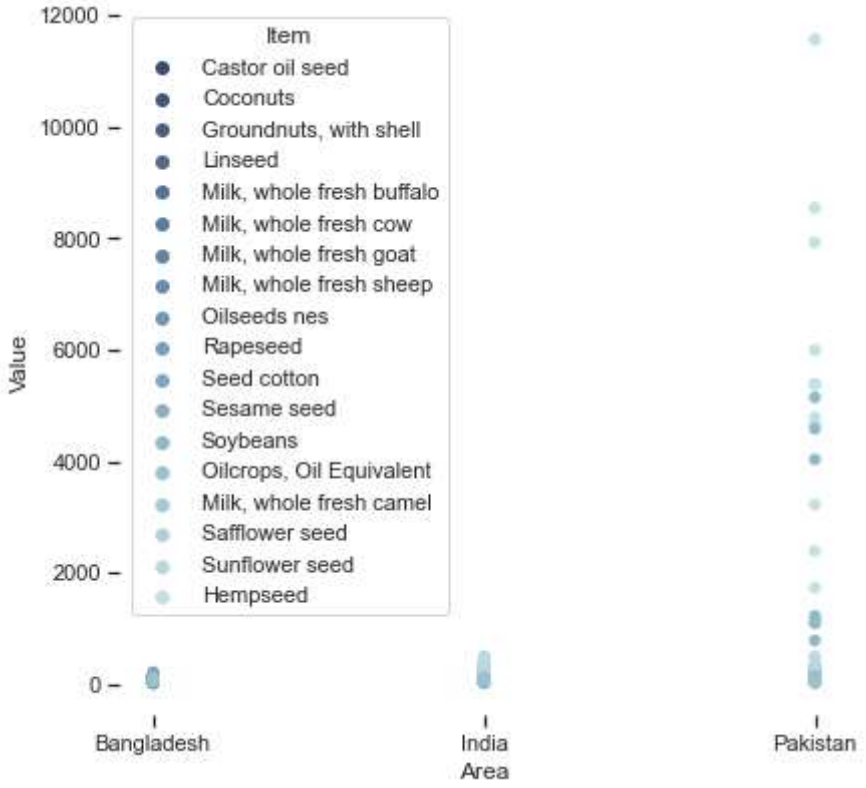


In [4]:

```
# Draw a scatter plot while assigning point colors and sizes to different
# variables in the dataset
f, ax = plt.subplots(figsize=(6.5, 6.5))
sns.despine(f, left=True, bottom=True)
sns.scatterplot(x="Area", y="Value",hue="Item",
                palette="ch:r=-.2,d=.3_r",
                sizes=(1, 8), linewidth=0,
                data=chilla, ax=ax)
```

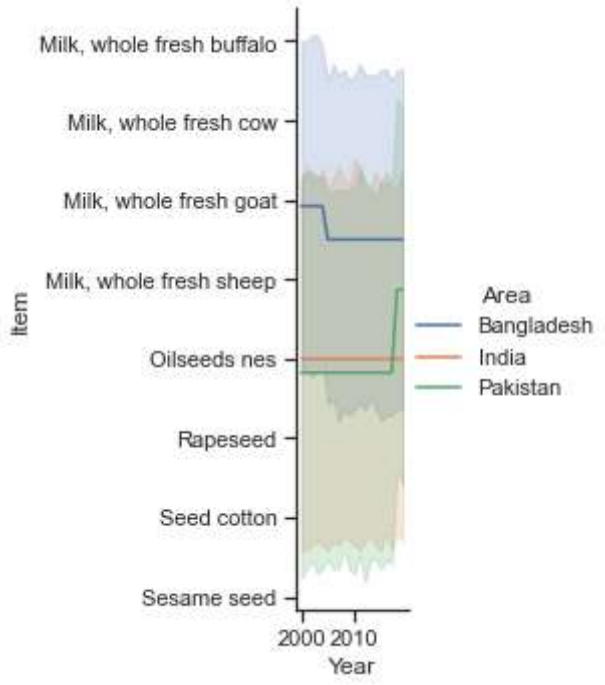
Out[4]:

```
<AxesSubplot:xlabel='Area', ylabel='Value'>
```

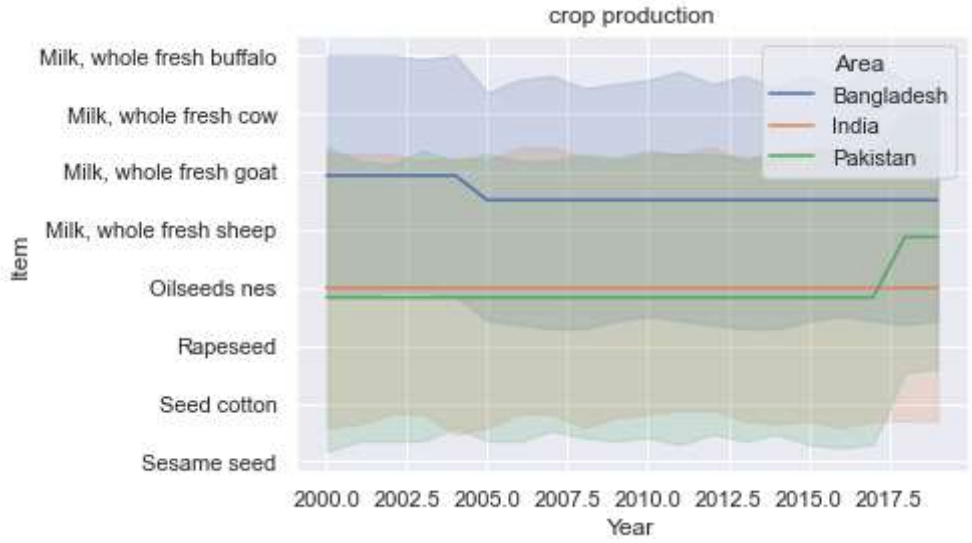


In [5]:
```
sns.set_theme(style="ticks")

# Define the palette as a list to specify exact values
#palette = sns.color_palette("viridis")

# Plot the lines on two facets
sns.relplot(
    data=chilla,
    x="Year", y="Item",
    hue="Area",
    kind="line", size_order=["T1", "T2"],
    height=5, aspect=.75, facet_kws=dict(sharex=False),
)
```

Out[5]:
```
<seaborn.axisgrid.FacetGrid at 0x261c9a2fa00>
```

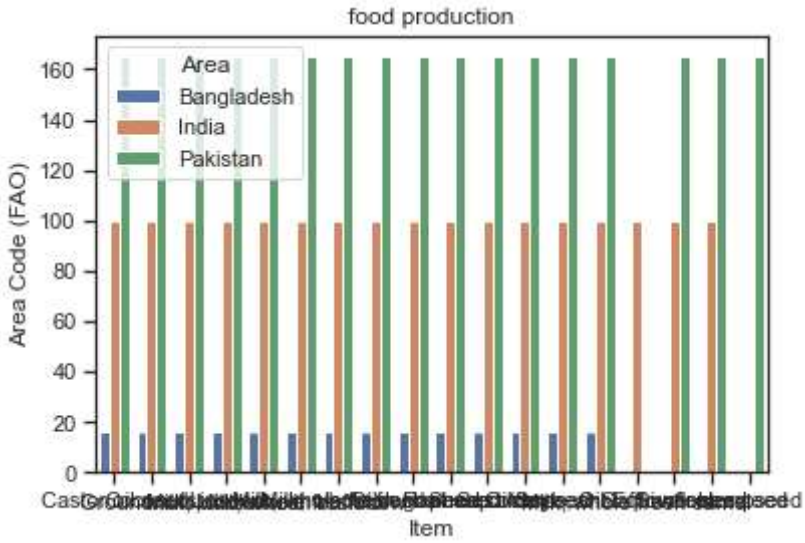


In [13]:
```
import seaborn as sns
sns.set_theme(style="darkgrid")
# Plot the responses for different events and regions
sns.lineplot(x="Year", y="Item",
             hue="Area",
             data=chilla)
sns.set_style("dark")
plt.title("food production") # add the title
plt.show()
```



In [30]:

```
#draw a barplot
sns.barplot(x="Item",y="Area Code (FAO)", hue="Area", data=chilla)

plt.title("food production")
plt.figure(figsize=(10,2))
# plt.xlim(2) #to put limit on ais
# plt.ylim(1)

plt.show()
```




```
<Figure size 720x144 with 0 Axes>
```

In [36]:

```
#import data from file for cereals production in pakistan
baba=pd.read_csv("cereals_prd.csv")
baba.head()
```

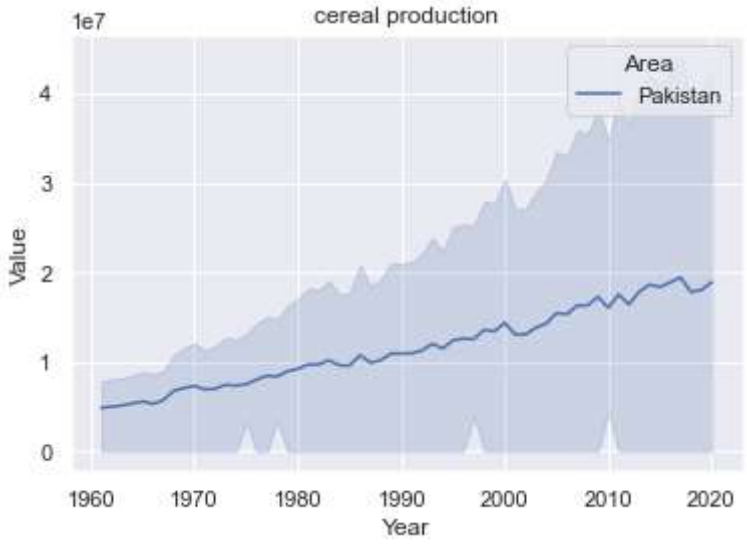
Out[36]:

Domain Code	Domain	Area Code	Area	Element Code	Element	Item Code	Item	Year Code	Year	Unit	Value

	Domain Code	Domain	Area Code	Area	Element Code	Element	Item Code	Item	Year Code	Year	Unit	Value
0	QCL	Crops and livestock products	165	Pakistan	5312	Area harvested	1717	Cereals, Total	1961	1961	ha	7858558
1	QCL	Crops and livestock products	165	Pakistan	5419	Yield	1717	Cereals, Total	1961	1961	hg/ha	8564
2	QCL	Crops and livestock products	165	Pakistan	5510	Production	1717	Cereals, Total	1961	1961	tonnes	6729680
3	QCL	Crops and livestock products	165	Pakistan	5312	Area harvested	1717	Cereals, Total	1962	1962	ha	8090856
4	QCL	Crops and livestock products	165	Pakistan	5419	Yield	1717	Cereals, Total	1962	1962	hg/ha	8580

In [38]:

```
import seaborn as sns
sns.set_theme(style="darkgrid")
# Plot the responses for different events and regions
sns.lineplot(x="Year", y="Value",
             hue="Area",color="red",
             data=baba)
sns.set_style("dark")
plt.title("cereal production") # add the title
plt.show()
```



In [37]:
```
baba.describe()
```

Out[37]:

	Area Code	Element Code	Item Code	Year Code	Year	Value
count	180.0	180.000000	180.0	180.000000	180.000000	1.800000e+02
mean	165.0	5413.666667	1717.0	1990.500000	1990.500000	1.161077e+07
std	0.0	81.146808	0.0	17.366409	17.366409	1.157594e+07
min	165.0	5312.000000	1717.0	1961.000000	1961.000000	8.400000e+03
25%	165.0	5312.000000	1717.0	1975.750000	1975.750000	2.596750e+04
50%	165.0	5419.000000	1717.0	1990.500000	1990.500000	1.104450e+07
75%	165.0	5510.000000	1717.0	2005.250000	2005.250000	1.413153e+07
max	165.0	5510.000000	1717.0	2020.000000	2020.000000	4.430000e+07

In [1]:
```
#Import librairies
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [2]:
```
#import data from file
chilla=pd.read_csv("emp.csv")
chilla.head()
```

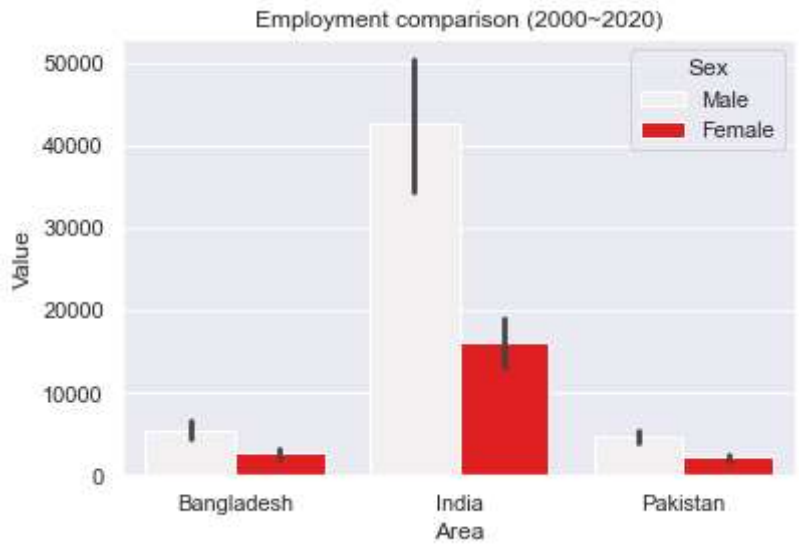
Out[2]:

	Domain Code	Domain	Area Code (FAO)	Area	Indicator Code	Indicator	Sex Code	Sex	Year Code	Year	Source Code	
0	OE	Employment Indicators	16	Bangladesh	21155	Share of employment in agriculture, forestry a...	21	Male	2000	2000	3023	
1	OE	Employment Indicators	16	Bangladesh	21156	Share of employment in agriculture, forestry a...	21	Male	2000	2000	3043	I I
2	OE	Employment Indicators	16	Bangladesh	21156	Share of employment in agriculture, forestry a...	21	Male	2001	2001	3043	I I
3	OE	Employment Indicators	16	Bangladesh	21156	Share of employment in agriculture, forestry a...	21	Male	2002	2002	3043	I I

	Domain Code	Domain	Area Code (FAO)	Area	Indicator Code	Indicator	Sex Code	Sex	Year Code	Year	Source Code
4	OE	Employment Indicators	16	Bangladesh	21155	Share of employment in agriculture, forestry a...	21	Male	2003	2003	3023

In [4]:

```
import seaborn as sns
sns.set_theme(style="darkgrid")
# Plot the responses for different events and regions
sns.barplot(x="Area", y="Value",
            hue="Sex",color="red",
            data=chilla)
sns.set_style("dark")
plt.title("Employment comparison (2000~2020)") # add the title
plt.show()
```



In [ ]: