

```
In [1]: #import libraries
import numpy as np
import pandas as pd
import plotly.express as px
import matplotlib.pyplot as plt
```

```
In [2]: #Load zip file and extract all files
from zipfile import ZipFile
file_name = "kiva_data.zip"
with ZipFile(file_name, "r") as zip:
    zip.printdir()
    zip.extractall()
```

File Name	Modified	Size
kiva_loans.csv	2019-09-27 06:19:48	195852823
kiva_mpi_region_locations.csv	2019-09-27 06:20:02	177359
loan_theme_ids.csv	2019-09-27 06:20:02	31641314
loan_themes_by_region.csv	2019-09-27 06:20:04	5055066

```
In [3]: #Load dataset
loans = pd.read_csv("kiva_loans.csv")
print(loans.shape)
print(loans.info())
loans.head()
```

```
(671205, 20)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 671205 entries, 0 to 671204
Data columns (total 20 columns):
#   Column              Non-Null Count  Dtype
---  -
0   id                   671205 non-null  int64
1   funded_amount       671205 non-null  float64
2   loan_amount         671205 non-null  float64
3   activity            671205 non-null  object
4   sector              671205 non-null  object
5   use                 666973 non-null  object
6   country_code        671197 non-null  object
7   country             671205 non-null  object
8   region              614405 non-null  object
9   currency            671205 non-null  object
10  partner_id          657698 non-null  float64
11  posted_time         671205 non-null  object
12  disbursed_time      668809 non-null  object
13  funded_time         622874 non-null  object
14  term_in_months      671205 non-null  float64
15  lender_count        671205 non-null  int64
16  tags                499789 non-null  object
17  borrower_genders    666984 non-null  object
18  repayment_interval  671205 non-null  object
19  date                671205 non-null  object
dtypes: float64(4), int64(2), object(14)
memory usage: 102.4+ MB
None
```

Out[3]:

	id	funded_amount	loan_amount	activity	sector	use	country_code	country	region	current
0	653051	300.0	300.0	Fruits & Vegetables	Food	To buy seasonal, fresh fruits to sell.	PK	Pakistan	Lahore	Pl
1	653053	575.0	575.0	Rickshaw	Transportation	to repair and maintain the auto rickshaw used ...	PK	Pakistan	Lahore	Pl
2	653068	150.0	150.0	Transportation	Transportation	To repair their old cycle-van and buy another ...	IN	India	Maynaguri	IN
3	653063	200.0	200.0	Embroidery	Arts	to purchase an embroidery machine and a variet...	PK	Pakistan	Lahore	Pl
4	653084	400.0	400.0	Milk Sales	Food	to purchase one buffalo.	PK	Pakistan	Abdul Hakeem	Pl

COUNTRY

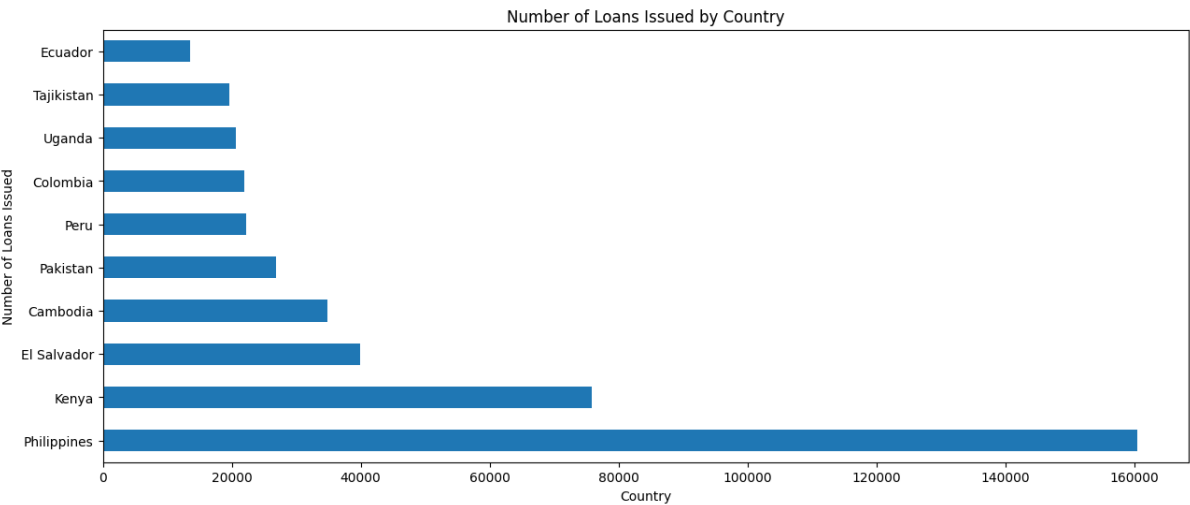
In [4]:

```
#create figure and axis
fig, ax = plt.subplots(figsize=(15,6))

#plot data
loans_by_country = loans["country"].value_counts().head(10)

loans_by_country.plot(kind="barh")
ax.set_xlabel("Country")
ax.set_ylabel("Number of Loans Issued")
ax.set_title("Number of Loans Issued by Country")

#display plot
plt.show();
```



PHILIPPINES

In [5]:

```
df_philippines = loans[loans["country"] == "Philippines"]
df_philippines["sector"].value_counts()
```

```
Out[5]: Retail      53581
      Food      42700
      Agriculture  36776
      Housing     6785
      Transportation 4568
      Services    4479
      Clothing    2761
      Arts        1990
      Personal Use 1943
      Education   1772
      Manufacturing 1444
      Construction 798
      Health      631
      Entertainment 139
      Wholesale   74
      Name: sector, dtype: int64
```

1. RETAIL

```
In [6]: df_philippines = df_philippines[df_philippines["sector"] == "Retail"]
      print(df_philippines["activity"].value_counts().head())
      df_philippines["use"].value_counts().head()
```

```
General Store      42960
Retail             3873
Personal Products Sales 2125
Charcoal Sales     1678
Fuel/Firewood      795
      Name: activity, dtype: int64
```

```
Out[6]: to purchase more groceries to sell.      1380
      to buy additional stocks of groceries to sell.      877
      to buy more groceries to sell.      827
      to buy additional items to sell in her general store 429
      to buy more stocks of groceries to sell.      399
      Name: use, dtype: int64
```

2. FOOD

```
In [7]: df_philippines1 = loans[loans["country"] == "Philippines"]
      df_philippines1 = df_philippines1[df_philippines1["sector"] == "Food"]
      print(df_philippines1["activity"].value_counts().head())
      df_philippines1["use"].value_counts().head()
```

```
Fish Selling      9060
Fishing           8682
Food Production/Sales 8675
Fruits & Vegetables 4486
Food              2959
      Name: activity, dtype: int64
```

```
Out[7]: to buy ingredients for her food production business 1419
      to buy fishing gear and other equipment.      687
      to buy ingredients for her food production business. 666
      to buy ingredients for her food vending business. 485
      to buy more fish to sell.      392
      Name: use, dtype: int64
```

3. AGRICULTURE

```
In [8]: df_philippines2 = loans[loans["country"] == "Philippines"]
      df_philippines2 = df_philippines2[df_philippines2["sector"] == "Agriculture"]
      print(df_philippines2["activity"].value_counts().head())
      df_philippines2["use"].value_counts().head()
```

```
Pigs      19985
Farming   11924
Livestock  1718
Agriculture 1298
Poultry    544
      Name: activity, dtype: int64
```

```
Out[8]: to buy feed and other supplies to raise her pigs.    1402
        to buy fertilizers and other farm supplies.          1272
        to buy feed and vitamins for her pigs.              1153
        to buy feeds and other supplies to raise her pigs    1027
        to buy fertilizers and other farm supplies            964
        Name: use, dtype: int64
```

KENYA

```
In [9]: df_kenya = loans[loans["country"] == "Kenya"]
df_kenya["sector"].value_counts()
```

```
Out[9]: Agriculture      33644
        Food             14072
        Retail           10185
        Services          5460
        Clothing          4753
        Personal Use      2130
        Transportation     1843
        Education         1329
        Construction       797
        Health             631
        Arts              347
        Manufacturing      295
        Housing           286
        Entertainment       32
        Wholesale          21
        Name: sector, dtype: int64
```

1. AGRICULTURE

```
In [10]: df_kenya = df_kenya[df_kenya["sector"] == "Agriculture"]
print(df_kenya["activity"].value_counts().head())
df_kenya["use"].value_counts().head()
```

```
Farming      20555
Agriculture  5244
Dairy        3983
Poultry      2340
Livestock    548
Name: activity, dtype: int64
```

[illegible]

2. FOOD

```
In [11]: df_kenya1 = loans[loans["country"] == "Kenya"]
df_kenya1 = df_kenya1[df_kenya1["sector"] == "Food"]
print(df_kenya1["activity"].value_counts().head())
df_kenya1["use"].value_counts().head()
```

```
Fruits & Vegetables    2989
Cereals                2647
Grocery Store          2578
Food Stall             1272
Food                   1091
Name: activity, dtype: int64
```

```
Out[11]: to buy cereals. 437
to purchase bundles of wheat flour, maize flour, tea leaves, and cooking fat 398
to purchase green vegetables, two crates of tomatoes, and onions for resale. 212
to purchase stock of sugar, rice, maize flour and wheat flour 181
to purchase green vegetables, 2 crates of tomatoes, and onions for resale 158
Name: use, dtype: int64
```

3. RETAIL

```
In [12]: df_kenya2 = loans[loans["country"] == "Kenya"]
df_kenya2 = df_kenya2[df_kenya2["sector"] == "Retail"]
print(df_kenya2["activity"].value_counts().head())
df_kenya2["use"].value_counts().head()
```

General Store	3830
Retail	2918
Charcoal Sales	924
Cosmetics Sales	721
Shoe Sales	293

Name: activity, dtype: int64

```
Out[12]: to purchase stock of sugar, rice, maize flour and wheat flour    239
to purchase sacks of charcoal for resale                            146
to purchase stock of sugar, rice, maize flour and wheat flour.      83
to buy more stock of charcoal.                                       73
to purchase sacks of charcoal for resale.                            63
Name: use, dtype: int64
```

EL SALVADOR

```
In [13]: df_el_salvador = loans[loans["country"] == "El Salvador"]
df_el_salvador["sector"].value_counts()
```

```
Out[13]: Agriculture    14009
Food                  8316
Housing               5977
Retail                5670
Clothing              1906
Services              1637
Arts                  554
Manufacturing         497
Transportation        363
Personal Use          338
Health                280
Construction          195
Education             111
Entertainment         19
Wholesale              3
Name: sector, dtype: int64
```

1. AGRICULTURE

```
In [14]: df_el_salvador = df_el_salvador[df_el_salvador["sector"] == "Agriculture"]
print(df_el_salvador["activity"].value_counts().head())
df_el_salvador["use"].value_counts().head()
```

Agriculture	4554
Farming	2939
Livestock	1464
Cattle	1419
Farm Supplies	1328

Name: activity, dtype: int64

```
Out[14]: to buy a cow.    68
to buy cattle.    64
to buy farming supplies.    51
to buy a heifer.    47
to buy a dairy cow.    41
Name: use, dtype: int64
```

2. FOOD

```
In [15]: df_el_salvador1 = loans[loans["country"] == "El Salvador"]
df_el_salvador1 = df_el_salvador1[df_el_salvador1["sector"] == "Food"]
print(df_el_salvador1["activity"].value_counts().head())
df_el_salvador1["use"].value_counts().head()
```

```
Food Production/Sales      4318
Fruits & Vegetables        992
Bakery                     606
Food Stall                 388
Fishing                    366
Name: activity, dtype: int64
```

```
Out[15]: to buy ingredients.                23
         to buy fishing nets.            15
         to buy corn by the quintal and lime. 15
         to buy more fruit and vegetables wholesale. 12
         to buy fruits and vegetables.       12
         Name: use, dtype: int64
```

3. HOUSING

```
In [16]: df_el_salvador2 = loans[loans["country"] == "El Salvador"]
         df_el_salvador2 = df_el_salvador2[df_el_salvador2["sector"] == "Housing"]
         print(df_el_salvador2["activity"].value_counts().head())
         df_el_salvador2["use"].value_counts().head()
```

```
Personal Housing Expenses  5925
Property                  52
Name: activity, dtype: int64
```

```
Out[16]: to buy construction materials.          24
         to buy necessary construction materials.  23
         to buy building materials.              15
         to purchase the necessary construction materials. 11
         to make home improvements.              8
         Name: use, dtype: int64
```

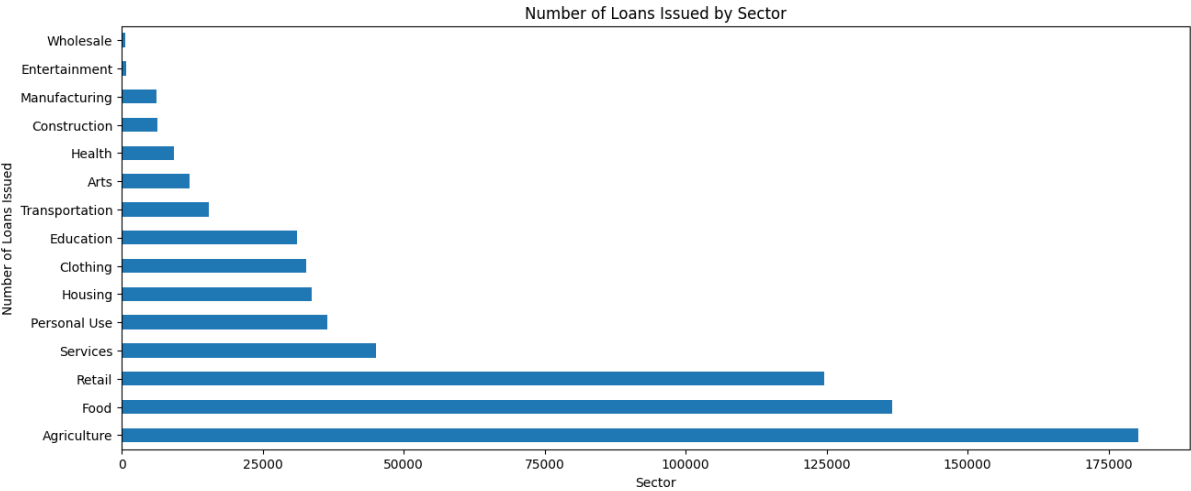
SECTOR

```
In [17]: #create figure and axis
         fig, ax = plt.subplots(figsize=(15,6))

         #plot data
         loans_by_sector = loans["sector"].value_counts()

         loans_by_sector.plot(kind="barh")
         ax.set_xlabel("Sector")
         ax.set_ylabel("Number of Loans Issued")
         ax.set_title("Number of Loans Issued by Sector")

         #display plot
         plt.show();
```



ACTIVITY

```
In [18]: #create figure and axis
         fig, ax = plt.subplots(figsize=(15,6))

         #plot data
```

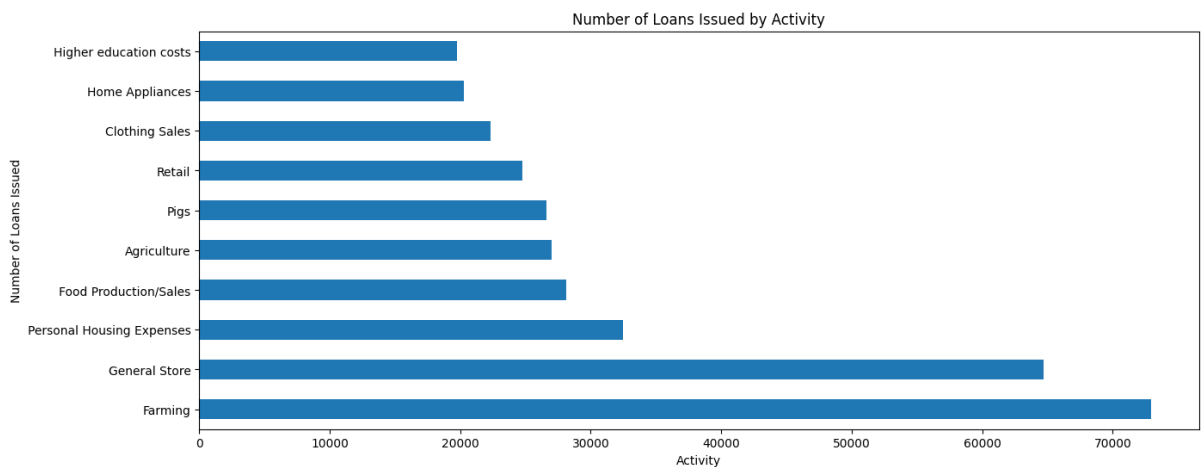
```

loans_by_activity = loans["activity"].value_counts().head(10)

loans_by_activity.plot(kind="barh")
ax.set_xlabel("Activity")
ax.set_ylabel("Number of Loans Issued")
ax.set_title("Number of Loans Issued by Activity")

#display plot
plt.show();

```



LOAN DISBURSED BY KIVA

```

In [19]: mean_funded_amount_by_country = loans.groupby("country")["funded_amount"].mean().sort_values(ascending=False)
mean_funded_amount_by_country

```

```

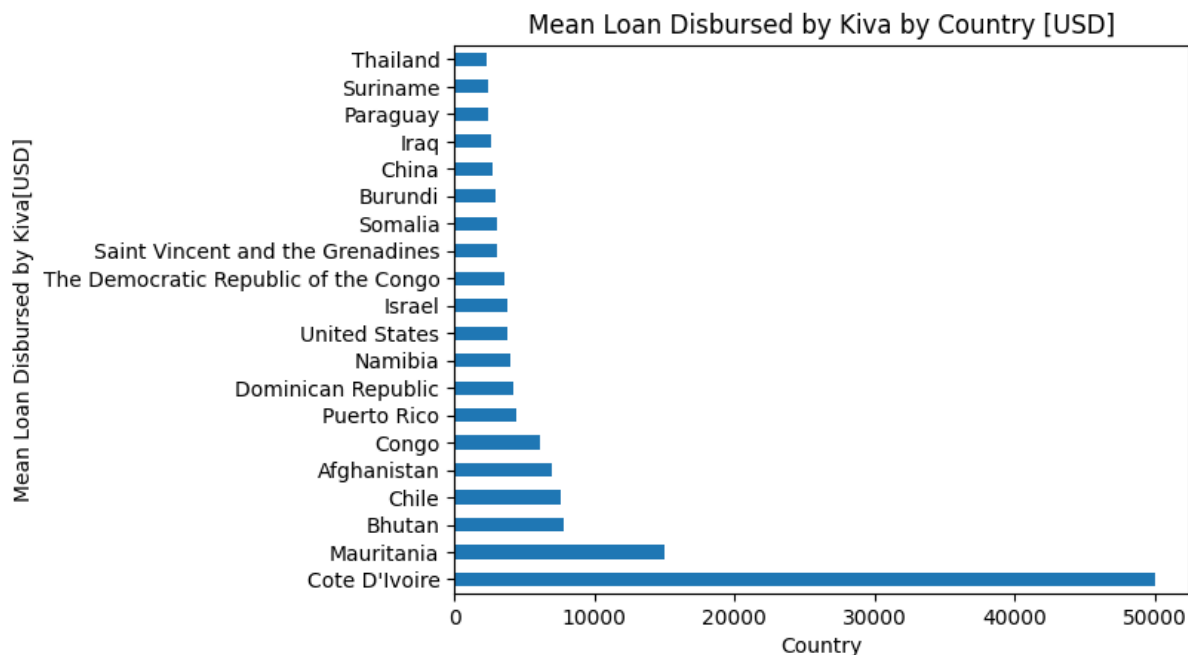
Out[19]: country
Cote D'Ivoire      50000.000000
Mauritania         15000.000000
Bhutan             7812.500000
Chile              7625.000000
Afghanistan        7000.000000
Congo              6142.578125
Puerto Rico       4409.191176
Dominican Republic 4200.604839
Namibia            4046.875000
United States      3800.843591
Israel             3786.578947
The Democratic Republic of the Congo 3586.161731
Saint Vincent and the Grenadines 3076.562500
Somalia            3011.666667
Burundi           2907.443182
China              2787.126866
Iraq               2629.632427
Paraguay           2471.032513
Suriname           2423.654709
Thailand           2351.944444
Name: funded_amount, dtype: float64

```

```

In [20]: mean_funded_amount_by_country.plot(
    kind="barh",
    xlabel="Country",
    ylabel="Mean Loan Disbursed by Kiva[USD]",
    title="Mean Loan Disbursed by Kiva by Country [USD]");

```



COTE D'IVOIRE

```
In [21]: df_cote_divoire = loans[loans["country"] == "Cote D'Ivoire"]
print(df_cote_divoire["sector"].value_counts().head())
print(df_cote_divoire["activity"].value_counts().head())
df_cote_divoire["use"].value_counts().head()
```

```
Agriculture    1
Name: sector, dtype: int64
Agriculture    1
Name: activity, dtype: int64
```

```
Out[21]: double cashew nut export output and hire about 200 new workers.    1
Name: use, dtype: int64
```

MAURITANIA

```
In [22]: df_mauritania = loans[loans["country"] == "Mauritania"]
print(df_mauritania["sector"].value_counts().head())
print(df_mauritania["activity"].value_counts().head())
df_mauritania["use"].value_counts().head()
```

```
Services    1
Name: sector, dtype: int64
Water Distribution    1
Name: activity, dtype: int64
```

```
Out[22]: to replace diesel power generators with solar plants for uninterrupted access to clean water.\r\n \r\n
1
Name: use, dtype: int64
```

BHUTAN

```
In [23]: df_bhutan = loans[loans["country"] == "Bhutan"]
print(df_bhutan["sector"].value_counts().head())
print(df_bhutan["activity"].value_counts().head())
df_bhutan["use"].value_counts().head()
```

```
Arts    2
Name: sector, dtype: int64
Weaving    2
Name: activity, dtype: int64
```

```
Out[23]: to develop a new product which will help create additional income opportunities to Bhutanese weavers.
1
Name: use, dtype: int64
```


LOCATION DATA

```
In [24]: regions = pd.read_csv("kiva_mpi_region_locations.csv")
print(regions.shape)
print(regions.info())
regions.head()
```

(2772, 9)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2772 entries, 0 to 2771
Data columns (total 9 columns):
Column Non-Null Count Dtype
--- ---
0 LocationName 984 non-null object
1 ISO 1008 non-null object
2 country 1008 non-null object
3 region 984 non-null object
4 world_region 1008 non-null object
5 MPI 984 non-null float64
6 geo 2772 non-null object
7 lat 892 non-null float64
8 lon 892 non-null float64
dtypes: float64(3), object(6)
memory usage: 195.0+ KB
None

Out[24]:

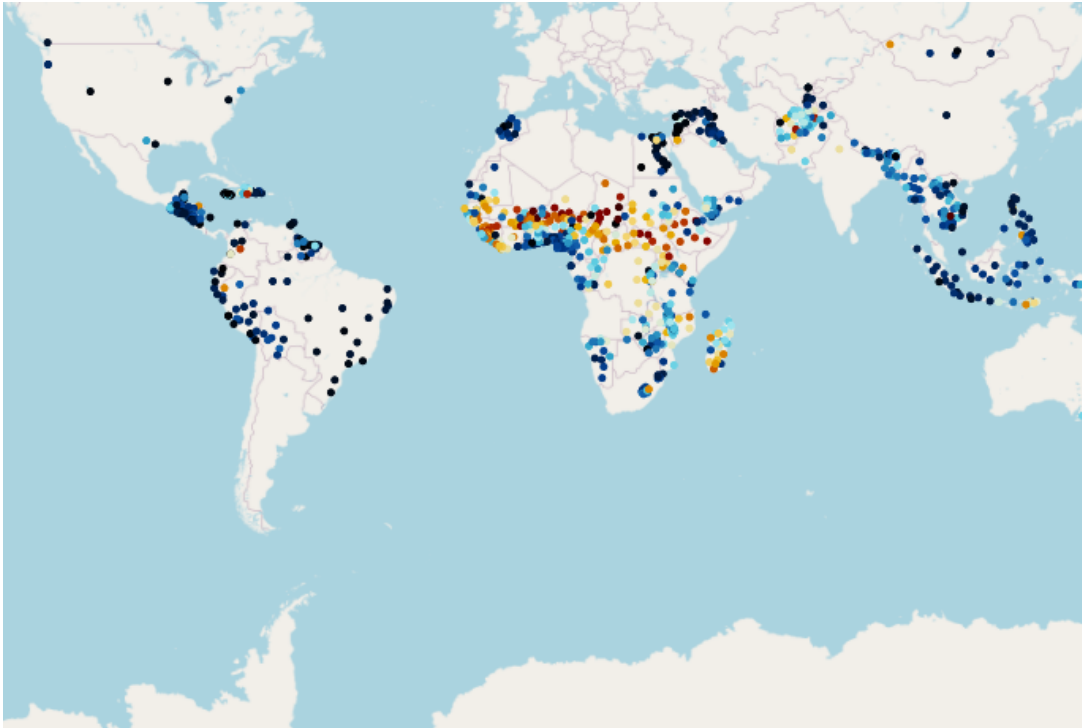
	LocationName	ISO	country	region	world_region	MPI	geo	lat	lon
0	Badakhshan, Afghanistan	AFG	Afghanistan	Badakhshan	South Asia	0.387	(36.7347725, 70.81199529999999)	36.734772	70.811995
1	Badghis, Afghanistan	AFG	Afghanistan	Badghis	South Asia	0.466	(35.1671339, 63.7695384)	35.167134	63.769538
2	Baghlan, Afghanistan	AFG	Afghanistan	Baghlan	South Asia	0.300	(35.8042947, 69.2877535)	35.804295	69.287754
3	Balkh, Afghanistan	AFG	Afghanistan	Balkh	South Asia	0.301	(36.7550603, 66.8975372)	36.755060	66.897537
4	Bamyan, Afghanistan	AFG	Afghanistan	Bamyan	South Asia	0.325	(34.8100067, 67.8212104)	34.810007	67.821210

```
In [25]: df_region = regions[regions["country"] == "Nigeria"]
df_region.head()
```

Out[25]:

	LocationName	ISO	country	region	world_region	MPI	geo	lat	lon
664	Sokoto, Nigeria	NGA	Nigeria	Sokoto	Sub-Saharan Africa	0.548	(13.0058731, 5.2475518)	13.005873	5.247552
665	Zamfara, Nigeria	NGA	Nigeria	Zamfara	Sub-Saharan Africa	0.605	(6.907529, 3.5812692)	6.907529	3.581269
666	Katsina, Nigeria	NGA	Nigeria	Katsina	Sub-Saharan Africa	0.520	(12.5139317, 7.6114217)	12.513932	7.611422
667	Jigawa, Nigeria	NGA	Nigeria	Jigawa	Sub-Saharan Africa	0.552	(12.4460001, 9.7232673)	12.446000	9.723267
668	Yobe, Nigeria	NGA	Nigeria	Yobe	Sub-Saharan Africa	0.635	(12.1871412, 11.7068294)	12.187141	11.706829

```
In [26]: fig = px.scatter_mapbox(
regions,
lat = "lat",
lon = "lon",
center = {"lat": 1.284996, "lon": 36.82723},
color = "MPI",
width = 900,
height = 600,
hover_data = ["MPI"],
color_continuous_scale=px.colors.cyclical.IceFire, size_max=15, zoom=10
)
fig.update_layout(mapbox_style = "open-street-map")
fig.show()
```



```
In [27]: themes = pd.read_csv("loan_theme_ids.csv")
print(themes.shape)
print(themes.info())
themes.head()
```

(779092, 4)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 779092 entries, 0 to 779091
Data columns (total 4 columns):
Column Non-Null Count Dtype
--- ---
0 id 779092 non-null int64
1 Loan Theme ID 764279 non-null object
2 Loan Theme Type 764279 non-null object
3 Partner ID 764279 non-null float64
dtypes: float64(1), int64(1), object(2)
memory usage: 23.8+ MB
None

Out[27]:

	id	Loan Theme ID	Loan Theme Type	Partner ID
0	638631	a1050000000skGl	General	151.0
1	640322	a1050000000skGl	General	151.0
2	641006	a1050000002X1ij	Higher Education	160.0
3	641019	a1050000002X1ij	Higher Education	160.0
4	641594	a1050000002VbsW	Subsistence Agriculture	336.0

```
In [28]: themes["Loan Theme Type"].value_counts()
```

```
Out[28]: General          380693
Underserved          62575
Agriculture          40108
Rural Inclusion        21228
Water                20131
...
CAMEO Partnership      2
Motorbike              2
SME (Tanzania)         1
Vulnerable Populations (Iraqi)  1
Property Rights        1
Name: Loan Theme Type, Length: 203, dtype: int64
```

```
In [29]: themes_region = pd.read_csv("loan_themes_by_region.csv")
print(themes_region.shape)
print(themes_region.info())
themes_region.head()
```

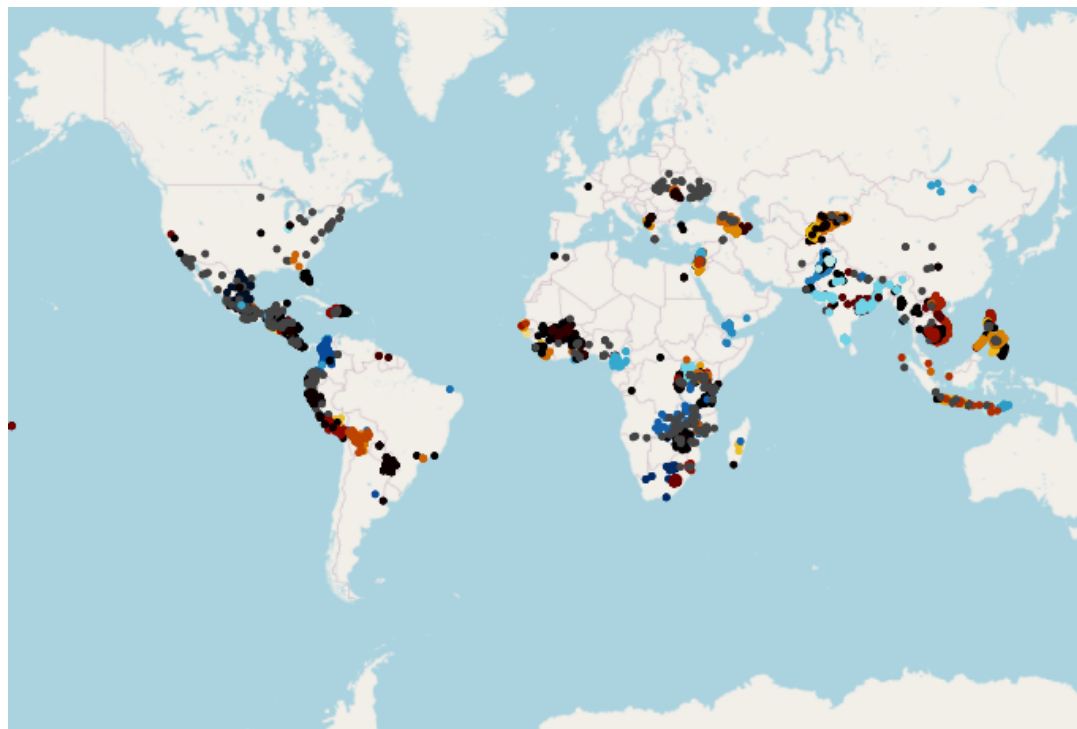
```
(15736, 21)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15736 entries, 0 to 15735
Data columns (total 21 columns):
#   Column              Non-Null Count  Dtype
---  -
0   Partner ID          15736 non-null  int64
1   Field Partner Name  15736 non-null  object
2   sector              15736 non-null  object
3   Loan Theme ID       15736 non-null  object
4   Loan Theme Type     15736 non-null  object
5   country             15736 non-null  object
6   forkiva             15736 non-null  object
7   region             15736 non-null  object
8   geocode_old         1200 non-null   object
9   ISO                 15722 non-null  object
10  number              15736 non-null  int64
11  amount              15736 non-null  int64
12  LocationName        15736 non-null  object
13  geocode             13662 non-null  object
14  names               13661 non-null  object
15  geo                 15736 non-null  object
16  lat                 13662 non-null  float64
17  lon                 13662 non-null  float64
18  mpi_region          15722 non-null  object
19  mpi_geo             9671 non-null   object
20  rural_pct           14344 non-null  float64
dtypes: float64(3), int64(3), object(15)
memory usage: 2.5+ MB
None
```

Out[29]:

	Partner ID	Field Partner Name	sector	Loan Theme ID	Loan Theme Type	country	forkiva	region	geocode_old	ISO	...	amount
0	9	KREDIT Microfinance Institution	General Financial Inclusion	a10500000000sIfi	Higher Education	Cambodia	No	Banteay Meanchey	(13.75, 103.0)	KHM	...	
1	9	KREDIT Microfinance Institution	General Financial Inclusion	a105000000068jPe	Vulnerable Populations	Cambodia	No	Battambang Province	NaN	KHM	...	20
2	9	KREDIT Microfinance Institution	General Financial Inclusion	a10500000000sIfi	Higher Education	Cambodia	No	Battambang Province	NaN	KHM	...	5
3	9	KREDIT Microfinance Institution	General Financial Inclusion	a105000000068jPe	Vulnerable Populations	Cambodia	No	Kampong Cham Province	(12.0, 105.5)	KHM	...	60
4	9	KREDIT Microfinance Institution	General Financial Inclusion	a1050000002X1Uu	Sanitation	Cambodia	No	Kampong Cham Province	(12.0, 105.5)	KHM	...	

5 rows × 21 columns

```
In [30]: fig = px.scatter_mapbox(
    themes_region,
    lat = "lat",
    lon = "lon",
    center = {"lat": 1.284996, "lon": 36.82723},
    color = "rural_pct",
    width = 900,
    height = 600,
    hover_data = ["rural_pct"],
    color_continuous_scale=px.colors.cyclical.IceFire, size_max=15, zoom=10
)
fig.update_layout(mapbox_style = "open-street-map")
fig.show()
```



```
In [31]: themes_region["sector"].value_counts()
```

```
Out[31]: General Financial Inclusion    13679
Clean Energy                        840
Other                               338
Agriculture                         300
Mobile Money and ICT                210
Education                           143
Water and Sanitation                 92
Artisan                             55
DSE Direct                           45
SME Financial Inclusion               32
Health                               2
Name: sector, dtype: int64
```