MAPs to Brazil States

Developed by Manuel Robalinho

Set/2018

In [39]:

```
# Map Sources:
# https://gadm.org/download_country_v3.html
# Using GeoPandas
```

In [63]:

```
%matplotlib inline

import matplotlib.pyplot as plt
import geopandas as gpd
```

In [3]:

```
path = 'ML/GeoPandas/Brazil/'
```

In [32]:

```
# Tables Brasil and states
uf_br0 = gpd.read_file(path + 'shp/gadm36_BRA_0.shp')
uf_br1 = gpd.read_file(path + 'shp/gadm36_BRA_1.shp')
```

In [33]:

```
# Tables Brasil Distritos and municipios
uf_br2 = gpd.read_file(path + 'shp/gadm36_BRA_2.shp')
uf_br3 = gpd.read_file(path + 'shp/gadm36_BRA_3.shp')
```

In [64]:

```
# BRASIL
uf_br0
```

Out[64]:

	GID_0	NAME_0	geometry
0	BRA	Brazil	(POLYGON ((-52.07069397 -32.0284729, -52.07069...

In [65]:

```
# Statues - UF  BRAZIL
uf_br1
```

Out[65]:

	GID_0	NAME_0	GID_1	NAME_1	VARNAME_1	NL_NAME_1	TYPE_1	EN(
0	BRA	Brazil	BRA.1_1	Acre	None	None	Estado	Stat
1	BRA	Brazil	BRA.2_1	Alagoas	None	None	Estado	Stat
2	BRA	Brazil	BRA.3_1	Amapá	None	None	Estado	Stat
3	BRA	Brazil	BRA.4_1	Amazonas	Amazone	None	Estado	Stat
4	BRA	Brazil	BRA.5_1	Bahia	Ba¡a	None	Estado	Stat
5	BRA	Brazil	BRA.6_1	Ceará	None	None	Estado	Stat
6	BRA	Brazil	BRA.7_1	Distrito Federal	None	None	Distrito Federal	Fed Dist
7	BRA	Brazil	BRA.8_1	Espírito Santo	Espiritu Santo	None	Estado	Stat
8	BRA	Brazil	BRA.9_1	Goiás	Goiáz|Goyáz	None	Estado	Stat
9	BRA	Brazil	BRA.10_1	Maranhão	São Luíz de Maranhão	None	Estado	Stat
10	BRA	Brazil	BRA.12_1	Mato Grosso	Matto Grosso	None	Estado	Stat

	GID_0	NAME_0	GID_1	NAME_1	VARNAME_1	NL_NAME_1	TYPE_1	ENG
11	BRA	Brazil	BRA.11_1	Mato Grosso do Sul	None	None	Estado	Stat
12	BRA	Brazil	BRA.13_1	Minas Gerais	Minas|Minas Geraes	None	Estado	Stat
13	BRA	Brazil	BRA.14_1	Pará	None	None	Estado	Stat
14	BRA	Brazil	BRA.15_1	Paraíba	Parahyba	None	Estado	Stat
15	BRA	Brazil	BRA.16_1	Paraná	None	None	Estado	Stat
16	BRA	Brazil	BRA.17_1	Pernambuco	Pernambouc	None	Estado	Stat
17	BRA	Brazil	BRA.18_1	Piauí	Piauhy	None	Estado	Stat
18	BRA	Brazil	BRA.19_1	Rio de Janeiro	None	None	Estado	Stat
19	BRA	Brazil	BRA.20_1	Rio Grande do Norte	None	None	Estado	Stat
20	BRA	Brazil	BRA.21_1	Rio Grande do Sul	None	None	Estado	Stat
21	BRA	Brazil	BRA.22_1	Rondônia	Guaporé	None	Estado	Stat

	GID_0	NAME_0	GID_1	NAME_1	VARNAME_1	NL_NAME_1	TYPE_1	ENG
22	BRA	Brazil	BRA.23_1	Roraima	Rio Branco	None	Estado	Stat
23	BRA	Brazil	BRA.24_1	Santa Catarina	Santa Catharina	None	Estado	Stat
24	BRA	Brazil	BRA.25_1	São Paulo	None	None	Estado	Stat
25	BRA	Brazil	BRA.26_1	Sergipe	None	None	Estado	Stat
26	BRA	Brazil	BRA.27_1	Tocantins	None	None	Estado	Stat

In [66]:

```
# Municipios
uf_br2.head()
```

Out[66]:

	GID_0	NAME_0	GID_1	NAME_1	NL_NAME_1	GID_2	NAME_2	VARNAME
0	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.1_1	Acrelândia	None
1	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.2_1	Assis Brazil	None
2	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.3_1	Brasiléia	None
3	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.4_1	Bujari	None
4	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.5_1	Capixaba	None

In [67]:

```
# Distritos
uf_br3.head()
```

Out[67]:

	GID_0	NAME_0	GID_1	NAME_1	NL_NAME_1	GID_2	NAME_2	NL_NAME
0	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.1_1	Acrelândia	None
1	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.2_1	Assis Brazil	None
2	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.3_1	Brasiléia	None
3	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.4_1	Bujari	None
4	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.5_1	Capixaba	None

In [68]:

```
uf_br1.plot(cmap='YlOrRd')
```

Out[68]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x212179a15c0>
```
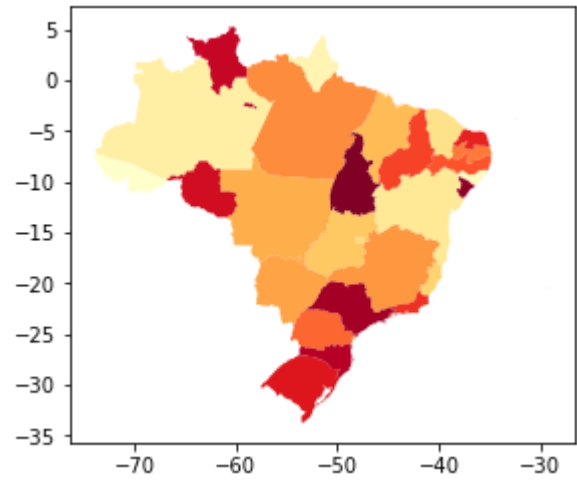
In [69]:

```
# List of states - UF
uf_br1x = uf_br1[['NAME_1', 'geometry']]
uf_br1x
```

Out[69]:

	NAME_1	geometry
0	Acre	POLYGON ((-68.10552979000001 -10.72191715, -68...
1	Alagoas	(POLYGON ((-35.88986206 -9.844305990000001, -3...
2	Amapá	(POLYGON ((-50.82569504 2.522084, -50.82569504...
3	Amazonas	POLYGON ((-58.13698959 -7.35613918, -58.136821...
4	Bahia	(POLYGON ((-38.69208145 -17.95958328, -38.6920...
5	Ceará	(POLYGON ((-40.83180618 -2.8812499, -40.831806...
6	Distrito Federal	POLYGON ((-47.93915176 -16.05134773, -48.02110...
7	Espírito Santo	(POLYGON ((-40.88402939 -21.16124916, -40.8840...
8	Goiás	POLYGON ((-49.0048027 -18.53718185, -49.012477...
9	Maranhão	(POLYGON ((-44.54986191 -1.88458395, -44.54986...
10	Mato Grosso	POLYGON ((-52.47142792 -16.12672043, -52.48841...
11	Mato Grosso do Sul	POLYGON ((-54.16836166 -23.99920082, -54.17062...
12	Minas Gerais	POLYGON ((-50.84402084 -19.9675808, -50.853321...
13	Pará	(POLYGON ((-47.29791641 -0.63375098, -47.29791...
14	Paraíba	(POLYGON ((-43.01207733 -9.408011439999999, -4...
15	Paraná	(POLYGON ((-48.36513901 -25.73402786, -48.3651...
16	Pernambuco	(POLYGON ((-35.13597107 -8.83791733, -35.13597...
17	Piauí	POLYGON ((-42.69078445 -9.54547024, -42.691215...
18	Rio de Janeiro	(POLYGON ((-44.67124939 -23.35458374, -44.6712...
19	Rio Grande do Norte	(POLYGON ((-35.10958481 -6.1937499, -35.109584...
20	Rio Grande do Sul	(POLYGON ((-52.07069397 -32.0284729, -52.07069...
21	Rondônia	POLYGON ((-62.89218903 -12.86013603, -62.89211...
22	Roraima	POLYGON ((-60.0378952 0.26348999, -60.04676437...
23	Santa Catarina	(POLYGON ((-48.54236221 -27.93180466, -48.5423...
24	São Paulo	(POLYGON ((-48.08235931 -25.28430557, -48.0823...
25	Sergipe	(POLYGON ((-37.38458252 -11.45985985, -37.3845...
26	Tocantins	POLYGON ((-49.23736191 -12.88397312, -49.23532...

In [70]:

```python
# Transform POLYGON to POINT
# copy poly to new GeoDataFrame
points = uf_br1x.copy()
# change the geometry
points.geometry = points['geometry'].centroid
# same crs
#points.crs = poly.crs
points.head()
```

Out[70]:

	NAME_1	geometry
0	Acre	POINT (-70.44666617524693 -9.309916735880247)
1	Alagoas	POINT (-36.62319561662706 -9.51555361360583)
2	Amapá	POINT (-51.96103309015383 1.440043429560412)
3	Amazonas	POINT (-64.71488057979415 -4.186776806608581)
4	Bahia	POINT (-41.73007712980874 -12.47331113316648)

In [71]:

```python
# PLOT State points
f, ax = plt.subplots(1, figsize=(8,10))
ax.set_axis_on()
f.suptitle('BRAZIL')
# Plot the states area
ax = uf_br1x.plot(ax=ax, facecolor='blue', alpha=1, linewidth=0, cmap='YlOrRd')

# Plot the labels
for x, y, label in zip(points.geometry.x, points.geometry.y, points.NAME_1):
    ax.annotate(label, xy=(x, y), xytext=(3, 3), alpha=3, textcoords="offset points",co
lor='blue')
```
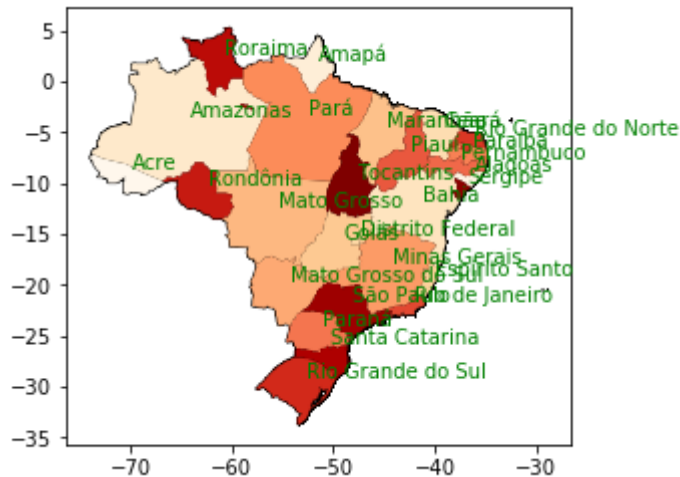
BRAZIL
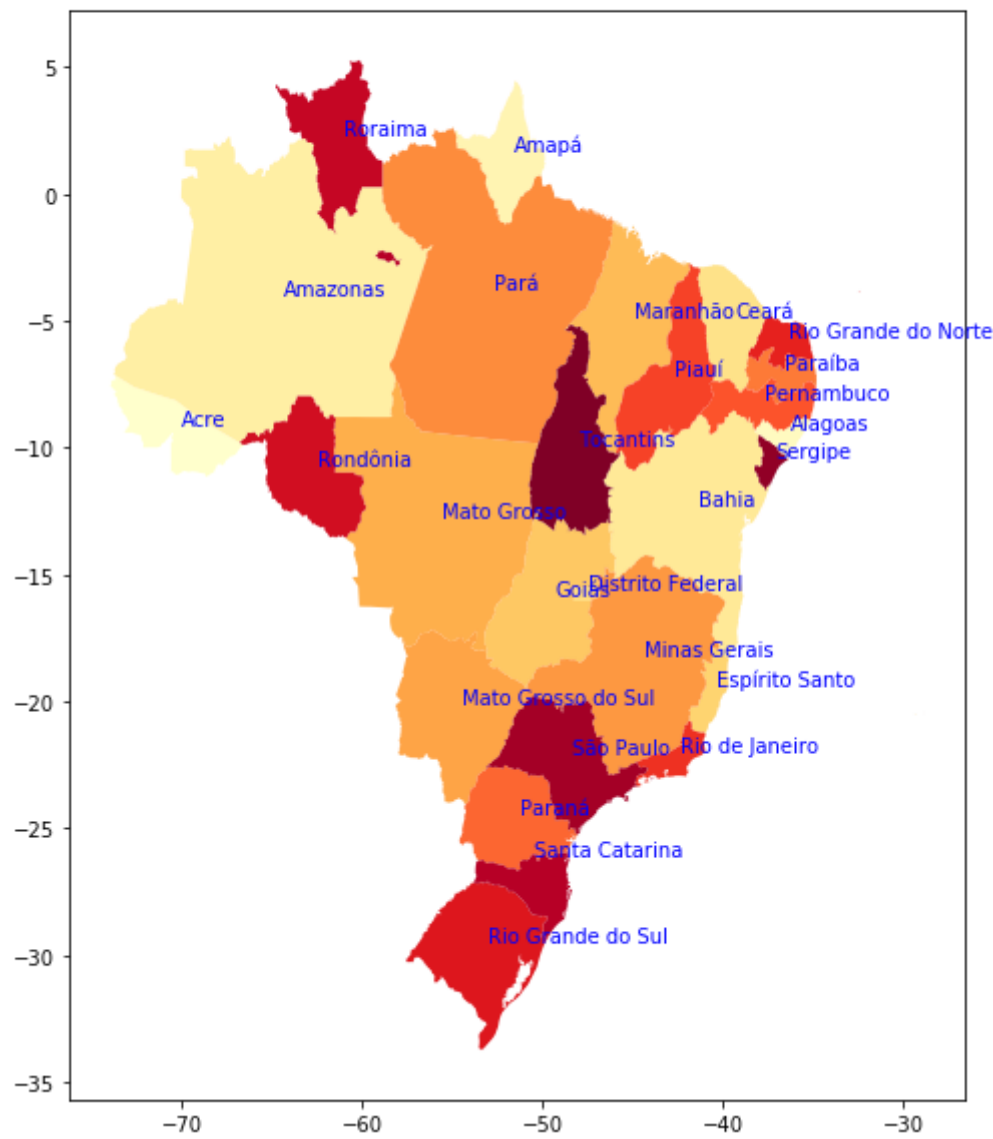
In [72]:

```python
# Plot Base Country ans states
ax.figsize=(8,10)
# Plot the states area
base = uf_br1x.plot( edgecolor='black', cmap='OrRd')
# Plot Base continent - Cities
ax = uf_br1x.plot(ax=base, facecolor='blue', alpha=1, linewidth=0,  cmap='OrRd')

# Plot Base continent - Cities
for x, y, label in zip(points.geometry.x, points.geometry.y, points.NAME_1):
    ax.annotate(label, xy=(x, y), xytext=(3, 3), textcoords="offset points",color='gree
n')
```

In [73]:

```
uf_br2.head()
```

Out[73]:

	GID_0	NAME_0	GID_1	NAME_1	NL_NAME_1	GID_2	NAME_2	VARNAME
0	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.1_1	Acrelândia	None
1	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.2_1	Assis Brazil	None
2	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.3_1	Brasiléia	None
3	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.4_1	Bujari	None
4	BRA	Brazil	BRA.1_1	Acre	None	BRA.1.5_1	Capixaba	None

In [74]:

```
# CEARA - UF
uf_br1_ce = uf_br1[uf_br1.GID_1 == 'BRA.6_1']
uf_br1_ce
```

Out[74]:

	GID_0	NAME_0	GID_1	NAME_1	VARNAME_1	NL_NAME_1	TYPE_1	ENGTY
5	BRA	Brazil	BRA.6_1	Ceará	None	None	Estado	State

In [75]:

```
# MAP Of CEARA - Filter UF
# Statues - UF  BRAZIL
uf_br2_ce = uf_br2[uf_br2.GID_1 == 'BRA.6_1']
```

In [76]:

```
uf_br2_ce.head(10)
```

Out[76]:

	GID_0	NAME_0	GID_1	NAME_1	NL_NAME_1	GID_2	NAME_2	VARN/
615	BRA	Brazil	BRA.6_1	Ceará	None	BRA.6.1_1	Abaiara	None
616	BRA	Brazil	BRA.6_1	Ceará	None	BRA.6.2_1	Acarapé	None
617	BRA	Brazil	BRA.6_1	Ceará	None	BRA.6.3_1	Acaraú	None
618	BRA	Brazil	BRA.6_1	Ceará	None	BRA.6.4_1	Acopiara	None
619	BRA	Brazil	BRA.6_1	Ceará	None	BRA.6.5_1	Aiuaba	None
620	BRA	Brazil	BRA.6_1	Ceará	None	BRA.6.6_1	Alcântaras	None
621	BRA	Brazil	BRA.6_1	Ceará	None	BRA.6.7_1	Altaneira	None
622	BRA	Brazil	BRA.6_1	Ceará	None	BRA.6.8_1	Alto Santo	None
623	BRA	Brazil	BRA.6_1	Ceará	None	BRA.6.9_1	Amontada	None
624	BRA	Brazil	BRA.6_1	Ceará	None	BRA.6.10_1	Antonina do Norte	None

In [77]:

```
# List of Cities from UF
uf_br2x = uf_br2_ce[['NAME_2', 'geometry']]
uf_br2x.head()
```

Out[77]:

	NAME_2	geometry
615	Abaiara	POLYGON ((-38.99930573 -7.31215, -38.98537445 ...
616	Acarapé	POLYGON ((-38.6201973 -4.25379992, -38.6841697...
617	Acaraú	(POLYGON ((-39.99142456 -3.23660398, -40.00414...
618	Acopiara	POLYGON ((-39.49590683 -6.33128881, -39.502334...
619	Aiuaba	POLYGON ((-40.33156967 -6.80437183, -40.342914...

In [78]:

```
# Transform POLYGON to POINT
# copy poly to new GeoDataFrame
points = uf_br2x.copy()
# change the geometry
points.geometry = points['geometry'].centroid

points.head()
```

Out[78]:

	NAME_2	geometry
615	Abaiara	POINT (-39.03744666881347 -7.360263034220808)
616	Acarapé	POINT (-38.66316186615578 -4.222426778424921)
617	Acaraú	POINT (-40.08873728909125 -2.96786068805975)
618	Acopiara	POINT (-39.49275273464827 -6.122398590402195)
619	Aiuaba	POINT (-40.29881732797426 -6.605116010704714)

In [79]:

```python
# PLOT Municipios from UF
f, ax = plt.subplots(1, figsize=(12,14))
ax.set_axis_on()
f.suptitle('BRAZIL - CEARA')
# Plot the states area
ax = uf_br1_ce.plot(ax=ax, facecolor='blue', alpha=1, linewidth=0, cmap='YlOrRd')

# Plot the labels
for x, y, label in zip(points.geometry.x, points.geometry.y, points.NAME_2):
    ax.annotate(label, xy=(x, y), xytext=(3, 3), alpha=3, textcoords="offset points",color='blue')
```
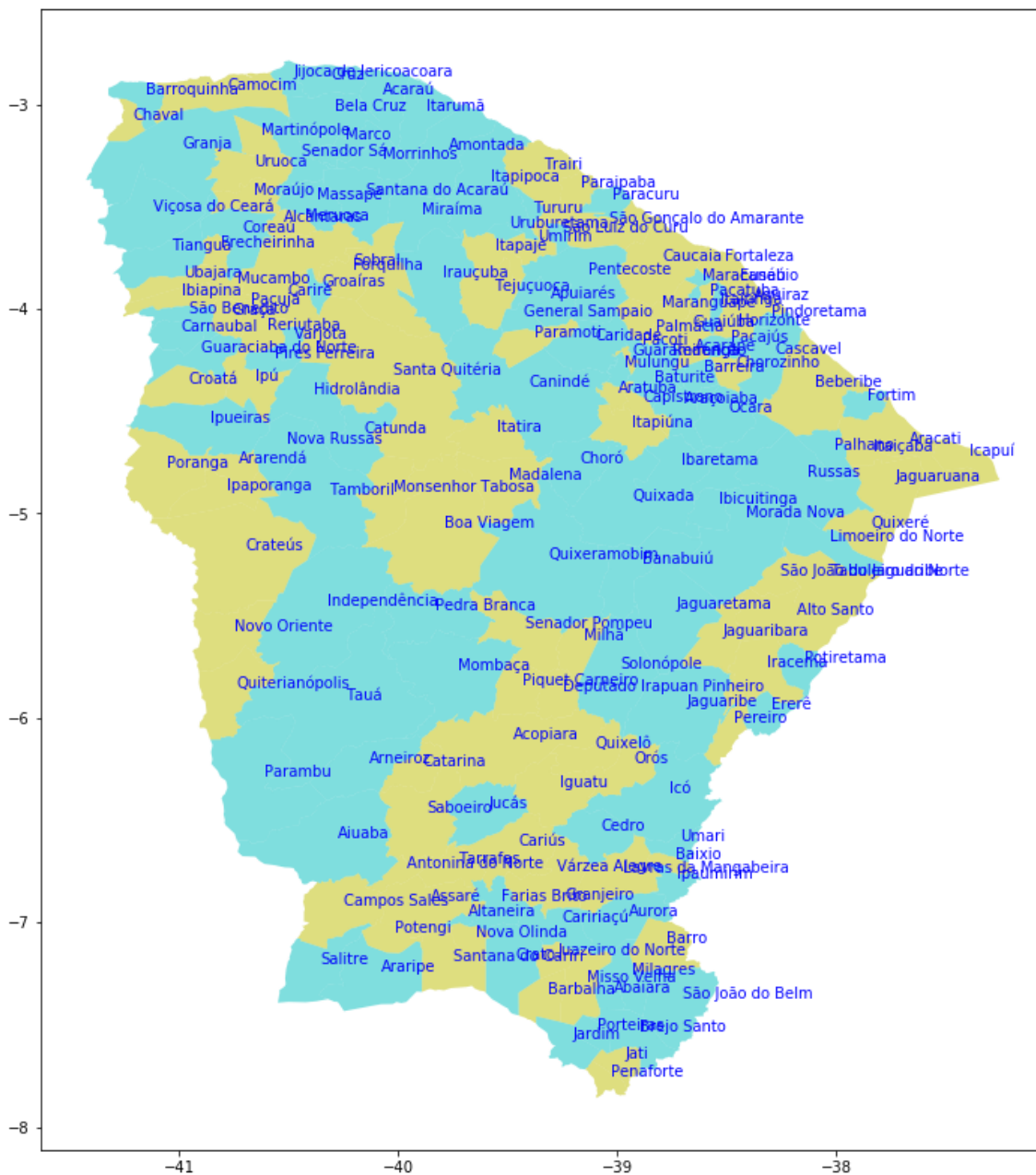
BRAZIL - CEARA
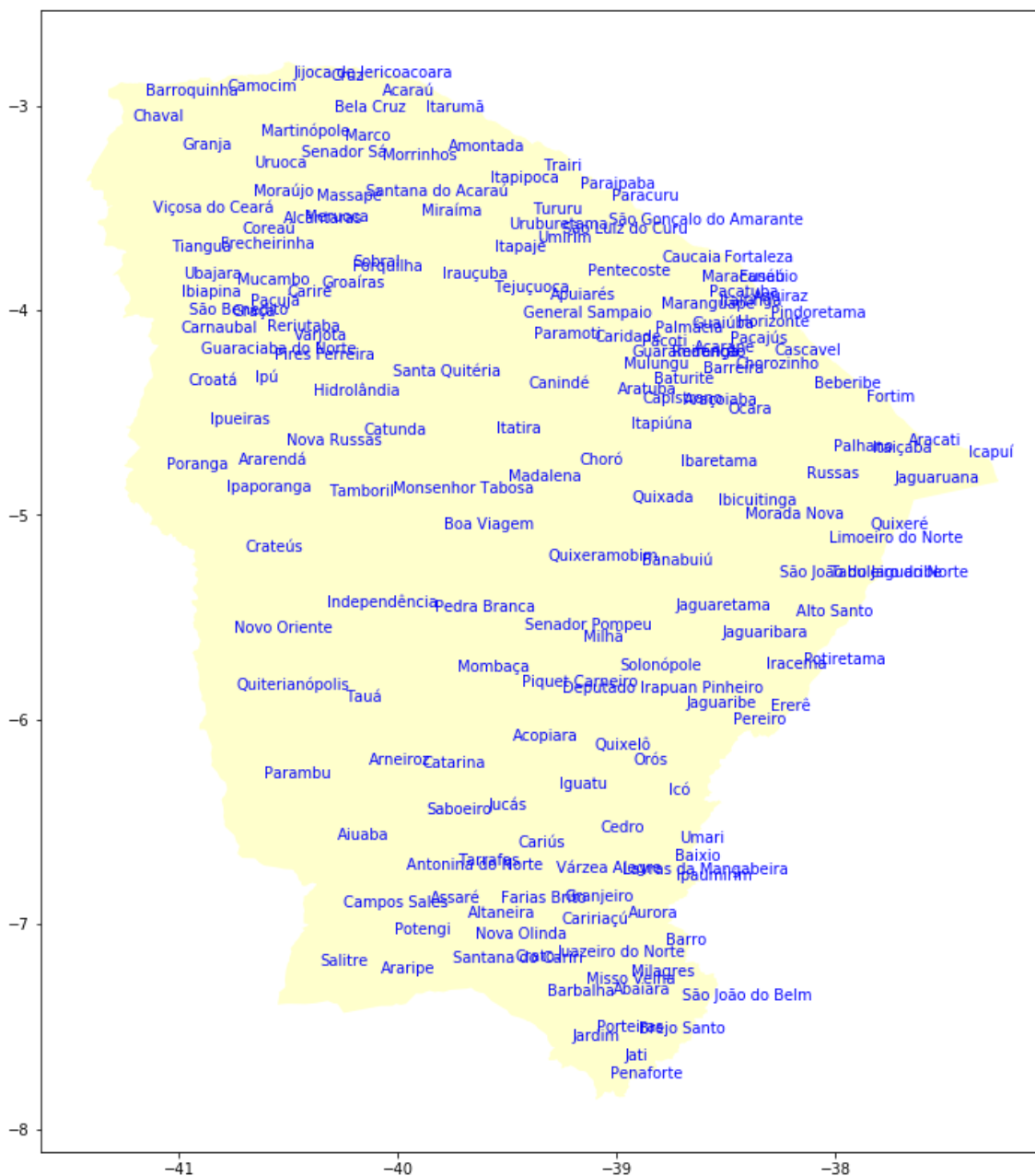
In [80]:

```python
# PLOT Municipios from UF
f, ax = plt.subplots(1, figsize=(12,14))
ax.set_axis_on()
f.suptitle('BRAZIL - CEARA')
# Plot the states area
ax = uf_br2x.plot(ax=ax, facecolor='cy', alpha=0.5, linewidth=1)

# Plot the labels
for x, y, label in zip(points.geometry.x, points.geometry.y, points.NAME_2):
    ax.annotate(label, xy=(x, y), xytext=(3, 3), alpha=3, textcoords="offset points",co
lor='blue')
```
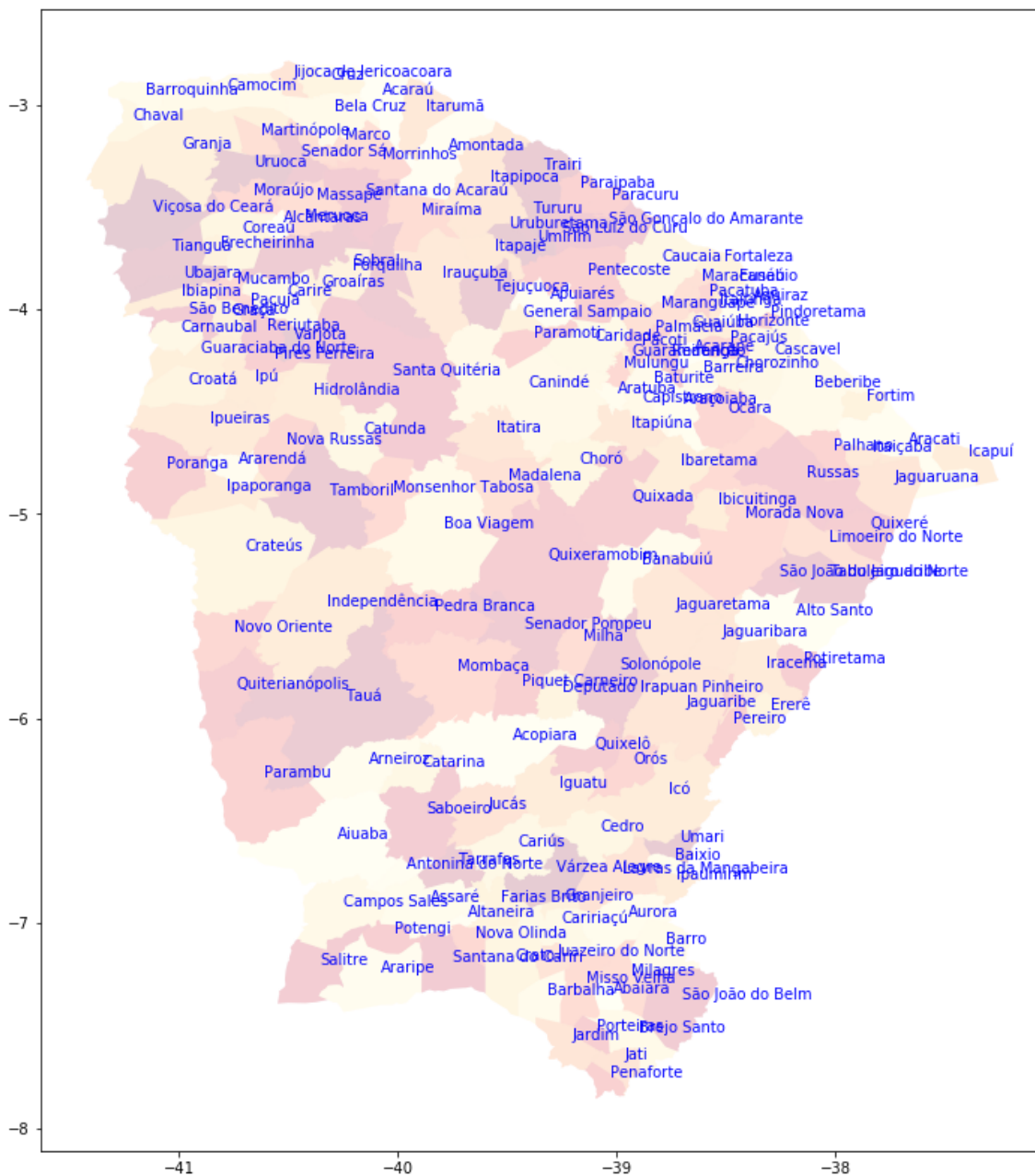

BRAZIL - CEARA

In [81]:

```python
# PLOT Municipios from UF
f, ax = plt.subplots(1, figsize=(12,14))
ax.set_axis_on()
f.suptitle('BRAZIL - CEARA')
# Plot the states area
ax = uf_br2x.plot(ax=ax, facecolor='blue', alpha=0.2, linewidth=0, cmap='YlOrRd')

# Plot the labels
for x, y, label in zip(points.geometry.x, points.geometry.y, points.NAME_2):
    ax.annotate(label, xy=(x, y), xytext=(3, 3), alpha=3, textcoords="offset points",co
lor='blue')
```


BRAZIL - CEARA

In [ ]:

In [ ]: