7/5/25, 11:13 StackEdi

# **RESIDUOS APROVECHABLES**



#### MAPA DE CALOR CON LEYENDA

#### Importar librerias a utilizar

import geopandas as gpd import pandas as pd import matplotlib.pyplot as plt from matplotlib.patches import Patch

### Cargar el shapefile de los distritos de Lambayeque

```
shapefile_path = "DISTRITO.shp" # Ruta al archivo .shp
distritos = gpd.read_file(shapefile_path)
```

# Cargar el archivo Excel con los datos de residuos sólidos

```
excel_path = "mapa1.xlsx" # Ruta al archivo Excel
residuos = pd.read_excel(excel_path)
```

# Renombrar las columnas para que coincidan con el shapefile

```
residuos = residuos.rename(columns={
"DISTRITO": "DISTRITO", # Ajusta si la columna tiene un nombre diferente
"aprovechable": "INTENSIDAD" # Cambia a la columna correspondiente de datos
})
```

# Clasificar la intensidad en categorías (Bajo, Medio, Alto)

```
bins = [0, residuos["INTENSIDAD"].quantile(0.33), \\ residuos["INTENSIDAD"].quantile(0.66), residuos["INTENSIDAD"].max()] \\ labels = ["Bajo", "Medio", "Alto"] \\ residuos["CATEGORÍA"] = pd.cut(residuos["INTENSIDAD"], bins=bins, labels=labels, include_lowest=True)
```

https://stackedit.io/app#

7/5/25, 11:13 StackEdit

### Unir los datos geoespaciales con los datos de residuos sólidos

mapa = distritos.merge(residuos, on="DISTRITO", how="left")

#### Crear el mapa

```
fig, ax = plt.subplots(1, 1, figsize=(12, 10))
mapa.plot(column="CATEGORÍA",
cmap="coolwarm", # Escala de colores adecuada para categorías
legend=True,
legend_kwds={'title': "Categoría de generación de residuos sólidos"},
ax=ax)
```

### Agregar los nombres de los distritos al mapa

```
for x, y, label in zip(mapa.geometry.centroid.x, mapa.geometry.centroid.y, mapa["DISTRITO"]):

ax.text(x, y, label, fontsize=8, ha="center", color="black") # Personaliza el estilo del texto

###Crear leyenda personalizada al costado del gráfico

categorias_colores = {"Bajo": "blue", "Medio": "grey", "Alto": "red"} # Colores por categoría distritos_categorias = residuos[["DISTRITO", "CATEGORÍA"]].sort_values("CATEGORÍA")
```

# Posición inicial de la leyenda personalizada

```
x_text = 1.02 # Posición en el eje X (fuera del mapa)
y_text = 1.0 # Posición inicial en el eje Y
for index, row in distritos_categorias.iterrows():
distrito = row["DISTRITO"]
categoria = row["CATEGORÍA"]
color = categorias_colores[categoria]
```

# Agregar un punto de color seguido del nombre del distrito

```
ax.text(x_text, y_text, "●", color=color, fontsize=10, transform=ax.transAxes, va="top") ax.text(x_text + 0.03, y_text, distrito, fontsize=10, transform=ax.transAxes, va="top") y_text -= 0.03 # Desplazar hacia abajo para el siguiente distrito
```

https://stackedit.io/app# 2/3

7/5/25, 11:13 StackEdit

# Personalización del mapa

ax.set\_title("Mapa de calor de los distritos de residuos aprovechables, 2019-2022", fontsize=14)

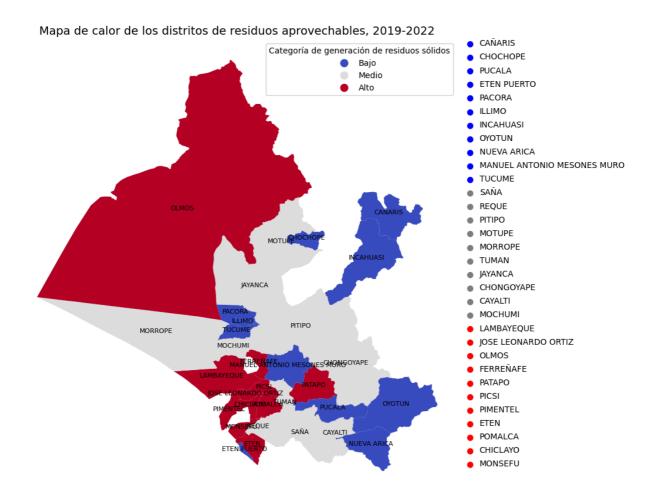
ax.set\_axis\_off() # Oculta los ejes

### Guardar el mapa como imagen

plt.savefig("mapa\_residuos\_lambayeque\_leyenda.png", dpi=300, bbox\_inches="tight")

### Mostrar el mapa

plt.show()



https://stackedit.io/app# 3/3