

## Reto 1: Diagnóstico de Salud Pública por Regiones

### Archivos:

- encuesta\_salud.csv  
Columnas: ID, Edad, Sexo, Peso, Altura, Presion\_Arterial, Region, Ingresos
- region\_referencia.csv  
Columnas: Region, Nombre\_Completo, Zona

### Instrucciones:

1. Carga los datos.
  2. Inspecciona el DataFrame (.head(), .info()).
  3. Convierte a numérico Peso, Altura, y Presion\_Arterial, aplicando .to\_numeric(errors="coerce").
  4. Elimina duplicados.
  5. Calcula el IMC (Índice de Masa Corporal) = peso / altura<sup>2</sup> y discretiza en categorías: Bajo, Normal, Sobrepeso, Obesidad.
  6. Rellena los valores nulos de ingresos con la **media** por región.
  7. Une con el archivo de regiones (merge tipo left).
  8. Agrupa por Zona y calcula promedios de IMC y presión arterial.
  9. Crea un índice jerárquico con Zona y Sexo.
  10. Usa .unstack() para mostrar comparativas entre sexos y zonas.
- 

## Reto 2: Rendimiento Académico Nacional

### Archivos:

- calificaciones\_escuelas.csv  
Columnas: ID\_Estudiante, Edad, Genero, Escuela, Estado, Materia, Calificacion
- materias.csv  
Columnas: Materia, Tipo (Científica, Humanística, Técnica)

### Instrucciones:

1. Carga los datos y explora los primeros registros.
  2. Elimina duplicados.
  3. Convierte Calificacion a numérico con coerción de errores.
  4. Rellena valores nulos con la mediana general.
  5. Aplica pd.qcut() para categorizar calificaciones en Q1, Q2, Q3, Q4.
  6. Une con materias.csv para clasificar las materias por tipo.
  7. Agrupa por Escuela y Tipo de materia, y calcula el promedio por grupo.
  8. Usa .pivot() para crear una tabla dinámica por Escuela vs Tipo.
  9. Convierte la tabla de nuevo con .melt() para análisis posterior.
- 

## Reto 3: Consumo de Energía por País

### Archivos:

- energia.csv  
Columnas: ID, Pais, Año, Consumo\_kWh, Tipo\_Energia
- paises\_info.csv  
Columnas: Pais, Continente, Region\_Climatica

### Instrucciones:

1. Carga e inspecciona.
  2. Usa `.drop_duplicates()`.
  3. Rellena nulos en consumo con el promedio por país.
  4. Aplica `pd.cut()` para clasificar el consumo en rangos: Bajo, Medio, Alto.
  5. Une con `países_info.csv` usando merge tipo inner.
  6. Crea una estructura jerárquica con `Continente > País`.
  7. Agrupa por Continente y Tipo\_Energia y calcula suma de consumo.
  8. Aplica `.swaplevel()` y `.stack()` para reorganizar los índices.
  9. Usa `.reset_index()` para dejarlo plano nuevamente.
- 

## Reto 4: Empleo y Sueldos en el Sector Tecnológico

### Archivos:

- `empleados_tech.csv`  
Columnas: ID, Nombre, Edad, Departamento, Salario, Nivel\_Educativo, Proyecto
- `niveles_estudios.csv`  
Columnas: Nivel\_Educativo, Descripcion

### Instrucciones:

1. Lee e inspecciona el archivo.
  2. Convierte salario a numérico con coerción.
  3. Discretiza salarios con `pd.cut()` en 4 rangos personalizados.
  4. Rellena valores nulos de edad con la media por Departamento.
  5. Elimina duplicados.
  6. Une con `niveles_estudios.csv` (merge left).
  7. Agrupa por Nivel\_Educativo y Departamento para obtener promedio salarial.
  8. Usa `.pivot()` para visualizar por Nivel\_Educativo vs Departamento.
  9. Usa `.melt()` para convertir a formato largo para visualización.
- 

## Reto 5: Análisis de Producción Automotriz

### Archivos:

- `produccion_autos.csv`  
Columnas: ID, Marca, Modelo, Año, Unidades\_Producidas, Defectos\_Reportados, Planta
- `plantas_info.csv`  
Columnas: Planta, País, Tipo\_Producción

### Instrucciones:

1. Carga los archivos y explora.
2. Corrige datos erróneos (ej. Defectos como texto) con `.to_numeric(errors="coerce")`.
3. Elimina duplicados.
4. Imputa NaN en unidades producidas con la mediana por planta.
5. Une con `plantas_info.csv` (merge outer).
6. Calcula tasa de defectos = defectos / unidades.
7. Agrupa por Tipo\_Producción y Marca, y calcula medias.
8. Usa `.set_index()` con Tipo\_Producción y Marca, reorganiza con `.unstack()`.

---

## Reto 6: Estudios de Genética Poblacional

### Archivos:

- `muestras_geneticas.csv`  
Columnas: ID, Individuo, Edad, Grupo\_Etnico, Gen\_A, Gen\_B, Gen\_C, Región
- `grupos_etnicos.csv`  
Columnas: Grupo\_Etnico, Continente, Lengua\_Origen

### Instrucciones:

1. Lee los datos y analiza su estructura.
2. Corrige errores intencionales en Gen\_A, Gen\_B, Gen\_C (texto convertido a NaN).
3. Calcula la suma de los tres genes por individuo.
4. Discretiza la suma genética en niveles con `pd.qcut()`.
5. Rellena valores de edad faltantes por mediana del grupo étnico.
6. Une con `grupos_etnicos.csv` (`merge left`).
7. Agrupa por Continente y Región, calcula promedios genéticos.
8. Usa `.pivot()` para crear matriz Continente vs Región con media genética.
9. Usa `.concat()` para añadir una tabla extra con conteo de individuos por grupo étnico.