

Taller — Creación de un tablero en Power BI o Python

Propósito: que el estudiante desarrolle un tablero interactivo en **Power BI o Python** (usando librerías como Matplotlib, Seaborn o Plotly), aplicando principios de visualización efectiva, métricas clave y storytelling con datos.

1) Entregable principal — Tablero de visualización

Qué es: un tablero que permita explorar datos, identificar tendencias y comunicar hallazgos clave mediante visualizaciones claras y efectivas.

1.1 Definición de métricas

- Ventas totales y promedio mensual.
- Top 5 productos más vendidos.
- Clientes más frecuentes.
- Evolución de ventas por región/tiempo.

1.2 Visualizaciones recomendadas

- Gráfico de barras comparativo.
- Serie temporal de ventas.
- Mapa o gráfico de dispersión por región.
- KPI con indicadores clave.

1.3 Herramientas

- **Opción 1:** Power BI (conexiones a CSV/SQL, creación de medidas DAX).
- **Opción 2:** Python con Plotly, Matplotlib o Seaborn.

2) Actividades del taller

1. Definir un dataset (puede ser de ventas, clientes, finanzas o público disponible).
2. Seleccionar métricas principales para el tablero.
3. Crear al menos **4 visualizaciones** diferentes.
4. Diseñar el tablero con enfoque en claridad y usabilidad.
5. Incluir KPIs que resuman el estado general.
6. Documentar el proceso de construcción.
7. Elaborar una reflexión crítica sobre cómo la visualización ayuda en la toma de decisiones.

3) Instrucciones paso a paso para la entrega

1. Selecciona y prepara un dataset.
2. Define métricas y visualizaciones clave.
3. Construye el tablero en Power BI o Python.
4. Exporta capturas o el archivo final.
5. Documenta decisiones de diseño y justifica elecciones visuales.
6. Concluye con reflexiones sobre la utilidad del tablero.

4) Entregables (carpeta sugerida)

```
taller-dashboard/
└── Dataset.csv          # Dataset usado
└── Tablero.pbix / dashboard.py # Archivo del tablero en Power BI o Python
└── CapturasVisualizaciones/ # Evidencias gráficas
└── DocumentoExplicativo.md # Explicación y justificación
└── Reflexion.md          # Conclusión y aprendizajes
```