

Guía práctica para graficar usando json, html en Pentaho.

Crear los siguientes csv.

productos.csv

	producto_id	nombre	categoria	precio
1	P001	Laptop A	Electrónica	850.00
2	P002	Mouse Inalámbrico	Accesorios	25.50
3	P003	Teclado Mecánico	Accesorios	75.00
4	P004	Smartphone X	Electrónica	550.00
5	P005	Smartwatch B	Wearables	180.00
6	P006	Auriculares C	Electrónica	120.00
7	P007	Funda para Móvil	Accesorios	15.00
8	P008	Monitor Z	Electrónica	280.00
9	P009	Webcam HD	Accesorios	50.00
10	P010	Tablet Y	Electrónica	300.00

ventas.csv

	venta_id	fecha	cantidad	producto_id	cliente_id
1	1001	2025-01-05	1	P001	C101
2	1002	2025-01-05	2	P002	C102
3	1003	2025-01-06	1	P004	C103
4	1004	2025-01-07	3	P007	C104
5	1005	2025-01-08	1	P005	C105
6	1006	2025-02-10	1	P001	C106
7	1007	2025-02-12	2	P003	C107
8	1008	2025-03-01	1	P004	C108
9	1009	2025-03-02	1	P008	C109
10	1010	2025-04-15	1	P010	C110
11	1011	2025-04-16	4	P002	C111
12	1012	2025-04-18	1	P006	C112
13	1013	2025-05-20	2	P009	C113
14	1014	2025-05-22	1	P001	C114
15	1015	2025-06-01	1	P008	C115

1016,2025-06-03,1,P004,C116
1017,2025-07-05,2,P005,C117
1018,2025-07-07,1,P001,C118
1019,2025-08-10,3,P007,C119
1020,2025-08-12,1,P003,C120
1021,2025-08-15,1,P001,C121

Crear la siguiente transformación en Pentaho



En salida deberá tener el archivo *json* y el *dashboard.html*

Este debe ser el contenido del archivo *html*

```
<!DOCTYPE html>
<html lang="es">
<head>
    <meta charset="UTF-8">
    <title>Dashboard de Ventas de Electrónicos</title>
    <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
    <style>
        body { font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
background-color: #f4f4f4; color: #333; margin: 0; padding: 20px; }
.dashboard-container { max-width: 900px; margin: auto; background-color: #fff;
padding: 30px; border-radius: 10px; box-shadow: 0 4px 8px rgba(0,0,0,0.1); }
h1, h2 { text-align: center; color: #0056b3; }
.chart-container { margin: 20px 0; }
canvas { width: 100% !important; height: auto !important; }
```



```
    data: ingresos,
    backgroundColor: [
        'rgba(255, 99, 132, 0.7)',
        'rgba(54, 162, 235, 0.7)',
        'rgba(255, 206, 86, 0.7)',
        'rgba(75, 192, 192, 0.7)',
        'rgba(153, 102, 255, 0.7)',
        'rgba(255, 159, 64, 0.7)'
    ],
    borderColor: [
        'rgba(255, 99, 132, 1)',
        'rgba(54, 162, 235, 1)',
        'rgba(255, 206, 86, 1)',
        'rgba(75, 192, 192, 1)',
        'rgba(153, 102, 255, 1)',
        'rgba(255, 159, 64, 1)'
    ],
    borderWidth: 1
},
options: {
    scales: {
        y: {
            beginAtZero: true,
            title: {
                display: true,
                text: 'Ingresos Totales ($)'
            }
        },
        x: {
            title: {
                display: true,
                text: 'Categoría de Producto'
            }
        }
    },
    plugins: {
        legend: {
            display: true,
            position: 'top',
        },
        title: {
            display: false,
            text: 'Ingresos Totales por Categoría'
        }
    },
    responsive: true,
```

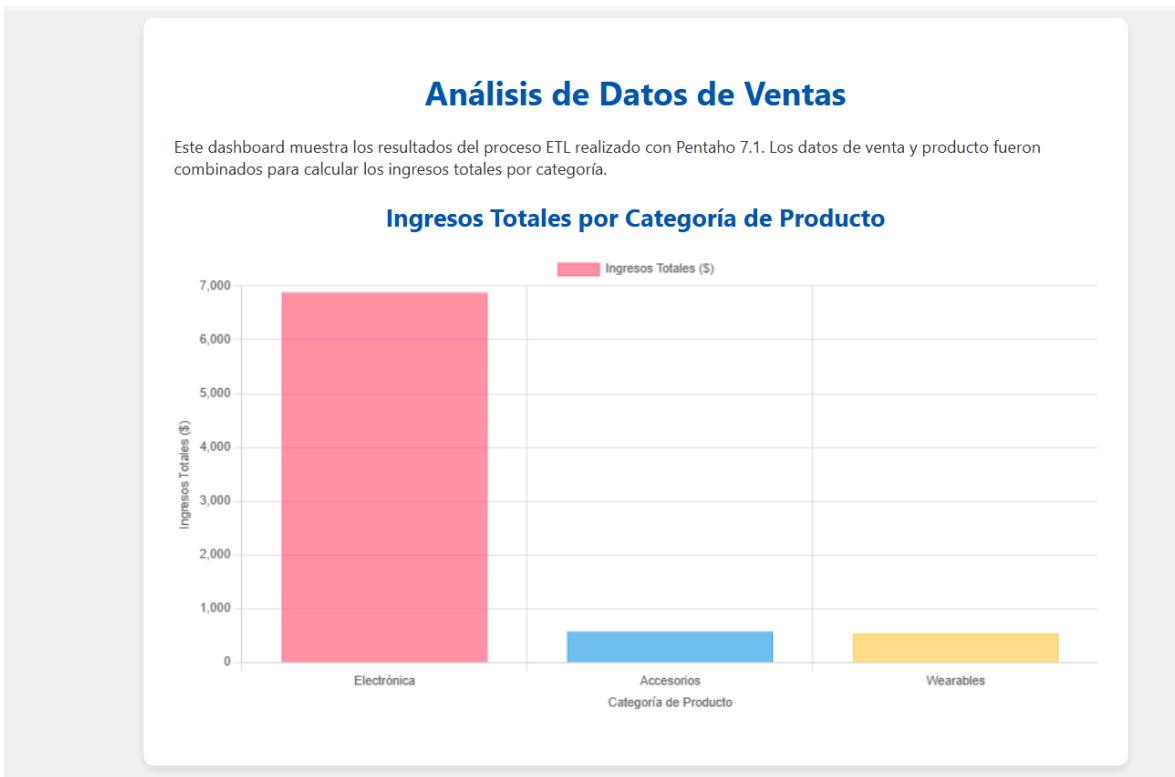
```

        }
    });
})
.catch(error => {
    console.error('Error:', error);
    document.querySelector('.dashboard-container').innerHTML = '<h2>Error al
cargar los datos. Asegúrate de que el archivo JSON exista y de que el nombre
coincida.</h2>';
});
</script>

</body>
</html>

```

Abrir del dashboard usando un servidor local para que funcione



Se puede usar Python , abrir una terminal en la carpeta donde están los archivos html y json

Y ejecutar el siguiente código

```
python -m http.server 8000
```

Abre el navegador y pegar la dirección del dashboard

<http://localhost:8000/dashboard.html>