

- **Desacoplar los datos de las entradas del flujo principal.**
- **Mejorar la organización:** ahora los datos pueden mantenerse, editarse o incluso ser reemplazados por otras fuentes (como un archivo JSON o una base de datos) sin tocar el archivo principal.
- **Facilitar la escalabilidad:** a medida que el contenido crezca, se podrá segmentar, versionar o generar dinámicamente.

4.3.3. Archivo datos/arrayobjetos.php

Este archivo define las entradas estáticas mediante `array_push`, utilizando la misma estructura que en los pasos anteriores. Importante: este archivo **no declara el array \$entradas**; se asume que ese array ya ha sido definido en el archivo que lo incluye. Este detalle obliga a que, en `index.php`, el array `$entradas` se declare antes de hacer el `include`.

El archivo contiene instrucciones como estas:

```
array_push(  
    $entradas,  
    new Entrada(  
        "Primera entrada",  
        "Contenido de la primera entrada"  
    )  
)
```

Se repiten varias veces para simular un conjunto de entradas disponibles en el blog.

4.3.4. Cambios en index.php

El archivo principal sigue cargando la clase desde `clases/Publicacion.php`. Luego, se declara el array `$entradas`, y **se incluye** el archivo `datos/arrayobjetos.php` que lo rellena.

Finalmente, se recorre el array con un bucle y se llama a `damePublicacion` sobre cada objeto para mostrar las entradas en