



SERVLETS

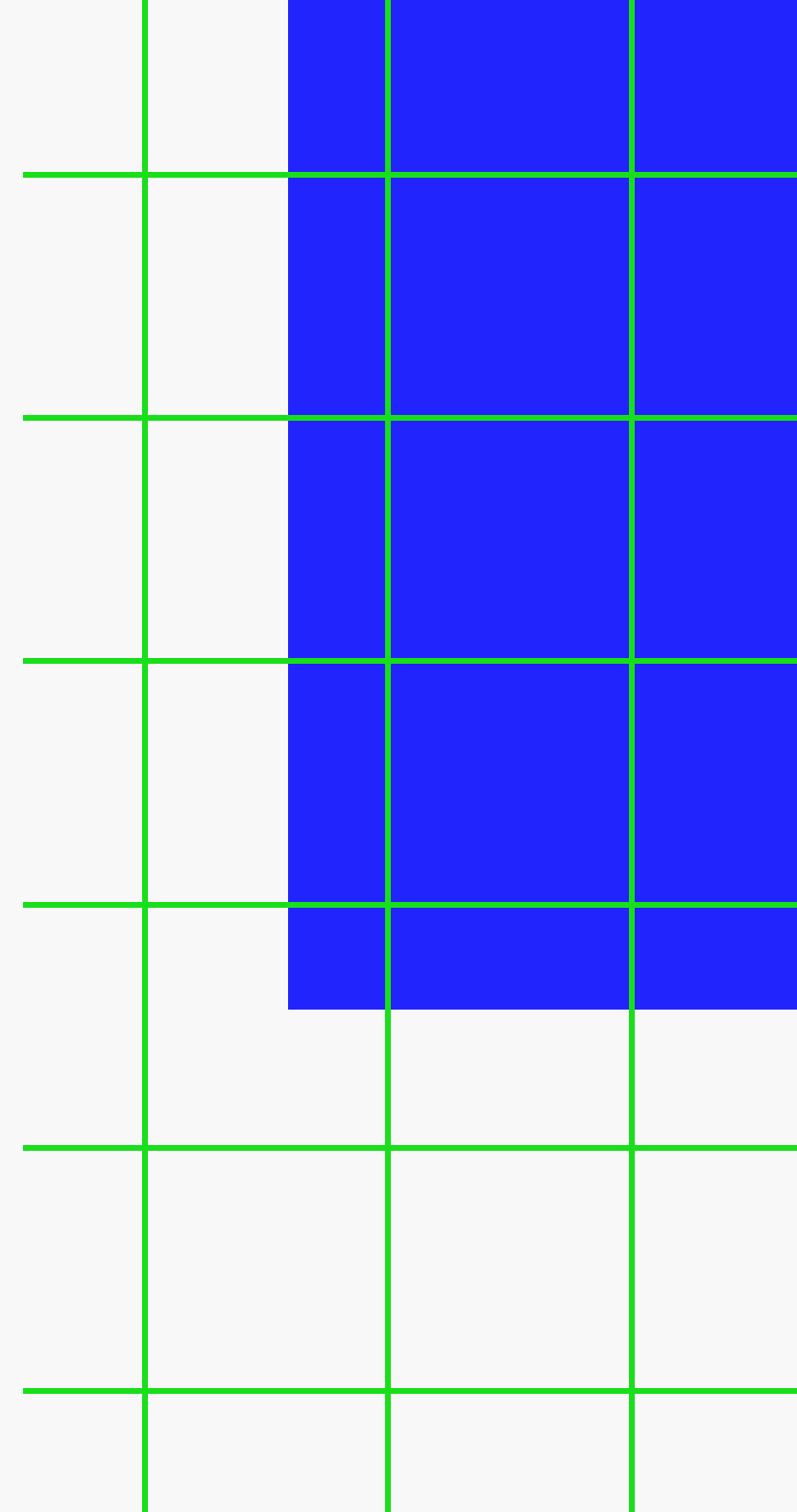
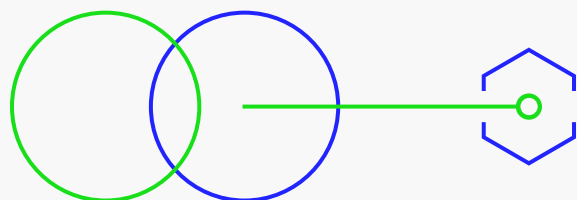


- ISAIAS SIMANCAS DIAZ
- BREINER SOLER TORRES
- AIRON MAURICIO URRUTIA

¿QUE ES UN SERVLETS?

Un Servlet es un componente de Java que se ejecuta en un servidor web y permite crear aplicaciones web dinámicas.

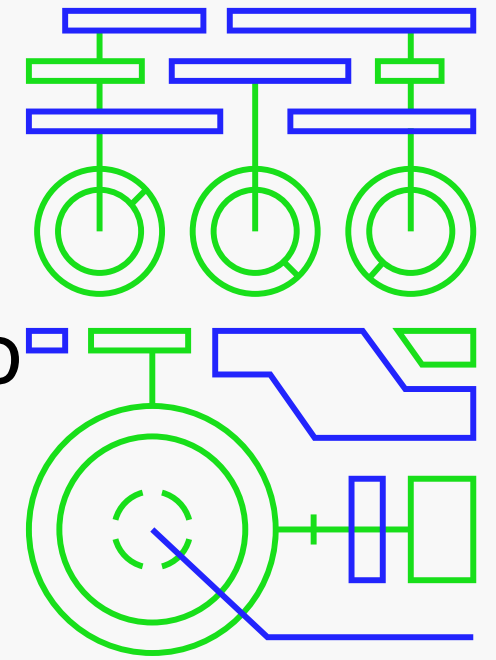
Actúa como intermediario entre el navegador del cliente y el servidor, procesando solicitudes y generando respuestas.



CÓMO FUNCIONA UN SERVLETS

El ciclo de vida de un servlet incluye tres etapas: inicialización, servicio y destrucción.

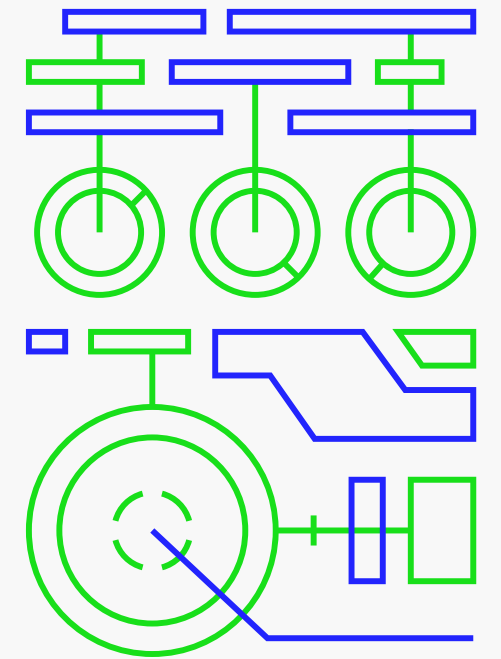
- Durante la **inicialización** (**init**), el servlet se configura y se prepara para manejar solicitudes.
- En la etapa de **servicio** (**service**), el servlet procesa las solicitudes entrantes y genera respuestas, utilizando para ello métodos como 'doGet' y 'doPost' para diferentes tipos de solicitudes HTTP.
- Por último, en la etapa de **destrucción** (**destroy**), el servlet realiza tareas de limpieza antes de ser eliminado del contenedor



EJEMPLOS PRÁCTICOS DE SERVLETS EN JAVA

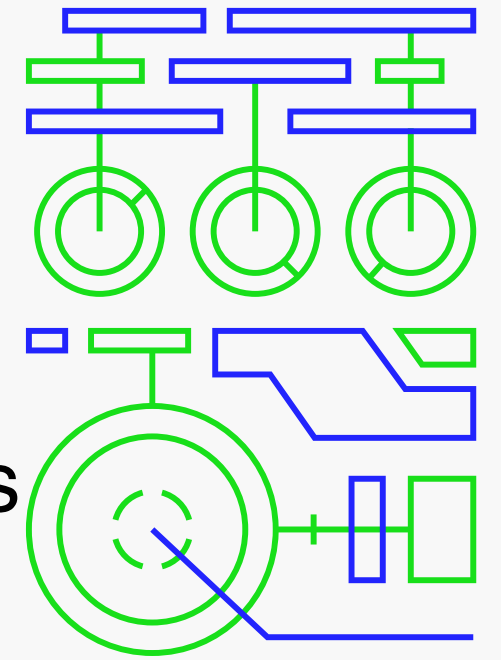
Los servlets pueden utilizarse en diferentes aplicaciones web como:

- **Gestión de formularios**, ya que un servlet puede procesar datos enviados desde un formulario HTML, validar la información y generar una respuesta adecuada.
- **Interacción con bases de datos**, para realizar consultas SQL y mostrar los resultados en una página web.
- **Generación dinámica de contenido HTML**, basado en las entradas del usuario o en datos en tiempo real.



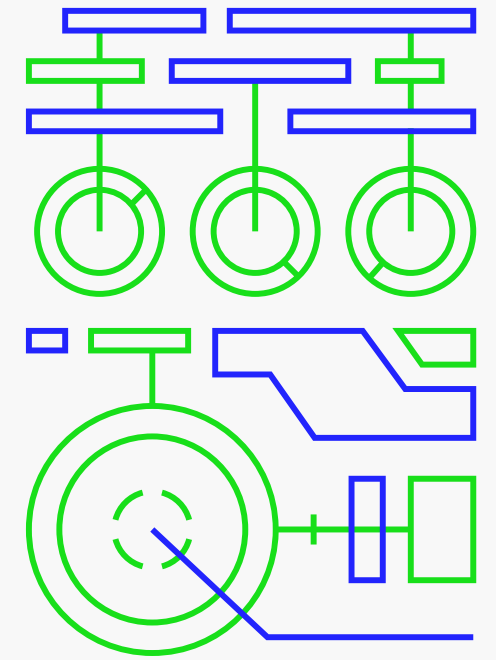
VENTAJAS DE USAR SERVLETS

- **Eficiencia y rendimiento:** Los servlets operan en un único proceso del servidor, lo que les confiere una velocidad y eficacia superiores a otras tecnología.
- **Portabilidad:** Al estar escritos en Java, pueden ejecutarse en cualquier servidor que soporte Java EE (como Apache Tomcat, GlassFish, entre otros).
- **Seguridad:** Java ofrece un entorno seguro, con un manejo robusto de excepciones, y se integra con fluidez a través de HTTPS y otras medidas de protección.



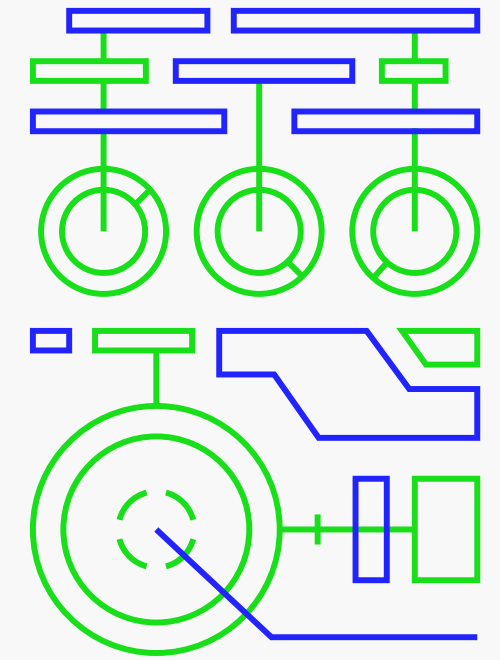
VENTAJAS DE USAR SERVLETS

- **Mantenimiento sencillo:** Al estar centralizados en el servidor, es más fácil actualizar o corregir errores sin necesidad de cambiar nada en el cliente.
- **Multihilo:** Un solo servlet puede manejar múltiples peticiones concurrentes utilizando hilos, lo que reduce la sobrecarga del servidor.



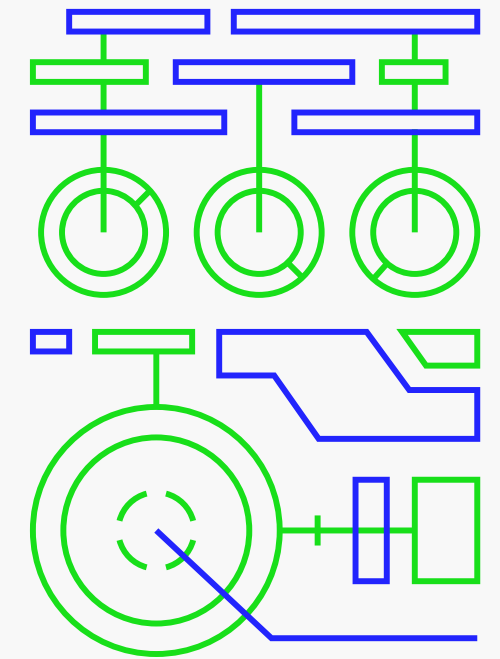
DESVENTAJAS DE USAR SERVLETS

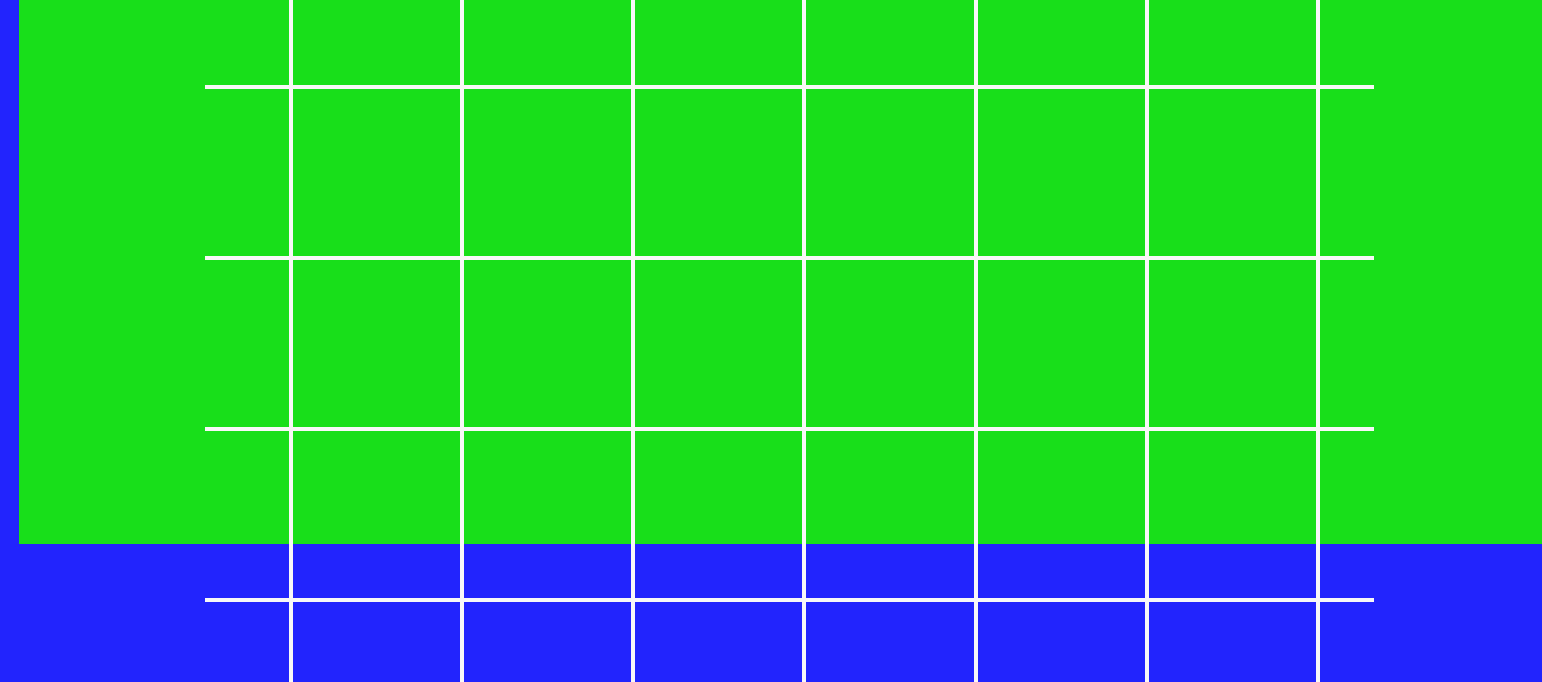
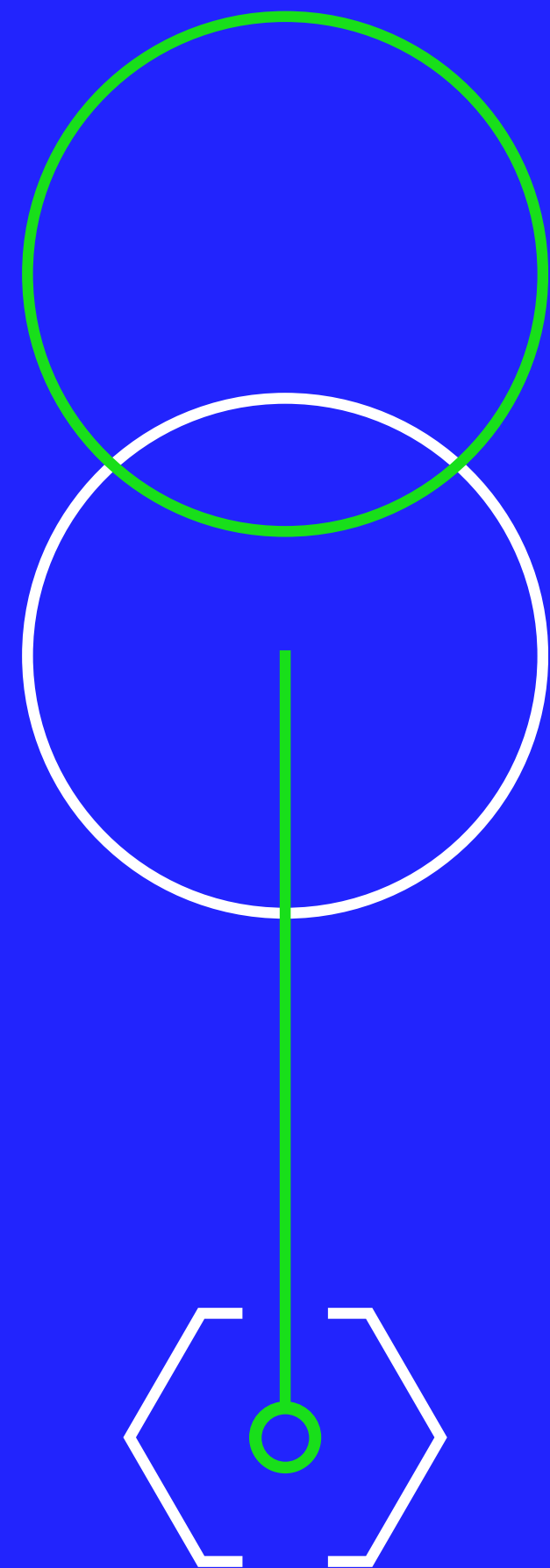
- **Es complejo en comparación con otras tecnologías similares:** Aunque evita escribir HTML desde Java (como en un servlet puro), sigue siendo más difícil de usar correctamente que otras tecnologías como PHP o templates de Python.
- **Necesita de un contenedor de servlets para ejecutarse:** Como cualquier Servlet, un JSP necesita un servidor Java como Tomcat para funcionar. No se puede abrir directamente en el navegador.
- **Difícil de mantener a gran escala:** Cuando se mezcla mucha lógica Java en archivos .jsp, el código se vuelve confuso. Esto no es culpa del Servlet en sí, sino de usar JSP para lógica de negocio, lo cual no se recomienda.



BUENAS PRÁCTICAS Y CONSEJOS PARA DESARROLLAR CON SERVLETS

- Sanitizar y validar todas las entradas del usuario les permitirá prevenir ataques de inyección SQL y XSS.
- Utilizar HTTPS les ayudará a proteger la transmisión de datos sensibles.
- Utilizar pools de conexiones para bases de datos hará que se reduzca el tiempo de conexión.
- Documentar su código y seguir un estilo de codificación consistente hará mucho más fácil el mantenimiento y la colaboración.





**MUCHAS
GRACIAS**

