**🗂️ Estructura de Carpetas Sugerida para tu Plan de Java**

Sugiero un sistema de dos niveles basado en las **Fases** de tu plan y la **Tecnología** que estás aprendiendo. Esto te permite una trazabilidad inmediata de tu progreso.

**Nivel 1: Carpeta Raíz (El Ecosistema)**

Crea una carpeta principal para todo tu aprendizaje.

* **Nombre:** Java-Estrategico-Portafolio (o algo similar que refleje el plan).

**Nivel 2: Carpetas por Fase**

Dentro de la carpeta raíz, usa las Fases de tu plan. Esto te permite medir tu progreso semanal y te ayuda a ver qué tecnologías dominaste en cada etapa.

1. **01-Bases-Solidas-Java**
2. **02-Java-Backend-Spring**
3. **03-Escalabilidad-Cloud-AWS**
4. **04-Recursos-Hacks** (Para archivos que no son código de proyecto)

**🗂️ Nomenclatura Detallada (Nivel 3: Proyectos)**

Dentro de cada carpeta de Fase, nombra tus subcarpetas de proyectos de forma **progresiva y descriptiva**.

**1. Dentro de 01-Bases-Solidas-Java**

Aquí guardas tus ejercicios de consola, enfocándote en el concepto principal del proyecto.

| Carpeta Proyecto | Contenido / Foco Clave |
| --- | --- |
| P1-Juego-Adivina-Numero | Lógica, if/else, while, Scanner. |
| P2-Agenda-Contactos-HashMap | **Colecciones**, **HashMap**, Métodos, CRUD en memoria. |
| P3-POO-Sistema-Empleados | **POO Avanzada**, Herencia, Polimorfismo, Interfaces. |
| E-Ejercicios-Diarios-Sintaxis | Ejercicios rápidos de LeetCode/HackerRank, snippets de código (Streams, Lambdas). |

Exportar a Hojas de cálculo

**2. Dentro de 02-Java-Backend-Spring**

Aquí introduces el nombre del **Framework** o la **Tecnología de persistencia** clave.

| Carpeta Proyecto | Contenido / Foco Clave |
| --- | --- |
| P1-CRUD-JDBC-Oracle | Conexión a la base de datos con JDBC (el paso previo a Spring). |
| P2-API-Barberia-Spring-JPA | **Spring Boot**, REST, **JPA/Hibernate**, Oracle DB. |
| P3-Mini-Ecommerce-Security-JWT | Ampliación de P2, añadiendo **seguridad (JWT)**. |

Exportar a Hojas de cálculo

**3. Dentro de 03-Escalabilidad-Cloud-AWS**

Aquí enfatizas las tecnologías de despliegue y calidad del código.

| Carpeta Proyecto | Contenido / Foco Clave |
| --- | --- |
| P1-API-Notas-Testing-JUnit | Añadir **Testing (JUnit/Mockito)** y documentación. |
| P2-Microservicio-Pagos-Docker | Introducción a **Docker** y **Contenedores**. |
| P3-Proyecto-Integrador-AWS | Proyecto final. Despliegue en **AWS (EC2/RDS)** y **CI/CD**. |

Exportar a Hojas de cálculo

**4. Carpeta Adicional: 04-Recursos-Hacks**

Esta es una carpeta de apoyo para todo el proceso.

* **04-Recursos-Hacks/Apuntes-Anki**: Apuntes clave o las preguntas para tus flashcards.
* **04-Recursos-Hacks/Configuracion-IDE**: Scripts de instalación, configuraciones de IntelliJ, o archivos .env de ejemplo.

**✅ Recomendación Clave de Organización**

**Utiliza Git y GitHub desde el Día 1.**

Cada una de las carpetas de proyecto (P1, P2, P3, etc.) debería ser un **repositorio de Git independiente** y ser subida a tu perfil de **GitHub**.

* **Beneficio:** No solo organizas el código localmente, sino que creas una **Hoja de Vida de Código** profesional, mostrando tu progreso con *commits* detallados.

**Tu carpeta principal en GitHub se vería así:**

* Java-Estrategico-Portafolio (Repositorio madre solo para documentación)
  + P1-Juego-Adivina-Numero (Repositorio de GitHub)
  + P2-Agenda-Contactos-HashMap (Repositorio de GitHub)
  + ...
  + P3-Proyecto-Integrador-AWS (Repositorio de GitHub)