Una solución de *software* es un conjunto de elementos enfocados al tratamiento y administración de datos e información, estos deben estar organizados y listos para cumplir sus objetivos. Teniendo en cuenta el componente formativo “las pruebas de *software*” y profundizando con otras fuentes, realice el siguiente taller con base en las preguntas orientadoras:

**Elementos a tener en cuenta en el taller:**

● ¿Qué tipos de pruebas de *software* existen? Explique sus características y beneficios.

RTA: Existen varios tipos de pruebas de software, cada uno con sus propias características y beneficios. Algunos ejemplos incluyen:

1. Pruebas unitarias: se realizan en el nivel de código fuente para asegurar que cada unidad o componente del software funciona correctamente. Estas pruebas son eficaces para detectar errores temprano en el desarrollo y son fácilmente automatizadas.
2. Pruebas de integración: se realizan para asegurar que los diferentes componentes del software funcionan correctamente juntos. Estas pruebas son útiles para detectar errores de configuración y problemas de compatibilidad.
3. Pruebas de sistema: se realizan para asegurar que el software funciona correctamente en un entorno de producción real. Estas pruebas son importantes para detectar problemas de rendimiento y escalabilidad.
4. Pruebas de aceptación: se realizan para asegurar que el software cumple con los requisitos de los usuarios y está listo para su lanzamiento. Estas pruebas son importantes para garantizar la satisfacción del cliente.
5. Pruebas de regresión: se realizan para asegurar que los cambios realizados en el software no causan problemas en las funcionalidades existentes. Estas pruebas son importantes para garantizar la estabilidad del software.

En resumen, cada tipo de pruebas de software tiene sus propios beneficios y objetivos y es importante considerar todos ellos para asegurar la calidad y fiabilidad del software.

● Según la consulta que realizó, ¿qué tipos de pruebas se adaptan mejor al proyecto de *software* que está desarrollando?

**RTA:** Se podría adaptar mejor al proyecto de su portafolio web los siguientes tipos de pruebas:

1. Pruebas unitarias: para asegurar que cada componente o sección de su portafolio web funciona correctamente.
2. Pruebas de Integración: para asegurar que los diferentes componentes de su portafolio web funcionan correctamente juntos.
3. Pruebas de Usuario: para asegurar que su portafolio web cumple con las necesidades y expectativas de los visitantes y que es fácil de navegar y atractivo.
4. Pruebas de rendimiento: para asegurar que su portafolio web funciona correctamente en diferentes dispositivos y con un gran volumen de visitas.
5. Pruebas de seguridad: para asegurar que su portafolio web es seguro y protege la información de los visitantes.

Teniendo en cuenta las pruebas mencionadas anteriormente, se podría garantizar que su portafolio web funciona correctamente y cumple con las necesidades de los usuarios. Sin embargo, es importante tener en cuenta que puede haber otras pruebas necesarias dependiendo de las características y requerimientos específicos de su portafolio web.

● Investigue e instale unas herramientas de pruebas de *software* en su computador, “una de su gusto”.

**RTA:** Existen muchas herramientas de pruebas de software disponibles, cada una con sus propias características y beneficios. Algunas herramientas populares incluyen:

1. **JUnit**: es una herramienta de pruebas unitarias para Java. Es fácil de usar y ofrece una gran cantidad de características avanzadas.
2. **TestNG**: es una herramienta de pruebas unitarias para Java que ofrece una serie de características avanzadas, como la posibilidad de ejecutar pruebas paralelas y la creación de informes detallados.
3. **Selenium**: es una herramienta de automatización de pruebas para aplicaciones web. Es compatible con varios lenguajes de programación y ofrece una gran cantidad de características avanzadas.
4. **Appium**: es una herramienta de automatización de pruebas para aplicaciones móviles. Es compatible con varios sistemas operativos móviles y lenguajes de programación.
5. **Jmeter**: es una herramienta de pruebas de rendimiento para aplicaciones web. Es fácil de usar y ofrece una gran cantidad de características avanzadas.

Para instalar una de estas herramientas en su computadora, deberá seguir las instrucciones de instalación específicas para esa herramienta. Por ejemplo, JUnit se puede descargar e instalar a través de un gestor de paquetes de Java como Maven o Gradle.

En resumen, hay muchas herramientas de pruebas de software disponibles, cada una con sus propias características y beneficios. Es recomendable investigar y comparar varias opciones antes de elegir una herramienta y seguir las instrucciones de instalación específicas para esa herramienta.

● Con la herramienta instalada, realice unas pruebas básicas de su solución de *software*, tome capturas de pantalla del proceso y anéxelas al trabajo.

● Elabore un resumen de las pruebas que realizó.