

Java SE 21 Programmer

Nro. Horas: 45

Costo para Profesionales: S/ 1000.00 (incluido IGV)

SOBRE ESTE CURSO

Este nuevo curso sobre Java SE 21 (la última versión de soporte a largo plazo) te equipa con las últimas herramientas y conocimientos para construir aplicaciones seguras, eficientes y mantenibles. Te posiciona bien para el avance en tu carrera, asegura que tus habilidades sigan siendo relevantes y te permite aprovechar al máximo las mejoras continuas en la plataforma Java.

Ofrece una cobertura en profundidad de todos los temas centrales de Java y las API más utilizadas con frecuencia. También ofrece un conjunto de ejercicios prácticos que enseñan cómo construir una aplicación Java completamente funcional desde cero.

Beneficios para ti

Después de completar este curso, deberías ser capaz de:

- Describir el enfoque de programación orientada a objetos
- Explicar la sintaxis de Java y las convenciones de codificación
- Utilizar constructos y operadores de Java
- Utilizar las API principales de Java, como Collections, Streams, IO y Concurrency
- Desplegar aplicaciones Java SE

SOBRE LA CERTIFICACION

Este es un taller de preparación para la certificación Java SE 21 Developer de Oracle (1Z0-830). Este examen consta de **50 preguntas**, una duración de **120 minutos** y se necesita mínimo un **68% de aprobación**. Es de múltiple elección.

PROPUESTA DEL CURSO

- Clases grabadas online vía zoom.
- Simulacros por tópico
- Simulacros tipo examen de 50 preguntas
- Ejercicios y material por cada clase.

TOPICOS JAVA 21

Esos son los tópicos que se estudiarán en el taller:

1. Introducción a Java
 - a. Explicar características Java y Conceptos orientados a objetos
 - b. Definir, compilar y ejecutar un programa Java
2. Tipos primitivos, Operadores y Control de Flujo
 - a. Declarar, inicializar primitivos y entender restricciones
 - b. Describir operadores Java, Asignación, operadores aritméticos y type casting
 - c. Representación de números binarios y operadores lógicos bitwise
 - d. Operadores de igualdad, relacionales, condicionales, ternario y if/else
 - e. Control de flujo usando Switch
3. Texto, Fechas, Tiempo y Objetos Numéricos
 - a. Inicialización de String y Operaciones
 - b. Indexing String, Objetos de texto mutables y Blocks
 - c. Describir Wrapper de primitivos, Clases BigDecimal y encadenamiento de métodos
 - d. Manejar fechas y tiempo
 - e. Representar lenguajes y países, formatear y parsear números, fechas y tiempo
 - f. Describir localización y formateo de clases
4. Clases y Objetos
 - a. Ejercicio de creación de clases
 - b. Documentar clases
5. Diseño de clases
 - a. Sobrecarga de métodos
 - b. Constructores
 - c. Encapsulación e inmutabilidad usando modificadores de acceso
 - d. Explicando el provisionamiento de memoria en Java y pase de parámetros
 - e. Explicando la recolección de memoria en Java

6. Herencia y uso de Records
 - a. Herencia de clases y reutilización de código usando herencia
 - b. Instanciar clases y acceso a objetos, reglas de casting de tipos de referencia
 - c. Verificando el tipo de un objeto usando el instanceof, ejecutar type casting de tipos de referencia
 - d. Referenciar código dentro de objetos y definir constructores de subclases
 - e. Sobreescritura de métodos y uso de polimorfismo
 - f. Definir clases abstractas, final, sealed e interfaces
 - g. Sobreescibir operaciones de la clase Object
 - h. Definir clases Record, Pattern matching para switch
 - i. Definir y usar métodos Factory
7. Interfaces y Genéricos
 - a. Describir interfaces
 - b. Implementar interfaces
 - c. Heredar de interfaces y usar interfaces funcionales
 - d. Usar genéricos
 - e. Explorar interfaces comunes de Java y usar el patrón composition
8. Arrays y Loops
 - a. Declarar, crear e inicializar arrays
 - b. Usar arrays multidimensionales y clases Array
 - c. Procesar arrays en un loop
 - d. Usar loops embebidos, break y continue
9. Colecciones
 - a. Introducir el Java Collection API, interfaces y clases de implementación
 - b. Crear y manejar List, Set, Deque, HashMap y contenidos
 - c. Iterar, a través de, colecciones, usar Sequenced Collections y clases Collections
 - d. Acceder a colecciones concurrentemente y prevenir corrupción de colecciones
10. Clases Nested y Expresiones Lambdas
 - a. Introducir clases Nested e Inner
 - b. Clases Inner anónimas e interfaces funcionales
 - c. Definir parámetros de una expresión lambda y body
 - d. Usar referencias a métodos y métodos no abstractos en interfaces
11. API Java Stream
 - a. Describir el API Java Stream y procesar stream pipelines

- b. Implementar interfaces funcionales y Primitive y variantes Bi-argument
 - c. Procesar Stream Pipelines y ejecutar operaciones Stream
 - d. Procesar Stream y agregar stream data
 - e. Usar collectors, ejecutar grouping y partitioning del contenido de un Stream
 - f. Describir el procesamiento de un stream paralelo, guía y restricciones
 - g. Usar Spliterator
12. Manejo de Excepciones, Logging y Debugging
- a. Java Logging
 - b. Manejo de excepciones
 - c. Debugging
13. API Java IO
- a. Explorar el API Java IO
 - b. Leer y escribir data binaria y caracteres
 - c. Usar entrada estándar, salida y clase Console
 - d. Entender la serialización, el grafo de objetos serializable y serialización de objetos
 - e. Personalizar la serialización y deserialización
 - f. Construyendo el Filesystem Paths y navegando en el Filesystem
 - g. Operar en objetos Path y Properties, manejar archivos temporales y directorios
 - h. Explorar ejemplos de I/O: trabajar con archivos Zip y acceder a recursos HTTP
14. Concurrencia en Java y Multithreading
- a. Conceptos e implementación de Hilos
 - b. Manejar el ciclo de vida de un Hilo, interrumpir y bloquear un Hilo
 - c. Wait y propiedades
 - d. Manejar e implementar un Executor Service
 - e. Discutir problemas de locking y asegurar ejecución segura de hilos
 - f. Acciones atómicas no bloqueantes, lock intrínseco y automatización de concurrencia
 - g. CPU vs IO Bound Concurrent Tasks y uso de Virtual Threads
15. Módulos y Despliegue
- a. Compilar, empaquetar y ejecutar aplicaciones no modulares
 - b. Describir el Java Module System
 - c. Crear dependencias de módulos y exportar contenido de módulo
 - d. Definir y usar Module Services
 - e. Compilar, empaquetar, ejecutar y migrar aplicaciones modulares
 - f. Crear y ejecutar una imagen runtime personalizada

16. Anotaciones
17. Conectividad a Base de Datos con JDBC
18. Seguridad Java
19. Genericos avanzado
 - a. Erasure y uso de genéricos
20. Aplicaciones Java en Oracle Cloud
 - a. Principios de diseño de aplicaciones cloud
21. Temas adicionales
 - a. Describir patrones de diseño, Java IO, File Watch Service

Prerrequisitos:

- Conocimientos de Fundamentos de programación

Formas de Pago:

- Transferencia Bancaria:

INTERBANK
CUENTA CORRIENTE SOLES
200-3001605915

CCI: 003-200-003001605915-33

- Pago con Tarjeta de Débito y Crédito
- Solicitar el pago con tarjeta de débito y crédito al +51 966 025 115 (WhatsApp).

Consultas:

Si tiene alguna consulta o comentario nos puede escribir al WhatsApp al +51 939 965 148.