

Rúbrica Analítica para Evaluación de Programación en Java - Herencia y POO

Rúbrica Analítica | Tecnología e Informática | Informática | 4 niveles

Descripción

Esta rúbrica está diseñada para evaluar el conocimiento y aplicación de conceptos de programación orientada a objetos en Java, incluyendo atributos de clase padre y subclases, herencia, sobrescritura, encapsulamiento, uso de constructores y métodos con retorno. La puntuación máxima es 50 puntos.

Rúbrica

Rúbrica Analítica para Evaluación de Programación en Java - Herencia y POO

Esta rúbrica está diseñada para evaluar el conocimiento y aplicación de conceptos de programación orientada a objetos en Java, incluyendo atributos de clase padre y subclases, herencia, sobrescritura, encapsulamiento, uso de constructores y métodos con retorno. La puntuación máxima es 50 puntos.

Criterios de Evaluación	Excelente (5 puntos)	Bueno (4 puntos)	Aceptable (3 puntos)	Bajo (1-2 puntos)
Uso correcto de atributos en clase padre	Define y utiliza adecuadamente los atributos en la clase padre, con tipos y nombres correctos y coherentes.	Define los atributos en la clase padre con pequeños errores de tipo o nombre, pero funcionales.	Define atributos en la clase padre pero con errores significativos que afectan la funcionalidad.	No define atributos en la clase padre o los define incorrectamente.
Herencia y uso adecuado de la palabra clave extends	Implementa correctamente la herencia utilizando extends para todas las subclases indicadas.	Implementa la herencia con extends pero con algunos errores menores en la sintaxis o estructura.	Usa la herencia de forma limitada o incorrecta, afectando la relación entre clases.	No utiliza herencia o la implementa de forma incorrecta o incoherente.

Criterios de Evaluación	Excelente (5 puntos)	Bueno (4 puntos)	Aceptable (3 puntos)	Bajo (1-2 puntos)
Sobrescritura de métodos en subclases	Sobrescribe correctamente los métodos heredados, respetando firmas y comportamiento esperado.	Sobrescribe métodos pero con pequeños errores en la firma o lógica.	Intenta sobrescribir métodos, pero con errores que impiden la correcta sobreescritura.	No sobrescribe métodos o lo hace de forma incorrecta.
Encapsulamiento utilizando modificador private	Aplica encapsulamiento con private en atributos y proporciona métodos getters/setters adecuados.	Aplica encapsulamiento en la mayoría de atributos, pero con omisiones o errores menores.	Aplica encapsulamiento de forma inconsistente o incorrecta.	No aplica encapsulamiento o expone atributos directamente.
Uso correcto de constructores y llamada a super	Define constructores en subclases que llaman correctamente a super para inicializar atributos heredados.	Define constructores y usa super, pero con pequeños errores en parámetros o llamada.	Define constructores pero no llama adecuadamente a super o lo hace incorrectamente.	No define constructores o no utiliza super cuando es necesario.
Definición y uso de métodos con retorno	Implementa métodos con retorno correctamente, que devuelven valores apropiados y usan tipos adecuados.	Métodos con retorno definidos, pero con errores menores en tipo o lógica.	Métodos con retorno presentes pero con errores que afectan su funcionamiento.	No implementa métodos con retorno o los implementa incorrectamente.
Organización y legibilidad del código	Código bien organizado, con indentación correcta y comentarios claros que facilitan su comprensión.	Código organizado pero con algunos problemas menores de formato o falta de comentarios.	Código poco organizado y difícil de seguir, con escasos comentarios.	Código desordenado, sin indentación ni comentarios que dificultan su comprensión.
Correcta compilación y ejecución sin errores	El programa compila y se ejecuta sin errores, mostrando resultados esperados.	El programa compila y se ejecuta con errores menores que no afectan la funcionalidad principal.	El programa compila pero presenta errores en la ejecución que afectan parte de la funcionalidad.	El programa no compila o falla gravemente en la ejecución.