

Rúbrica para Evaluar Desarrollo Individual de un Programa Básico

Rúbrica Escalar | Tecnología e Informática | Informática | 5 niveles

Descripción

Esta rúbrica evalúa el desarrollo individual de un programa básico aplicando variables, tipos de datos y operadores en estudiantes de educación media (15-17 años). La evaluación se realiza en una escala numérica basada en criterios claros y específicos que reflejan la calidad y precisión en el uso de conceptos fundamentales de programación.

Rúbrica

Rúbrica para Evaluar Desarrollo Individual de un Programa Básico

Esta rúbrica evalúa el desarrollo individual de un programa básico aplicando variables, tipos de datos y operadores en estudiantes de educación media (15-17 años). La evaluación se realiza en una escala numérica basada en criterios claros y específicos que reflejan la calidad y precisión en el uso de conceptos fundamentales de programación.

Aspectos a Evaluar	Criterios de Evaluación	Puntuación
Definición y uso de variables	Excelente (90%+): Variables correctamente definidas y usadas apropiadamente en todo el programa. Bueno (80%+): Variables definidas correctamente con mínimos errores en su uso. Aceptable (50%+): Variables definidas, pero con varios errores en su uso o inconsistencias. Pobre (<50%): Variables mal definidas o no utilizadas correctamente.	Excelente: 4 Bueno: 3 Aceptable: 2 Pobre: 1
Aplicación de tipos de datos	Excelente (90%+): Tipos de datos apropiados y consistentes con el contexto del programa. Bueno (80%+): Tipos de datos mayormente adecuados con algunos errores menores. Aceptable (50%+): Tipos de datos usados de forma inconsistente o inadecuada en varias partes. Pobre (<50%): Uso incorrecto o confuso de tipos de datos.	Excelente: 4 Bueno: 3 Aceptable: 2 Pobre: 1

Aspectos a Evaluar	Criterios de Evaluación	Puntuación
Uso correcto de operadores	Excelente (90%+): Operadores aplicados correctamente para realizar cálculos y comparaciones. Bueno (80%+): Uso adecuado de operadores con pocos errores. Aceptable (50%+): Uso irregular de operadores que afecta la lógica del programa. Pobre (<50%): Operadores mal usados o ausentes, causando errores en el programa.	Excelente: 4 Bueno: 3 Aceptable: 2 Pobre: 1
Organización y estructura del código	Excelente (90%+): Código organizado, legible y estructurado de forma coherente. Bueno (80%+): Código generalmente organizado con algunas pequeñas inconsistencias. Aceptable (50%+): Organización del código pobre que dificulta la comprensión. Pobre (<50%): Código desorganizado y difícil de entender.	Excelente: 4 Bueno: 3 Aceptable: 2 Pobre: 1
Corrección funcional del programa	Excelente (90%+): Programa funciona correctamente sin errores en todas las pruebas. Bueno (80%+): Programa funciona con pequeños errores que no afectan resultados principales. Aceptable (50%+): Programa tiene errores que afectan algunas funcionalidades. Pobre (<50%): Programa no funciona o presenta errores graves.	Excelente: 4 Bueno: 3 Aceptable: 2 Pobre: 1
Aplicación de comentarios y documentación	Excelente (90%+): Comentarios claros y precisos que facilitan la comprensión del código. Bueno (80%+): Comentarios presentes pero con algunas omisiones o poca claridad. Aceptable (50%+): Comentarios escasos o poco útiles. Pobre (<50%): Ausencia de comentarios o documentación.	Excelente: 4 Bueno: 3 Aceptable: 2 Pobre: 1
Creatividad y originalidad	Excelente (90%+): Programa muestra ideas originales y solución creativa al problema. Bueno (80%+): Algunas ideas originales presentes. Aceptable (50%+): Solución básica sin elementos creativos. Pobre (<50%): Copia de ejemplos sin aportes propios.	Excelente: 4 Bueno: 3 Aceptable: 2 Pobre: 1

Aspectos a Evaluar	Criterios de Evaluación	Puntuación
Entrega y cumplimiento de requisitos	Excelente (90%+): Entrega puntual y cumple con todos los requisitos del proyecto. Bueno (80%+): Entrega casi puntual y cumple la mayoría de requisitos. Aceptable (50%+): Entrega tardía o con varios requisitos no cumplidos. Pobre (<50%): No entrega o incumple la mayoría de requisitos.	Excelente: 4 Bueno: 3 Aceptable: 2 Pobre: 1