

PREGUNTAS PROGRA

1. ¿Qué es la abstracción en POO?

R: Oculta la complejidad interna y expone solo las características esenciales al usuario.

2. ¿Cómo se implementa la abstracción en C#?

R: Utilizando métodos abstractos (firma sin implementación) y propiedades encapsuladas.

3. ¿Qué beneficios aporta la abstracción?

R: Modularidad, reutilización de código y facilidad de uso.

4. ¿Qué es la herencia en POO?

R: Permite a una clase heredar atributos y métodos de otra clase, creando jerarquías.

5. ¿Cómo se implementa la herencia en C#?

R: Utilizando la palabra clave `:`, indicando la clase base de la que hereda la clase derivada.

6. ¿Qué ventajas ofrece la herencia?

R: Reutilización de código, jerarquías de clases y especialización.

7. ¿Qué es el polimorfismo en POO?

R: Permite que un objeto tome diferentes formas en tiempo de ejecución.

8. ¿Cómo se implementa el polimorfismo en C#?

R: Utilizando la sobrecarga de métodos (mismo nombre, diferente implementación).

9. ¿Qué beneficios aporta el polimorfismo?

R: Código más flexible, adaptable y reutilizable.

10. ¿Qué es la encapsulación en POO?

R: Oculta los detalles de implementación y expone solo propiedades y métodos públicos.

11. ¿Cómo se implementa la encapsulación en C#?

R: Utilizando modificadores de acceso (public, private, protected, internal).

12. ¿Qué ventajas ofrece la encapsulación?

R: Protección de datos, modularidad y mayor control.

13. ¿Qué son las interfaces en C#?

R: Conjuntos de métodos abstractos que definen un comportamiento que las clases deben implementar.

14. ¿Cómo se diferencian las interfaces de las clases abstractas?

R: Las interfaces no contienen implementaciones de métodos, solo definen las firmas.

15. ¿En qué escenarios son útiles los delegados en C#?

R: Transferencia de métodos como argumentos, creación de eventos y comunicación asíncrona.