

Comenzado el	viernes, 12 de noviembre de 2021, 17:52
Estado	Finalizado
Finalizado en	viernes, 12 de noviembre de 2021, 17:59
Tiempo empleado	6 minutos 30 segundos
Calificación	5,00 de 5,00 (100%)

Pregunta 1

Correcta

Se puntúa 1,00 sobre 1,00

Seleccione la afirmación correcta

Seleccione una:

- ☒ a. En ECMAScript, cada objeto hereda comportamiento y estado de su prototipo. ✓
- ☐ b. En ECMAScript, cada objeto hereda estado de su prototipo. El comportamiento no se hereda.
- ☐ c. Al no ser basado en clases sino en prototipos, ECMAScript no implementa el concepto de herencia.
- ☐ d. En ECMAScript, cada objeto hereda comportamiento de su prototipo. El estado no se hereda.

La respuesta correcta es: En ECMAScript, cada objeto hereda comportamiento y estado de su prototipo.

Pregunta 2

Correcta

Se puntúa 1,00 sobre 1,00

Seleccione la afirmación correcta

Seleccione una:

- ☒ a. ECMAScript es un lenguaje dinámico, en el que no se indica explícitamente el tipo de las variables. ✓
- ☐ b. ECMAScript puede utilizarse tanto como un lenguaje dinámico, en el que no es necesario indicar explícitamente el tipo de las variables o como un lenguaje fuertemente tipado.
- ☐ c. ECMAScript es un lenguaje fuertemente tipado, en el que se declara el tipo de cada variable y parámetro, utilizando clases.
- ☐ d. ECMAScript es un lenguaje fuertemente tipado, en el que se declara el tipo de cada variable y parámetro, utilizando interfaces.

La respuesta correcta es: ECMAScript es un lenguaje dinámico, en el que no se indica explícitamente el tipo de las variables.

Pregunta 3

Correcta

Se puntúa 1,00 sobre 1,00

Seleccione la afirmación correcta

Seleccione una:

- ☒ a. A diferencia de Java y Smalltalk, que son lenguajes orientados a objetos basados en clases, ECMAScript es basado en prototipos. ✓
- ☐ b. ECMAScript es un lenguaje basado en clases si se lo usa en el navegador, y basado en prototipos si se lo usa en el servidor.
- ☐ c. ECMAScript es un lenguaje basado en clases si se lo usa en el servidor, y basado en prototipos si se lo usa en el navegador.
- ☐ d. Al igual que Java y Smalltalk, que son lenguajes orientados a objetos basados en clases, ECMAScript es también basado en clases.

La respuesta correcta es: A diferencia de Java y Smalltalk, que son lenguajes orientados a objetos basados en clases, ECMAScript es basado en prototipos.

Pregunta **4**

Correcta

Se puntúa 1,00 sobre 1,00

Seleccione la afirmación correcta

Seleccione una:

- ☒ a. En Smalltalk las clases también son objetos, tienen sus propios atributos y métodos. El new, por ejemplo, es un mensaje de clase. ✓
- ☐ b. En Smalltalk, no existen las clases. El lenguaje se basa en prototipos que se clonan y especializan.
- ☐ c. Las clases de Smalltalk son objetos especiales, que solo entienden mensajes predefinidos para crear instancias. No es posible extender una clase para que entienda otros mensajes.
- ☐ d. En Smalltalk, al igual que en java, tenemos métodos estáticos que se definen en las clases y se invocan directamente.

La respuesta correcta es: En Smalltalk las clases también son objetos, tienen sus propios atributos y métodos. El new, por ejemplo, es un mensaje de clase.

Pregunta **5**

Correcta

Se puntúa 1,00 sobre 1,00

Seleccione la afirmación correcta

Seleccione una:

- ☐ a. Smalltalk hace una clara diferencia entre las partes de nuestra aplicación implementadas en objetos, por nosotros, y los elementos base del lenguaje (por ejemplo, librerías) que han sido implementados combinando paradigmas (objetos, funcional, procedural).
- ☐ b. En Smalltalk casi todo se implementa con objetos. Los tipos primitivos (Integer, String, etc.) y las estructuras de control no.
- ☐ c. En Smalltalk todo se implementa con objetos. Sin embargo, la implementación de los objetos de las librerías base (como Boleanos, Strings, colecciones, etc.) no puede modificarse.
- ☒ d. En Smalltalk todo se implementa con objetos y está abierto a modificación. Incluso lo que comúnmente conocemos como estructuras de control (como el if, while, etc.) se implementa como envíos de mensajes a objetos. ✓

La respuesta correcta es: En Smalltalk todo se implementa con objetos y está abierto a modificación. Incluso lo que comúnmente conocemos como estructuras de control (como el if, while, etc.) se implementa como envíos de mensajes a objetos.

◀ [Template para solicitar certificado de parcial](#)

Ir a...

[Template para solicitar certificado de parcial](#) ▶