| **EVIDENCIAS DE COMPETENCIA**  **Demostración de conocimientos y estrategias cognitivas** | |
| --- | --- |
| **Resultados a comprobar** | |
| CE1.2 Relacionar la funcionalidad del componente software a desarrollar con las técnicas de desarrollo estructurado estándares para cumplir la funcionalidad del componente software.  CE1.3 Formular estructuras de datos y flujo de control mediante lenguajes de guion según la funcionalidad del componente software a desarrollar  CE1.5 Documentar el componente software desarrollado según especificaciones de diseño.  CE2.1 Identificar los formatos estándares de distribución y utilización de los componentes multimedia, audio, vídeo, ilustraciones, fotografías, entre otros para su integración en documentos del entorno cliente  CE3.1 Analizar los requisitos de uso de componentes software para ser utilizados por el documento en el entorno del cliente. | |
| **INDICADORES DE LOGRO** | **SISTEMA DE VALORACIÓN** |
| Identificación de la respuesta correcta entre varias alternativas | En los ítems tipo test, con cuatro respuestas posibles, se aplicará la siguiente fórmula:  **P.D= A-(E/n-1)** |
| **METODOS E INSTRUMENTOS DE EVALUACION** | |
| Formulación de preguntas mediante una prueba objetiva de 60 ítems distribuidos como sigue:   1. 30 tipo test, con cuatro alternativas, en donde sólo una es la correcta.   Antes de responder, hay que leer atentamente la pregunta y las posibles respuestas, con el fin de evitar errores.  Marca con un círculo la letra de la respuesta correcta. En caso de equivocarse, tacha la marcada con una cruz y marcar, con un nuevo círculo, la correcta.  **Su duración total será de 1 hora aproximadamente.** | |

1. **Una variable es…**

a) Un dato.

b) Un valor.

c) Un tipo de dato

d) Un objeto

1. **La palabra reservada *const* indica que su valor…**
2. Varía constantemente
3. Es constante, no varía.
4. Es variable
5. Ninguno de los casos anteriores

**3. En la expresión 3 + 2 \* 5 el resultado será:**

a) 13

b) 25

c) 15

d) Undefined

**4. En Javascript, Math es**

a) Un objeto

b) Una función

c) Una expresión

d) Un método

**5. Tenemos una String *casa = “villa”.* En este caso *casa.length* es**

a) Una función

b) Un método

c) Una propiedad

d) Una variable

**6. El siguiente código: *let a === 3; console.log(a);* devolverá**

a) 3

b) true

c) false

d) error

**7. En una sentencia en que todos los operandos son &&, para que el resultado sea true:**

a) Debe cumplirse al menos una de las condiciones

b) No debe cumplirse ninguna condición

c) Deben cumplirse todas las condiciones

d) El resultado nunca será true

**8. El operador ternario**

a) Se usa para comparar más de tres condiciones. De ahí su nombre.

b) Es un operador alternativo al uso de un condicional

c) Es un sustituto más reducido para operaciones matemáticas con enteros

d) Es un operador de negación

**9. Los bucles *for* sirven para**

a) Iterar indefinidamente sobre un bloque de código

b) Iterar un número de veces concreto sobre un bloque de código

c) Iterar siempre desde 0 hasta un número de veces sobre un bloque de código

d) Nada. Ya no se usan

**10. El siguiente código: *a = 1 < 2 < 3;* dará como resultado**

a) Error

b) true

c) false

d) NaN

**11. La siguiente declaración*: a = [ ];* crea**

a) Una variable de valor cero

b) Un objeto vacío

c) Un objeto nulo

d) Un array vacío

**12. ¿Es posible crear este array?:**

**lista = [1, 2, 3, true, ‘blanco’, -15]**

a) Si. Javascript lo permite

b) No, porque hay diferentes tipos de datos.

c) No, porque no se ha inicializado

d) Sí. Pero antes debe indicarse que es un array mixto.

**13. El método *indexOf* se usa para**

a) Calcular el índice de un array

b) Devolver el contenido de una posición en un array

c) Devolver la posición en un array del argumento que se le pasa.

d) Devolver el contenido de una posición en un objeto literal

**14. Utilizamos el destructuring para:**

a) Crear una lista con los contenidos de un objeto literal

b) Crear una lista con los contenidos de un array

c) Extraer determinados valores de un array a unas variables

d) Eliminar valores de un array

**15. El *spread operator* permite (entre otras cosas)**

a) Hacer un copia de un array

b) Eliminar los elementos de un array

c) Ordenar los elementos de un objeto

d) Crear un objeto a partir de un array

**16. Un método es equivalente a**

a) Una variable

b) Un array.

c) Un objeto.

d) Una función

**17. Un String realmente se comporta como:**

a) Una variable de tipo booleana

b) Un array

c) Un objeto

d) Un prototipo

**18. La expresión *true || false* devuelve**

a) Not

b) Si

c) True

d) False

**19. La expresión *!true && !false* devuelve**

a) True

b) False

c) Not

d) Si

**20. La expresión *haEntrado? true:false;*  devuelve**

a) True si *haEntrado* es falso. False si *haEntrado* es verdadero

b) True si *haEntrado* es verdadero. False si *haEntrado* es falso

c) True siempre

d) False siempre

**21. Para verificar que un array contiene determinado elemento usamos el método**

a) contains

b) indexOf

c) includes

d) parent

**22. En un elemento *flex* la propiedad *flex-basis* equivale a**

a) El alto del elemento

b) El ancho del elemento

c) El alto y el ancho del elemento

d) La proporción entre el ancho y el alto

**23. El grid es una rejilla para la colocar los elementos dentro de un contenedor de forma**

a) unidimensional

b) bidimensional

c) tridimensional

d) Ninguna de las anteriores

**24. En Javascript, Math.random() es**

a) Un objeto

b) Una declaración

c) Una expresión

d) Un método

**25. Javascript devuelve en consola el mensaje undefined si…**

a) Una variable se ha creado, pero tiene valor null cuando la usamos.

b) Una variable se ha creado, pero no se ha inicializado cuando la usamos

c) Una variable se ha creado con el *keyword* *var*

d) Una constante es *NaN*

**26. Para convertir un string que contiene un número a valor numérico se usa…**

1. Stringify
2. parseFloat
3. parseInt
4. No se puede

**27. El método pop() en Javascript…**

1. Se usa para añadir un elemento al final de un array
2. Se usa para añadir un elemento al principio de un array
3. Se usa para eliminar un elemento al principio de un array
4. Se usa para eliminar un elemento al final de un array

**28. La declaración break…**

1. Permite pausar un iterador
2. Permite saltar al siguiente elemento en un iterador
3. Permite salir del iterador
4. Permite continuar una iteración pausada

**29. El operador % sirve para…**

1. Calcular el porcentaje del valor numérico que se le pasa.
2. Calcular el porcentaje de la diferencia de los valores que se le pasan.
3. Calcular el resto de la división de los valores que se le pasan.
4. No existe

**30. ¿Cuál es la salida de este código?**

*var num = 8; var num = 10; console.log(num);*

1. 8
2. 10
3. NaN
4. Undefined