Programación con JavaScript I

Reactivos

Pregunta 1

¿Cuál es el lenguaje de programación más utilizado en la web?

Opciones:

a. Java.
b. C++.
c. C#.
d. JavaScript.

Pregunta 2

¿Qué es un entorno de desarrollo?

Opciones:

a. Un software que permite compilar el código.
b. Un software que permite ver la página web.
c. Un software que permite escribir y editar el código.
d. Un software que sirve para el diseño de una página web.

Pregunta 3

Elige el elemento que se necesita para ejecutar una página web.

Opciones:

a. IDE.
b. Navegador.
c. VCS.
d. Git.

Pregunta 4

Selecciona la tecla que te permite tener acceso a las herramientas del desarrollador en el navegador.

Opciones:

a. F12.
b. F11.
c. F10.
d. F9.

Pregunta 5

Elige la etiqueta que se utiliza en HTML para colocar el código JavaScript.

Opciones:

a. <p></p>
b. <head></head>
c. <body></body>
d. <script></script>

Pregunta 6

Selecciona el símbolo que representa una igualdad en programación.

Opciones:

a. !=
b. <
c. >=
d. ==

Pregunta 7

Establece cuál es el símbolo para el operador lógico or.

Opciones:

a. ||
b. &&
c. ??
d. !

Pregunta 8

¿Cuál es la condición más común en programación?

Opciones:

a. Switch.
b. If.
c. While.
d. For.

Pregunta 9

Selecciona la opción en la cual el bloque de código dentro de este bucle se ejecutará mientras se cumpla la condición.

Opciones:

a. Do/while.
b. If.
c. While.
d. Switch.

Pregunta 10

Elige la sentencia más potente que permite resumir en una línea la forma de un bucle while.

Opciones:

a. Break.
b. For/in.
c. For/of.
d. For.

Pregunta 11

Selecciona el elemento que es un bloque de código que se define una vez, pero que puede invocarse tantas veces como se desee.

Opciones:

a. Condicionales.
b. Bucles.
c. Operador.
d. Función.

Pregunta 12

Selecciona la opción que establece la sintaxis básica para una función.

Opciones:

a.
```
const cuadrado = function(numero) { return numero*numero; };
let diezcuadrado = (function(x) {return x*x;}(10));
```

b.
```
(function () {
var nombre = "Juan";
})();
console.log(nombre);
// Devuelve Uncaught ReferenceError: nombre is not defined
```

c.
```
function nombreDeLaFunción(parámetros) {
sentencias}
```

d.
```
const alumnos = {
personas: [],
agrega(...personas) {
this.personas.push(...personas);
},
saludo(i) {
return `Hola mi nombre es ${this.personas[i]}`;
}
};
alumnos.agrega('Hugo', 'Paco', 'Luis');
alumnos.saludo(0); // => 'Hola mi nombre es Hugo'
```

Pregunta 13

Elige la opción que muestra el ejemplo de una función de flecha.

Opciones:

a. function multiplica() {
   return numero1 * numero2;
   }

b. const saludo = (nombre) => {
   return `Hola ${nombre}`;
   }
   console.log(saludo('Pedro')) // => Devuelve Hola Pedro
   // Compactando un poco más el código.
   const salu2 = (nombre) => `Hola ${nombre}`;
   console.log(salu2('Pedro')); // => Devuelve Hola Pedro

c. const alumnos = {
   personas: [],
   agrega(...personas) {
   this.personas.push(...personas);
   },
   saludo(i) {
   return `Hola mi nombre es ${this.personas[i]}`;
   }
   };
   alumnos.agrega('Hugo', 'Paco', 'Luis');
   alumnos.saludo(0); // => 'Hola mi nombre es Hugo'

d. (function () {
   var nombre = "Juan";
   })();
   console.log(nombre);
   // Devuelve Uncaught ReferenceError: nombre is not defined

---

Pregunta 14

Selecciona el segmento de código correcto para definir una función dentro de otra función.

Opciones:

a. function factorial(num) {
   if (num <= 1) return 1; return num * factorial(num-1);
   }

b. const saludo = (nombre) => {
   return `Hola ${nombre}`;
   }
   console.log(saludo('Pedro')) // => Devuelve Hola Pedro
   // Compactando un poco más el código.
   const salu2 = (nombre) => `Hola ${nombre}`;
   console.log(salu2('Pedro')); // => Devuelve Hola Pedro

c. (function () {
   var nombre = "Juan";
   })();
   console.log(nombre);
   // Devuelve Uncaught ReferenceError: nombre is not defined

d. const cuadrado = function(numero) { return numero*numero; };
   let diezcuadrado = (function(x) {return x*x;}(10));

Pregunta 15

Elige el segmento de código que representa una función anidada.

Opciones:

a. const cuadrado = function(numero) { return numero*numero; };
   let diezcuadrado = (function(x) {return x*x;}(10));

b. function factorial(num) {
   if (num <= 1) return 1; return num * factorial(num-1); }

c. function multiplica() {
   return numero1 * numero2; }

d. function getMarcador() {
   //Variables definidas en ámbito local
   let numero1 = 10,
   numero2 = 7;

Pregunta 16

¿Cuáles son los tipos de datos más grandes que maneja JavaScript?

Opciones:

a. Objetos.
b. Primitivos.
c. Clases.
d. Objetos y primitivos.

Pregunta 17

Establece la sintaxis de la notación literal, la cual se utiliza para crear los objetos de forma abreviada.

Opciones:

a. const new objeto={}; //Esto es un objeto vacío
b. const new object objeto={}; //Esto es un objeto vacío
c. const objeto={}; //Esto es un objeto vacío
d. const object objeto={}; //Esto es un objeto vacío

Pregunta 18

Identifica la palabra que completa la siguiente frase:

Todo objeto tiene___________.

Opciones:

a. Elementos.
b. Propiedades.
c. Componentes.
d. Valores.

Pregunta 19

Elige el segmento de código que muestra la manera correcta para agregar propiedades a un objeto.

Opciones:

a. const jugador={};
    jugador.nombre="Ryu";
    jugador.poder=10;
    jugador.vida=99;

b. const jugador={};
    jugador.nombre="Ryu"
    jugador.poder=10
    jugador.vida=99;

c. const jugador={};
    jugador.nombre="Ryu";
    jugador.poder=10;
    jugador.vida=99

d. const jugador={}
    jugador.nombre="Ryu";
    jugador.poder=10;
    jugador.vida=99;

Pregunta 20

Elige el segmento de código que muestra la manera correcta para concatenar una cadena a un objeto.

Opciones:

a. console("Mi jugador favorito es " jugador); //"Mi jugador favorito es Ryu (4/6)"
b. log("Mi jugador favorito es "+ jugador); //"Mi jugador favorito es Ryu (4/6)"
c. consol.log("Mi jugador favorito es "+ jugador); //"Mi jugador favorito es Ryu (4/6)"
d. console.log("Mi jugador favorito es "+ jugador); //"Mi jugador favorito es Ryu (4/6)"

Pregunta 21

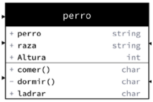
¿Cuál es el objetivo principal al utilizar la programación orientada a objetos?

Opciones:

a. Dividir a los programas para tener un mejor orden.
b. Dividir a los programas de tal forma que cada desarrollador pueda hacer su parte.
c. Dividir a los programas para tener una mejor organización en el proyecto.
d. Dividir a los programas en piezas más pequeñas y lograr que cada una de estas sea responsable de gestionar su propio estado.

Pregunta 22

Selecciona qué representa la siguiente imagen:



Opciones:

a. Elementos.
b. Objetos.
c. Métodos.
d. Propiedades.

Pregunta 23

Selecciona el fragmento de código que declara un método.

Opciones:

a.
```
let persona = {};
persona.hablar = function(linea) {
console.log(`la persona dice '${linea}'`);
};
```

b.
```
let persona = {};
persona.hablar = function(linea) {
console.log(`la persona dice '${linea}'`);
};
persona.hablar("Estoy vivo.");
// → La persona dice 'Estoy vivo.'
```

c.
```
persona = {};
persona.hablar = function(linea) {
console.log(`la persona dice '${linea}'`);
};
persona.hablar("Estoy vivo.");
// → La persona dice 'Estoy vivo.'
```

d.
```
let = {};
persona.hablar = function(linea) {
console.log(`la persona dice '${linea}'`);
};
persona.hablar("Estoy vivo.");
// → La persona dice 'Estoy vivo.'
```

Pregunta 24

Elige cuál es la palabra clave que vincula al objeto con el método cuando este es invocado.

Opciones:

a. This.
b. Those.
c. That.
d. Which.

Pregunta 25

Identifica la opción que completa la siguiente frase:

Para instanciar una clase, debes crear un:

Opciones:

a. Objeto vacío.
b. Método que derive del prototipo adecuado.
c. Método vacío.
d. Objeto que derive del prototipo adecuado.

Pregunta 26

Selecciona la manera en que se puede afrontar un problema de programación.

Opciones:

a. Lenguaje de programación.
b. Tipo de programación.
c. Lenguaje de programación.
d. Paradigma de la programación.

Pregunta 27

Identifica el paradigma de la programación que se basa en dar instrucciones de cómo hacer las cosas (receta de cocina).

Opciones:

a. Programación orientada a objetos.
b. Programación declarativa.
c. Programación imperativa o procedimental.
d. Programación funcional.

Pregunta 28

Selecciona la opción que representa un ejemplo de programación funcional.

Opciones:

a. JavaScript.
b. HTML.
c. C++
d. C#

Pregunta 29

Las funciones de ______________ tienen las siguientes características:

- Se pueden asignar a variables.
- Se pueden pasar como parámetro de otras funciones (*callbacks*).
- Se pueden retornar desde otras funciones.

Opciones:

a. Primera clase.
b. Segunda clase.
c. Tercera clase.
d. Cuarta clase.

Pregunta 30

Elige la opción que muestra un ejemplo de funciones de orden superior.

Opciones:

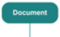
a.
```
const tiempo = () => Date.now();
tiempo();//1664171354245
tiempo();//1664171354251
```

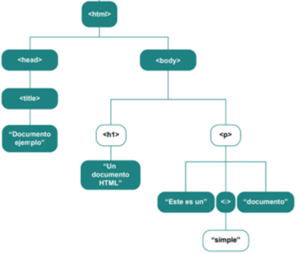
b.
```
const suma= (a,b)=> (a+b);
suma(2,5);// =>7
suma(2,5);// =>
```

c.
```
const sumar=(x,y) => console.log(x + y);
setTimeout(()=>sumar(2,2),1000);
sumar(4,4);
```

d.
```
const arreglo = [1, 2, 3];
const mapeo = arreglo.map(function(elemento){
return elemento + 1;
}); // mapeo es [2, 3, 4]
const reducido = arreglo.reduce(function(elem1, elem2){
return elem1 + elem2;
}); // reducido es 6
const filtrado = arreglo.filter(function(elemento){
return elemento !== 1;
}); // filtrado is [2, 3]
console.log(mapeo);
console.log(reducido);
console.log(filtrado);
```

Pregunta 31





La imagen anterior representa:

Opciones:

a. El contenido de un HTML.
b. La constitución de un HTML.
c. El cuerpo HTML.
d. Un API DOM.

Pregunta 32

Selecciona el fragmento de código que crea un nuevo objeto HTML usando API DOM.

Opciones:

a.
```
<p> Mi perro Rufo: </p>
<p> <img id = "rufo" src = "img/rufo.png"> </p>
<script>
let perro = document.getElementById ("rufo");
console.log (perro.src);
</script>
```

b.
```
<p><a href="http://cdctecmilenio.mx">Lernit</a></p>
<script>
let enlace = document.body.getElementsByTagName("a")[0];
console.log(enlace.href);
</script>
```

c.
```
<html>
<head>
<script>
// ejecuta la función cuando carga la página web
window.onload = function () {
// crea un par de elementos HTML en la página
var cabecera = document.createElement(“h1”);
var textoCabecera = document.createTextNode(“hola mundo”);
cabecera.appendChild(textoCabecera);
document.body.appendChild(cabecera);
}
</script>
</head>
<body>
</body>
```

d. `<body onload="window.alert('Bienvenido!');">`

Pregunta 33

Selecciona el fragmento de código que asigna atributos usando API DOM.

Opciones:

a.
```
<html>
<head>
<script>
// ejecuta la función cuando carga la página web
window.onload = function () {
// crea un par de elementos HTML en la página
var cabecera = document.createElement(“h1”);
var textoCabecera = document.createTextNode(“hola mundo”);
cabecera.appendChild(textoCabecera);
document.body.appendChild(cabecera);
}
</script>
</head>
```

<body>
</body>

b. <p> Mi perro Rufo: </p>
<p> <img id = "rufo" src = "img/rufo.png"> </p>
<script>
let perro = document.getElementById ("rufo");
console.log (perro.src);
</script>

c. <p>Lernit</p>
<script>
let enlace = document.body.getElementsByTagName("a")[0];
console.log(enlace.href);
</script>

d. <body onload="window.alert('Bienvenido!');">

Pregunta 34

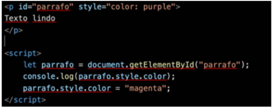
Elige el método que se utiliza para reemplazar un nodo hijo por otro.

Opciones:

a. replaceChild.
b. insertBefore.
c. repChild.
d. rChild.

Pregunta 35

Selecciona la opción que describe el resultado del siguiente código:



```
<p id="parrafo" style="color: purple">
Texto lindo
</p>

<script>
    let parrafo = document.getElementById("parrafo");
    console.log(parrafo.style.color);
    parrafo.style.color = "magenta";
</script>
```

Opciones:

a. Purple.
Y en la consola:
Texto lindo.

b. En la consola:
Purple.

c. Texto lindo.

d. Texto lindo.
Y en la consola:
Purple.

Pregunta 36

Selecciona cuál manejador de eventos se muestra en el siguiente fragmento de código:

<button onClick="alert('Hola!')">Saludar</button>
Opciones:

a. Propiedades JavaScript.
b. addEventListener().
c. Propiedades JavaScript.
d. Atributos HTML.

Pregunta 37

Selecciona el fragmento de código que crea un manejador de eventos.

Opciones:

a. <button onClick="alert('Hola!')">Saludar</button>

b. <button>Saludar</button>
<script>
const button = document.querySelector("button");
const saludar = () => alert("Hola!");
button.addEventListener("click", saludar);
</script>

c. <button>Saludar</button>
<script>
const button = document.querySelector("button");
button.onclick = function() {
alert("Hola!");
}
</script>

d. <body onload="window.alert('Bienvenido!');">

Pregunta 38

Selecciona el evento que se produce cuando se pulsa el botón izquierdo del *mouse* sobre un elemento.

Opciones:

a. Keydown.
b. Clic.
c. Load.
d. Unload.

Pregunta 39

Elige el método que también se conoce como evento de mutación.

Opciones:

a. Eventos DOM.
b. Eventos HTML.
c. Evento del teclado.
d. Evento del mouse.

Pregunta 40

Selecciona la opción que muestra el resultado del siguiente código:

<button>Saludar</button>
<style>
.red { background: red }
</style>

```
<script>
const button = document.querySelector("button");
const saludar = () => alert("hola!");
const toggle = () => button.classList.toggle("red");
button.addEventListener("click", saludar); // Agregar listener
button.addEventListener("click", toggle); // Toggle red CSS
button.removeEventListener("click", saludar); // Remueve listener
</script>
```

Opciones:

a. Se elimina el handler de saludar, de tal forma que al hacer clic ya no pase nada.
b. Se elimina el handler de toggle, de tal forma que al hacer clic ya no pase nada.
c. Se elimina el botón, de tal forma que al hacer clic ya no pase nada.
d. Se elimina el handler de saludar, de tal forma que al hacer clic ya no salude.