

Programación Orientada a Objetos

Conceptos:

Polimorfismo
Encapsulamiento
Herencia
Clase
Objeto
Constructor

Qué palabra venía en el prefijo de la clase que hace referencia en Herencia

R= [Extends](#)

¿Cual de estas oraciones definen herencia?

Clases que extienden pero no se pueden modificar
Alta cohesión y bajo acoplamiento

R= [La clase empleado tiene subclases Empleado Sindicalizado, Empleado...](#)

¿Cual es el modelo para un objeto?

Herencia
Instancia
constructor

R= [Clase](#)

Diferencia entre una clase y un objeto

Herencia
Instancia
constructor

R= [El objeto instancia una clase](#)

Que representa la clase en un objeto:

Herencia
Instancia
constructor

R= [Modelo](#)

Es el método que inicializa las propiedades de un objeto

instancia
clase

R= [El constructor](#)

Cómo se accede a los elementos privados de una clase

polimorfismo
constructor

R= [Encapsulamiento](#)

Los detalles internos de una clase pueden ser ocultados del exterior

Abstraccion
Herencia
Polimorfismo

R= [Encapsulamiento](#)

Dos clases base que heredan métodos y atributos a otra clase

herencia jerárquica

Polimorfismo

Herencia múltiple

R=Herencia Múltiple

Super clase con dos hijos

Composicion (Por si viene)

Herencia

Herencia jerárquica

Herencia Múltiple

R= Herencia

Información coherente y relacionada con la clase:

Bajo acoplamiento

Alto cohesión y bajo acoplamiento

R: Alta cohesión

La alta cohesión con que debe de ir:

Alto acoplamiento

R= Bajo acoplamiento

Qué es Polimorfismo:

encapsulamiento

polimorfismo

R= Parámetros sobre un método y cambia su comportamiento

Los modulos no dependen o conocen muy poco del funcionamiento interno de otros modelos.

R= Bajo acoplamiento

Person { nombre : string,

edad: number, Fec_Nac : date, nacionalidad: string

} ¿De acuerdo al principio de encapsulamiento, cual de estas propiedades se encapsula?

R: "Nombre"

¿Crea clases a partir de clases existentes, que extiende y refinan una clase?

R=Herencia

JAVASCRIPT

Que tipo de dato es null

undefined

null(en js existe null)

integer

R= Objeto

Menciona las variables de Javascript

R=var y let (por si viene y es la más correcta)

R=var let const

Menciona los tipos de datos en Javascript:

R=Number, string, boolean, null, undefined, Symbol, object

¿Como pasar un String JSON a un Objeto Javascript?

R=JSON.parse(stringJSON)

Como pasar un objeto javascript a String JSON?

R=JSON.stringify(objetoJS)

¿Cómo y cuándo utilizar las variables let y var?

R=Cuando queremos limitar el alcance de la variable

Afirmaciones, cuál NO es cierta en Javascript:

débilmente tipeado

sensible a mayúsculas y minúsculas

do

R = poner ; al final de una oración

let letras = ["lambda", "alfa", "gamma", "beta"];

letras.sort();

descendente

aleatorio

manda error

R = ordena el arreglo de forma ascendente

Resultado de este código document.querySelector('p.five')

Seleccionaba los 5 párrafos

Selecciona todos los párrafos con la clase "five"

manda error

R= Selecciona el primer elemento párrafo con una clase five

Elimina el elemento avión del siguiente arreglo:

let transportes = ["avion", "barco", "motocicletas", "automovil"];

transportes.shift();

transportes.pop();

transportes.unshift()

transporte.push()

R = transportes.shift();

Qué resultado es de esta función:

```
function myFuncion(){  
  const nombre = "pedro"  
  nombre = "juan"  
    return nombre  
}
```

R = Uncaught TypeError: Assignment to constant variable.

```
<script>  
  function f(){  
    let x= "4"+4+5;  
    let y= 4+4+"5";  
    console.log(x + ' '+ y );  
    console.log(type of);  
  }  
  f();  
</script>
```

R = Javascript hace type casting para convertir los tipos a strings, "445" "85"

Que resultado da: var square = number => number*number;

una función que recibe parámetros

una función que no recibe parámetros

una función vacía que trae dos parámetros

R= te daba un número al cuadrado

// Que retorna la siguiente function

```
function f(x, y=2, z=7){  
  return x + y + z;  
}
```

```
console.log(f(1) === 10);
```

https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Funciones/Parametros_por_defecto

false

error

undefined

R = true

```
function foo(){  
  return 5  
}
```

```
let myVar = foo;
```

retorna 5

myVar es igual a 5

R= la variable myVar hace referencia a la función

Como agregar un valor al final del arreglo

```
arr = ["val1","val2", "val3"];
```

```
arr[arr.length ] = "value"
```

```
function f (){  
var x; var y = x === null;  
  console.log(y)  
} f();
```

R = false

¿Que es lo que hace el siguiente código?

```
if(x===y){  
console.log("hola mundo");  
}
```

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Equality_comparisons_and_sameness

Hace la comparación del valor

Hace la comparación de tip e imprime hola mundo

Ninguna de las anteriores

R= Hace la comparación de tipo y valor e imprime "hola mundo"

```
(function f(f){  
    return typeof f();  
})(function(){return 1;})
```

<https://flaviocopes.com/javascript-iife/>
<https://developer.mozilla.org/es/docs/Glossary/IIFE>

¿Qué estados regresa tener una promesa?

rechazado

aceptado

en espera

Todas las anteriores

R= Todas

<https://developers.google.com/web/fundamentals/primers/promises>
https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Promise

Datos en Javascript:

- ☐ Number
- ☐ String
- ☐ Boolean
- ☐ undefined
- ☐ Symbol
- ☐ Object
- ☐ Null

Que hace esta función `var square = number => number*number; :`
Eleva un numero al cuadrado

Que hace este codigo

```
if(x===y){  
  console.log("hola mundo");  
}
```

Compara la variable x y que sean iguales en tipo y valor.

```
let letras = ["lambda", "alfa", "gamma", "beta"];  
letras.sort();  
descendente  
aleatorio  
manda error  
R = ordena el arreglo de forma ascendente
```

Que funcion reemplaza a Async().

Clase

R= Promesa

Prototype

Ninguna de las anteriores

Que hace esto `[1,2,3].map((n) => n*n)`

R= Un arreglo con cada uno de los números al cuadrado

Cuantos tipos de for se pueden usar en JavaScript

R= for, for in, for of, Array.forEach();

¿Qué oración define el funcionamiento de lo siguiente function

```
function(foo){  
  return foo && foo * 3 || null }
```

R= Si foo es diferente de 0 devuelve un true de lo contrario devuelve null

¿Como se define el metodo constructor en una clase?

R= Con la palabra reservada Constructor

```
class ClassName { constructor(){ name="anonymous"; age=0;  
} constructor(name,age){ name=name; age=age;}}
```

R= Sobrecarga

Que metodos están dentro de una promesa

R: then, catch, finally

En que momento se rechaza la promesa

R= Catch

Qué regresa la función bar ?

```
function foo () { console.log(a)}  
function bar() { var a= 3; foo(); }  
var a = 2; bar();
```

3

R= 2

error

undefined

Que retorna la siguiente función

```
function f () {  
    var x; var y = x === null;  
    console.log(y)  
}  
f();
```

R= false

Completa ["H","E","L","L","O"]

R= ["H", "E"].concat(["L", "L", "O"]);

Que arroja la siguiente funcion

```
(function f(f){  
    return typeof f();  
})  
(function(){  
    return 1;  
});
```

R= number

```
let mascotas = ["perro", "gato", "loro", "canario"];  
mascotas.filter(mascota=>mascota.length >=5 );  
console.log(mascotas);
```

Que se imprime en pantalla:

R= Perro, Canario

POLYMER

Keys no válidas en una declaración polymer:

readonly

notify

type

R= [_event](#)

<https://polymer-library.polymer-project.org/2.0/docs/devguide/properties>

Observers de qué forma son:

R= [sincrona](#)

<https://polymer-library.polymer-project.org/2.0/docs/devguide/observers#observers-are-synchronous>

Como se pone un listener a un campo de texto, para saber en que momento se pierde el focus:

on-lost-focus

on-change

on-leave

R= [on-blur](#)

https://www.w3schools.com/tags/ref_eventattributes.asp

Como se realiza el monitoreo de cambios en las propiedades de un elemento

molecule, observer

R = [observer](#), [databinding](#), [computed](#)(regresa valor)

Como se llama un observer de dos propiedades:

simple

doble

multiple

R = [Complejo](#)

<https://polymer-library.polymer-project.org/2.0/docs/devguide/observers#change-callbacks>

<https://polymer-library.polymer-project.org/2.0/docs/devguide/observers#complex-observers>

Propiedades virtuales cuyo valor se determina basado en otras propiedades

observer

behaviours

databinding

R = [Computed](#)

Cuales son los tipos de datos válidos en polymer

R = [number](#), [string](#), [array](#), [object](#), [date](#)

Boolean, Date, Number, String, Array and Object.

<https://polymer-library.polymer-project.org/2.0/docs/devguide/properties>

Como eliminar el valor de un arreglo en Polymer

R = [this.splice\('persons', 2, 1\)](#)

<https://polymer-library.polymer-project.org/2.0/docs/devguide/model-data#array-mutation>

El tipo de dato que representa un nulo:

R = Object

**De forma declarativa como se hace binding en un atributo de un componente
firstName**

<mi-componente first.Name="{{ }}">

<mi-componente first_name="{{ }}">

<mi-componente firstname="{{ }}">

R= <mi-componente first-name="{{ }}" >

Como hacer un binding dentro de un elemento input

{{input::firstName}}

{{first-name value= input}}

R= {{firstName::input}}

Que tipo de bindeo es este: {{ }}

one-way

one-one

face-to-face

R = two-way

Cómo declarar un dom-if, dom-repeat

<template is="dom-repeat" items="{{propPadre}}" as="algo">

</template>

<template is="dom-if" if="{{boolean}}">

</template>

Que propiedad modifica su valor dependiendo de otros atributos

R= computed

Sintaxis correcta de un component

<Mi-Componente></Mi-Componente>

R=<mi-componente> </mi-componente>

<https://html.spec.whatwg.org/multipage/custom-elements.html>

Properties mas utilizados

value, reflectToValidate

notify

R= type, reflectToAtributte, value

Que es un custom event y como se dispara

this.fire → version 1 de polymer

R= Es un evento personalizado que se ocupara para la comunicación entre elementos del DOM,se dispara con el elemento.dispatchEvent

De un arreglo de personas modificar el 5to elemento

R= this.set('person.4.lastName', 'nuevo valor')

this.set('person.[4].lastName', 'nuevo valor')

this.set('person.(4).lastName', 'nuevo valor')

this.person[4].lastName = 'nuevo valor';

Que tipo de binding es este [[]]

R = one way

Dentro de un componente como cambiar el color de texto

```
::p, ::  
::body  
::html  
R = :host {  
    var(--var-host-color, blue)  
}
```

Permite uso de la función filter como una de sus propiedades

dom-if

dom-filter

dom-module

R = Dom-repeat

<https://polymer-library.polymer-project.org/2.0/docs/devguide/templates#filtering-and-sorting-lists>

Ciclos de vida de un componente de polymer:

R= ConnectCallback, DisconnectCallback, attributeChangedCallback, constructor, ready

<https://polymer-library.polymer-project.org/2.0/docs/devguide/custom-elements#element-lifecycle>

Que evento se dispara la siguiente línea de código?

```
name: {  
    type: String,  
    notify: true  
}  
this.set('name', '');
```

Posibles respuestas:

R=name-changed

name-edit

name-notify

ningún evento

Cual de estos estados, regresa una promesa

received

rejected

declined

R = rejected

<https://developers.google.com/web/fundamentals/primers/promises>

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Promise

Notifica a polymer del cambio hecho solamente en una propiedad

this.push

this.notify

R=this.notifyPath

this.set

Metodo de polymer para asegurar que solamente se haga un cambio en una propiedad de polymer

setProperties

setAttributes

getValues

R= En polymer no se puede realizar este tipo de operaciones

Cuales son los Gesture Events

R= down, up, tap, track

Información <https://polymer-library.polymer-project.org/2.0/api/namespaces/Polymer.Gestures>

function foo(){ return 5 } let myVar = foo;

R= la variable myVar hace referencia a la función

Cual de los sig. estados es una promesa

R= Rechazado, aceptado y en espera (Todas las anteriores)

¿Que es lo que hace el siguiente código?

if(x===y){ console.log("hola mundo"); }

R= Compara que "x" y "y" sean exactamente iguales tanto en tipo y valor

COMO PASAR UN OBJETO A JAVASCRIPT?

R= JSON.parse()

Como compartias estilos entre diferentes

<link rel="import" href="src/static/style.html">

¿Qué métodos ofrece Polymer para mutar un arreglo?

R= push, pop, unshift, shift, splice

Pagina de referencia de la información

<https://polymer-library.polymer-project.org/1.0/docs/devguide/model-data>

Cuál es el funcionamiento de this.\$.id

R= Para seleccionar un nodo específico del Shadow DOM

Como importas un componente en polymer 2

R= <link rel="import" href="bower_components/input-paper/input-paper.html">

Como aplicas los siguientes estilos

**.clase { _____
 _____ }**

#apply

Var (____clase)

R= @apply(____clase)

apply (____clase)

Agregar un valor al final de un array

R= arr[arr.length]= value

arr[arr.length +1]= value

arr[arr.length - 1] = value

Como asignar un arreglo vacío a una propiedad de polymer

value:“[]”

Value:[]

Value:newArray()

R= value:()=>[]

Ejemplo

tasks: { type: Array,

value: ()=> [] }

Se requiere mostrar un resultado de color azul cuando se cumple una condición y cuando se cumple otra se resalta de color rojo. Que utilizarías

R= Dom-If para cada uno

Dom-if-else

Dom-if y un Dom-else

Todas las anteriores

```
<template is="dom-repeat" items="{{persons}}" as="person">
```

```
<template is="dom-if" if=[[!_graterThan(person.age,18)]] > <p>[[person.name]]
```

```
[[person.lastName]] => [[person.age]]</p>
```

```
</template> </template>
```

```
_graterThan(age, value){ return age >value ;
```

```
}
```

R= Todos los menores de 18 años