**Angular Js Conceptos**

1. ¿Qué componentes se pueden inyectar como dependencias en angularJS?
2. Contant
3. Application Module
4. Factory
5. Value
6. ¿Qué indica la bandera $dirty en angularJS?
7. Esta indica que un valor no ha cambiado.
8. Esta indica que un valor ha sido cambiado.
9. Esta indica a un formulario que tiene información inválida
10. Ninguna de la anteriores.
11. ¿La directiva ng-init en angularJS es usada para qué?
12. Determina si dos valores o dos objetos son equivalentes
13. Serializa entradas en una cadena JSON-formatted
14. Permiten evaluar una expresión en el Scope actual.
15. Envolver un elemento del DOM o una cadena HTML en un elemento jQuery
16. ¿Para qué es usado angular.bootstrap?
17. Determinar si una referencia es un elemento del DOM
18. Es usado para iniciar manualmente una aplicación en angular
19. Deserializa una cadena de tipo JSON
20. Todas las anteriores
21. ¿Cuál es la función del binding y doble binding en angularJS?
22. Enlaza la información que hay en el Scope y la muestra en html
23. Es un objeto que se refiere al modelo de aplicación
24. Configurar el estado inicial del objeto $scope
25. Formatear el valor de una expresión para mostrar al usuario
26. ¿Qué función cumple el ngRepeat?
27. Realizar el enlace de un objeto a un elemento html
28. Implementar una repetición (un bucle)
29. Inicia los datos de una aplicación
30. Dispara el evento clic de un elemento html
31. ¿Qué es una directiva?
32. Es la encargada de realizar las peticiones al servidor
33. Es cualquier expresión establecida dentro las etiquetas HTML
34. Son comandos los cuales se pueden incorporar como elementos html o como atributos
35. Son aplicaciones embebidas dentro de otras aplicaciones de angular.
36. ¿Qué función cumple la directiva ngClass?
37. Indica que es el elemento raíz de la aplicación AngularJS
38. Encargada de definir las clases CSS en las etiquetas html
39. Indicar expresiones a realizar cuando se produce un evento de cambio en el elemento
40. Enlaza el valor de los controles HTML
41. ¿Qué son los módulos en angularJS?
42. Son  plantillas con información del modelo y del controlador para representar la vista dinámica que un usuario ve en el navegador
43. Permite adjuntar comportamiento a cualquier elemento o atributo HTML e incluso crear nuevos elementos HTML
44. Son contenedores donde se sitúa el código de los controladores y directivas
45. Es un tipo especial de directiva que utiliza una configuración más sencilla que es adecuada para una estructura de aplicación basada en componentes
46. ¿Cuál es la función de los controladores en angularJS?
47. Proporcionar vinculación de datos a nodos de texto y valores de atributo
48. Ayuda a su aplicación a convertirse en una aplicación de una sola página
49. Monitorea el estado del formulario y los campos de entrada (input, textarea, select)
50. Permiten mediante programación implementar la lógica de la presentación en AngularJS

Respuestas:

1. b
2. b
3. c
4. b
5. a
6. b
7. c
8. b
9. c
10. d

**Test Javascript/Ecma Script**

1. ¿ Cuál es la salida de del siguiente código? (1 pto)

***let*** *x* = 42;  
***if*** (***true***) {  
 ***let*** *x* = 1337;  
}  
**console**.log(*x*);

1. ¿ Cuál es la salida de del siguiente código? (1 pto)

***var*** *x* = **`foo** ${*y*}**`**,  
 *y* = **`bar** ${*x*}**`**;  
  
**console**.log(*x*);

1. ¿ Cuál es la salida de del siguiente código? (1 pto)

***const*** *CLAVE* = **'conejo\_blanco'**;  
***if*** (***true***) {  
 ***const*** *CLAVE* = **'conejo\_angora'**;  
}  
**console**.log(*CLAVE*);

1. ¿ Cuál es la salida de del siguiente código? (1 pto)

***let*** *x* = 42;  
***if*** (***true***) {  
 **console**.log(*x*);  
 ***let*** *x* = 1337;  
}

1. ¿ Cuál es la salida de del siguiente código? (1 pto)

***var*** *x* = **`foo** ${*y*}**`**,  
 *y* = **`bar** ${*x*}**`**;  
  
**console**.log(*y*);

1. ¿Cuál es el equivalente al siguiente código?(1 pto)

***var*** *arr* = [12,5,42,55,10];  
  
***function*** hacer\_algo(arr){  
 ***for***(***let*** i=0; i<arr.**length**; i++) {  
 ***if***(arr[i]>32) {  
 ***return*** arr[i];  
 }  
 }  
}

* 1. *arr*.find(n => n > 32);
  2. *arr*.findIndex(n => n > 32);
  3. *arr*.**search**(n => n > 32);

1. En ES2015 existe una mejor manera de declarar el siguiente objeto. ¿Cuál? (1 pto)

***let*** *options* = {  
 **protocol**: protocol,  
 **url**: url,  
 **method**: method,  
 **callback**: callback  
};

1. ¿Cuáles son los resultados del siguiente código? (2 ptos)

***function*** sumar(a=10, b=5) {  
 ***return*** a+b;  
}  
  
*// i*sumar(11, 2);  
  
*// ii*sumar(3);  
  
*// iii*sumar(*undefined*, 20);  
  
*// iv*sumar();

* 1. i:15 ii:13 iii:30 iv:15
  2. i:13 ii:8 iii:30 iv:15
  3. i:13 ii:13 iii:30 iv:undefined

1. ¿Cuál es el código equivalente al siguiente? (1 pto)

***var*** *nombre*=**"Charlie"**, *trabajo*=**"agente secreto"**, *de*=**"Inglaterra"**;  
***var*** *persona* = {  
 **'nombre'**: *nombre*,  
 **'trabajo'**: *trabajo*,  
 **'de'**: *de*};

* 1. ***let*** *nombre*=**"Charlie"**, *trabajo*=**"agente secreto"**, *de*=**"Inglaterra"**;  
     ***let*** *persona* = ***new*** object(*nombre*, *trabajo*, *de*);
  2. ***let*** *nombre*=**"Charlie"**, *trabajo*=**"agente secreto"**, *de*=**"Inglaterra"**;  
     ***let*** *persona* = { *nombre*, **trabajo**: *trabajo*, *de*};

1. ¿ Cuál es la salida de del siguiente código? (1 pto)

***let*** *a* = 12,  
 *b* = 3;  
[*a*, *b*] = [*b*, *a*];  
  
**console**(*a*,*b*);

1. ¿Cuál es el valor de la variable resultado ? (1 ptos)

***function*** misterio(...params) {  
 ***return*** params;  
}  
  
***let*** *resultado* = misterio(1, 23, 4);

1. ***“1 23 4”***
2. ***[1, 23, 4]***
3. ***Undefined***
4. ¿Cuál es el resultado del siguiente código? (1 pto)

((...x, xs) =>x)(1,2,3)

1. ***1***
2. ***3***
3. ***[1,2,3]***
4. ***Error***
5. ¿Cuál es el resultado del siguiente código? (1 pto)

(***function***(x, f = () => x) {  
 ***var*** x;  
 ***var*** y = x;  
 x = 2;  
 ***return*** [x, y, f()];  
})(1)

1. ***[2,1,1]***
2. ***[2, undefined, 1]***
3. ***[2,1,2]***
4. ***[2, undefined, 2]***

Respuestas:

1. 42
2. foo undefined
3. conejo\_blanco
4. ERROR
5. bar foo undefined
6. a
7. let options = {

protocol,

url,

method,

callback

};

1. b
2. b
3. 3 12
4. b
5. (d): El parámetro rest debe ser el último en una función
6. (a): porque los parámetros crean un scope extra cuando tienen valores por defecto

**Mongo DB Conceptos**

1. Mongo db es un almacen de datos \_\_\_\_\_\_\_\_ para documentos \_\_\_\_\_\_
   1. Relacional, xml
   2. Relacional, JSON
   3. No relacional, TXT
   4. No relacional, JSON

|  |  |  |  |
| --- | --- | --- | --- |
| 2 | V | F | Mongo db es schemaless (sin esquema) |
| 3 | V | F | Mongo db soporta JOINS |
| 4 | V | F | Mongo db soporta transactions |
| 5 | V | F | MongoDB es más escalable que un sistema RDBMS (Sistema Gestor de Bases de Datos Relacionales). |

Mongo Db Shell

1. Comando que permite acceder al Mongo Shell
   * + - 1. Mongod
         2. Shell mongo
         3. Mongo
         4. Mongodb
2. Comando que permite mostrar las base de datos disponibles en una instancia

Show databases

Show db

List databases

Show dbs

1. ¿Qué comando se utiliza para cambiar de base de datos?
   1. Change db
   2. Use
   3. Change
   4. Define
2. El comando "show tables" muestra las colecciones que existen en la base de datos activa
3. Verdadero
4. Falso
5. El comando db.students.find() muestra el primer documento encontrado en la colección students.
6. Verdadero
7. Falso
8. Para insertar un documento en la colección "students", utilizaría el siguiente comando
9. db.students.insert({"nombre":"Carlos","edad":18})
10. db.students.add({"nombre":"Carlos","edad":18})
11. database.students.insert({"nombre":"Carlos","edad":18})
12. db.students.addrow({"nombre":"Carlos","edad":18})
13. El comando db.dropDatatabase() nos permite eliminar la base de datos activa.
14. Verdadero
15. Falso

Respuestas:

1. D
2. V
3. V
4. F
5. V
6. C
7. D
8. B
9. B
10. B
11. A
12. A