**CARGO: DESARROLLADOR FULL STACK**

**Nombres: Fredy Alejandro Camargo Sanchez**

**Fecha: 2023-11-28**

**Prueba técnica DESARROLLADOR FULL STACK**

**Parte teórica**

1. Defina la programación orientada a objetos y sus pilares.

La programación orientada a objetos tiene como principios la modularidad, abstracción, encapsulamiento, ocultación y accesibilidad, los objetos permiten el poder reutilizar código en vez de crear copias del mismo objeto o proceso.

1. ¿Qué es la mutabilidad e inmutabilidad en programación?

La mutabilidad es una característica que permite que una clase u objeto pueda cambiar acorde a las necesidades.

1. ¿Cuáles son las buenas prácticas de programación y como pueden ayudar al desarrollo a futuro?

Son como los estándares ayudan a tener un lenguaje en común y una lógica en común esto con el fin de entregar proyectos de alta calidad y proyectos de alto nivel.

1. ¿Cuál es la diferencia entre una clase y un objeto en la programación orientada a objetos y cómo se pueden utilizar para modelar el mundo real?

Ejemplo de clase esta nos sirve para hacer los procesos para este caso la clase representa un cálculo de suma de la calculadora:

Public int Calculadora.Suma(int x, int y)

{

Int resultado = Sumar(x, y);r

Return resultado;

}

Un objeto es un espacio o ubicación en memoria en donde se encuentran puede ser una variable o contante, o una reunión de variables que conforman un objeto compuesto, es de aclarar que aquí no se realizar cálculos por buenas prácticas aunque es posible realizarlos aquí.

Public Objeto.Parametros

{

Public int x {get; set;}

Public int y {get; set;}

}

1. ¿Qué es la encapsulación en la programación orientada a objetos?

La encapsulación es una característica de las clases y objetos de estar encapsulados o agrupados en donde para acceder a estos se hace mediante un llamado al punto de partida del objeto en memoria, esto es útil cuando en ves de tener varios cálculos de una calculadora desagrupados se tienen encapsulados dentro de la calculadora pudiendo proteger algunos procesos de que sean accesibles desde afuera y protegiendo su modificación.

1. ¿Qué es la arquitectura MVC? Describa brevemente cada uno de sus componentes.

El significado del acrónimo MVC es Modelo Vista y Controlador, Son capas o mejor un modelo de arquitectura en donde la Vista representa la capa de interacción del usuario, el Controlador representa la capa de negocios en donde se realizan los cálculos, y la capa de datos representa la abstracción de la base de datos las tablas de la base de datos en forma de clases u objetos.

**PHP O C#**

1. Si tenemos un string str = '1,2,3,4,5,6,7’; ¿cómo podemos obtener la suma de los enteros que contiene?

//Sumatoria de estos valores

string str = '1,2,3,4,5,6,7’;

//Definimos una lista en donde separaremos el texto '1,2,3,4,5,6,7’

//Resultado elementos separados “1”,”2”,”3”, etc. en una lista.

List<string> listNumerosString = str.Split([,]).Tolist();

//Opción 1: Recorremos la lista y hacemos la sumatoria.

Int sumatoriaNumeros = 0;

Foreach(var itemNumerosString in listNumerosString)

{

sumatoriaNumeros = SumatoriaNumeros + itemNumerosString;

//Resultado para T = 1 SumatoriaNumeros = 0 + 1

//Resultado para T = 2 SumatoriaNumeros = 1 + 2

//Resultado para T = 2 SumatoriaNumeros = 3 + 4 etc

}

//Opcion 2: Simple retorna la sumatoria de los números usando expresiones Lambda

Int sumatoriaNumeros = listNumerosString.Sum(p => Convert.ToInt(p));

1. Dado un string s y un número entero n, encuentra todas las subcadenas de s de longitud n que contengan al menos una letra mayúscula y al menos una letra minúscula.

Por ejemplo, para el string "HelloWorld" y n=3, las subcadenas válidas serían "Hel", "ell", "llo", "loW", y "oWo". La subcadena "Wor" no es válida ya que no contiene una letra minúscula.

//Cadena de texto de entrada

//"Hel loW orld”

//”012 345 678

string texto = "HelloWorld";

int n = 3;

//Paso 1: se divide el texto en partes

List<String> ListTexto = new List<String>();

Int rangoInicial = 0;

Int rangoFinal = (n-1);

//Aquí vamos a agregar las partes del texto

For (i = 1; i > 0;i++)

{

string texto = texto.Substring(rangoInicial, rangoFinal);

//Si ya no retorna texto salimos

if(String.IsnullorEmpty(texto))

{

i = -1;

}

//Agregmos el texto cortado a la lista

ListTexto.Add(texto);

rangoInicial = rangoInicial + n;

rangoFinal = rangoFinal + n;

//"Hel loW orl d”

//”012 345 678 9

//Para T = 1; rangoInicial 0; rangoFinal = 2;

//Para T = 2; rangoInicial 3; rangoFinal = 5;

//Para T = 2; rangoInicial 6; rangoFinal = 8;

}

Paso 2: Procesar el texto

//Utilizare una búsqueda en el texto por regex

//Definir la opción de búsqueda significa de la A a la Z en mayusculas

System.Text.Regex regexMayusculas = new Regex("[A-Z]");

//Definir la opción de búsqueda significa de la A a la Z en minusculas

System.Text.Regex regexMinusculas = new Regex("[a-z]");

List<string> listResultado = new List<String>();

//Recorrer la lista ListTexto = { "Hel”, “loW”, “orl”, “d” };

Foreach (var itemListTexto in ListTexto)

{

//Condición para saber si contiene almenos 1 mayusculas

MatchCollection matchesMayusculas = regexMayusculas.Matches(texto);

If(matchesMayusculas.Count() > 0)

{

listResultado.Add(itemListTexto);

}

MatchCollection matchesMayusculas = regexMinusculas.Matches(texto);

//Condición para saber si contiene almenos 1 minusculas

If(regexMinusculas.Count() > 0)

{

listResultado.Add(itemListTexto);

}

}

//Resultados: listResultado = { "Hel”, “loW”}

1. ¿Cómo consumirías un servicio de tipo REST en PHP o C#?

Rest en PHP:

Rest en C#:

Var url = “http://LocalHost/Servicios/ServicioName”

//Consumido como Rest

HttpClient client = new HttpClient();

Var Return = Client.PostAsync(url);

//consumido como un WCF.

Servicios Servicios = new Servicios ();

Servicios.ServicioName(Parametros);

//Servicio asincrono

Public Class Servicios

{

[HttpPost]

Public Async<Task> string ServicioName(string Parametros)

{

Retun “Hola Mundo”;

}

}

1. ¿Cuáles son los formatos más comunes en que se procesan las respuestas de un RESTFUL?

Los formatos más comunes son las respuestas http ejemplo.

Cuando el servidor retorna un error 500 retorno en realidad una variable HttpResponse con Http.status.Fail

Dentro de esta variable se adjunta el valor de la respuesta como resultado de las operaciones del servicio HttpResponse soporta objetos dinámicos

1. Que función convierte una cadena en mayúsculas en C# o PHP.

En ambos casos sea PHP o C# la funci9on que retorna una cadena de caracteres en formato de mayúsculas es ToUpperCase(), o en minúsculas el .ToLowerCase().

**Problema picas y fijas**

El siguiente problema se puede resolver en un proyecto de consola, formulario web o interfaz básicos, para esto se debe entregar en un repositorio Github, bit bucket o cualquier otro de su preferencia las fuentes o código para poder ejecutarlo y analizarlo, el problema es el siguiente:

El juego de Picas y Fijas es un juego de adivinanza en el que el sistema genera un número secreto de cuatro dígitos. El objetivo del jugador es adivinar este número secreto. En cada intento, el jugador proporciona un número de cuatro dígitos y el sistema le indica cuántas "picas" y cuántas "fijas" tiene el número ingresado.

Una "pica" indica que un dígito del número ingresado existe en el número secreto, pero está en una posición incorrecta. Una "fija" indica que un dígito del número ingresado existe en el número secreto y está en la posición correcta.

Por ejemplo, si el número secreto es 1234 y el jugador ingresa 2031, el sistema debería indicar que hay 2 picas (los dígitos 2 y 0 están en el número secreto, pero en posiciones incorrectas) y 1 fija (el dígito 1 está en la posición correcta).

El juego continúa hasta que el jugador adivina el número secreto (es decir, hasta que el número ingresado tenga 4 "fijas")

(Se puede entregar C# o PHP)

1. Para la tabla de "**Alumnos**" generar los query´s que se usarían para hacer un CRUD

Ver: Select \* From **Alumnos**

Filtrar: Select \* From **Alumnos** as A Where A.Campo1 = ‘Valor1’

Insertar: Insert into **Alumnos** (Campo1, Campo2)Values (‘Valor1’, Valor2)

Actualizar: Update **Alumnos** Set Campo1 = ‘Valor1’ where Campo1 = ‘Condición1’

Eliminar: Delete From **Alumnos** where Campo1 = ‘Condición1’

Nota esto mismo se puede hacer en Linq y expresiones Lambda.

1. ¿Cuál es la función principal de una clave primaria en una tabla?

Por buenas prácticas y teniendo en cuenta que en un modelo de base de datos relacional existen entidades, atributos, relaciones y consultas, las relaciones entre tablas se identifican por esta llave o clave primaria o principal, además es el índice por defecto y que debe tener una posición única por defecto excepto en las llaves compuestas; esta posición, . identifica de forma única o singular a la fila.

1. ¿Qué es el DML y el DDL?

DML: Se usa generalmente para ver, crear, actualizar, o eliminar en cuanto a las entidades o tablas de la base de datos.

DDL: Se usa generalmente para crea consultas de ver, insertar, actualizar, eliminar enfocadas a los valores de la base de datos.

**Notas aclaratorias:**

El objetivo de esta prueba es medir el nivel de razonamiento y conocimiento base del candidato, por ende, el uso de cualquier inteligencia artificial (u otras fuentes en internet) durante la solución de estos puntos puede ser considerada como fraude o plagio, el interés particular no esta en medir de manera exacta el conocimiento, sino que por el contrario busca medir los puntos fuertes y débiles del candidato.

**¡Buena suerte!**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ESPACIO EXCLUSIVO PARA EL CALIFICADOR** | | | | |
| Calificación (0 –5):  5 la máxima calificación | **3.2** | Fecha de calificación | | **14/11/2023** |
| Nombre de Calificador: |  | | Firma |  |