

Test de Selección Múltiple para Desarrolladores de Software

Instrucciones: Selecciona la opción correcta en cada una de las siguientes preguntas. Cada pregunta tiene una sola respuesta correcta.

1. ¿Qué es un algoritmo de ordenamiento estable?

A) Un algoritmo que siempre tarda el mismo tiempo en ejecutarse, independientemente del orden de entrada.

B) Un algoritmo que conserva el orden de registros con valores iguales.

C) Un algoritmo que utiliza una cantidad constante de memoria adicional.

D) Un algoritmo que puede adaptarse a diferentes tipos de arquitecturas de hardware.

2. En el patrón de diseño MVC, ¿qué componente es responsable de gestionar la lógica de negocio?

A) Modelo

B) Vista

C) Controlador

D) Cliente

3. ¿Qué es un deadlock en sistemas operativos?

A) Un proceso que termina sin liberar los recursos.

B) Una situación donde dos o más procesos están esperando indefinidamente por recursos que están siendo ocupados por el otro.

C) Un error que ocurre cuando un proceso trata de ejecutar una operación ilegal.

D) Una técnica para mejorar el rendimiento de los procesos.

4. ¿Qué afirmación sobre las bases de datos NoSQL es correcta?

A) No soportan transacciones

ACID.

B) Siempre requieren esquemas fijos y estructurados.

C) No permiten consultas sobre grandes volúmenes de datos.

D) Están diseñadas para escalar horizontalmente.

5. ¿Qué lenguaje de programación introduce el concepto de Promesas para manejar la asincronía?

A) Java

B) Python

C) JavaScript

D) C#

6. ¿Qué estructura de datos es óptima para implementar una caché que necesita frecuentes eliminaciones del elemento menos recientemente usado?

A) Array

B) Pila

C) Cola de prioridades

D) Mapa ordenado

7. ¿Qué es una función pura en programación funcional?

A) Una función que utiliza variables globales para mantener su estado.

B) Una función que no produce efectos secundarios y el retorno solo depende de sus argumentos.

C) Una función que siempre retorna null.

D) Una función que cambia el estado de sus argumentos.

8. ¿Cuál es el propósito principal de los contenedores Docker?

A) Incrementar la seguridad en las aplicaciones.

B) Proporcionar una máquina virtual completa para cada aplicación.

C) Virtualizar el sistema operativo para ejecutar aplicaciones en ambientes aislados.

D) Ejecutar exclusivamente bases de datos NoSQL.

Preguntas de profundización.

Net Core

1. ¿Qué es .NET Core?

R: Es un lenguaje de Microsoft el cual está diseñado para ser multiplataforma, y que las aplicaciones funcionen en todos los sistemas operativos .

2. **¿Cuál es la diferencia entre .NET Core y .NET Framework?**

R: La diferencia principal radica en que .NET Core es multiplataforma funcionando para diferentes SO, mientras que el .Net Framework es cerrado a windows.

3. **¿Qué son los Middlewares en .NET Core?**

R: Es la parte que se encarga de las solicitudes web de la aplicación

4. **¿Cómo se maneja la inyección de dependencias en .NET Core?**

R:

5. **¿Cómo se manejan las excepciones en .NET Core?**

R: Las excepciones se utilizan con el "Try - Catch", poniendo el código que podría generar una excepción en el "Try" y como se manejara esta excepción en el "Catch"

6. **¿Cómo se realiza la autenticación y autorización en .NET Core?**

R:

7. **¿Cómo se puede implementar una API RESTful en .NET Core?**

R:

ReactJS

1. **¿Qué es ReactJS y por qué se usa?**

R: Es un framework de JavaScript fabricado por facebook el cual se centra en tener una interfaz agradable y dinámica para los usuarios

2. **¿Puedes explicar qué son los componentes en React?**

R: Los componentes funcionan como contenedores, los cuales segmentan el código de diseño y la lógica de este diseño, con el fin de editar si es necesario sin afectar el código completo.

3. **¿Cómo se maneja el estado en un componente de React?**

R:

4. **¿Qué son las props en React?**

R:

5. **¿Cómo se manejan los eventos en React?**

R:

6. **¿Qué es un componente controlado en React?**

R:

7. **¿Cómo se implementan los formularios en React?**

R:

8. **¿Qué es el ciclo de vida de un componente en React?**

R: El ciclo de vida de un componente en react hace referencia a las diferentes fases que pasará este componente, desde que se crea hasta que se elimina.

9. **¿Cómo se realiza la representación condicional en React?**

R:

SQL Server

1. **¿Qué es SQL Server y para qué se utiliza?**

R: SQL Server es servidor de gestión de base de datos relacionales creado por microsoft, se utiliza para almacenar y gestionar bases de datos pequeñas o muy grandes

2. **¿Cómo se crea una tabla en SQL Server?**

R: CREATE TABLE "Nombre de la tabla"(
"Campos de la tabla"
);

3. **¿Cómo se inserta, actualiza y elimina datos en SQL Server?**

R:

Insertar:

INSERT INTO "Nombre de la tabla"("Campos de la tabla")

VALUES ("valores a ingresar");

Actualizar:

UPDATE "Nombre de la tabla"

SET "Campo de la tabla a actualizar" = "Valor nuevo"

WHERE "Campo de referencia de la tabla" = "Valor del campo"

Eliminar:

DELETE FROM "Nombre de la tabla"

WHERE "Campo de referencia de la tabla" = "Valor del campo"

4. **¿Cómo se realiza una consulta básica de selección en SQL Server?**

R: SELECT *FROM "nombre de la tabla";

5. **¿Cómo se realiza una consulta de unión en SQL Server?**

R: (SELECT * FROM "Nombre de la tabla 1")

UNION ALL

(SELECT * From "Nombre de la tabla 2")

6. **¿Cómo se realiza una consulta de agrupación en SQL Server?**

R: SELECT "Campo al que se va a agrupar", SUM ("Campo a agregar) AS "Alias
del campo a agregar

FROM "Nombre Tabla"

GROUP BY "Campo que se va a agrupar";

7. **¿Qué son las claves primarias y foráneas en SQL Server?**

R: Las claves primarias son los campos de los registros los cuales son únicos y que no se repiten, normalmente son los ID, los cuales son números consecutivos.

Las claves foráneas son datos o campos que comparten dos tablas, y este campo las relaciona, quiere decir que este campo debe existir en ambas tablas, de manera que una de las tablas toma este campo de la otra tabla y lo utiliza.

8. **¿Cómo se utiliza la cláusula WHERE en SQL Server?**

R: SELECT *FROM "Nombre de la tabla"

WHERE "Campo de referencia 1" = "Valor del campo"

AND "Campo de referencia 2" (<, >, =, <=, >=, <>) "Valor de condicion"

9. **¿Cómo se utiliza la cláusula ORDER BY en SQL Server?**

R: SELECT "Nombre de campo 1", "Nombre de campo 2"
FROM "Nombre de la tabla"
ORDER BY "Nombre campo"

Azure

1. **¿Qué es Azure y por qué se usa?**

R: Es una plataforma de servicios en la nube creada por microsoft la cual ofrece muchas funciones de desarrollo y seguridad.

2. **¿Puedes enumerar algunos de los servicios que ofrece Azure?**

R: -Azure Database
-Azure DevOps
-Azure machine learning
-Azure SQL database

3. **¿Cómo se implementa una aplicación web en Azure?**

R:

4. **¿Qué es una cuenta de almacenamiento de Azure?**

R: Como su nombre lo dice es un servicio que permite almacenar archivos en la nube, como imágenes, archivos y en general datos, nos permite tener accesibilidad amplia a ellos y con buena seguridad

5. **¿Cómo se maneja la seguridad en Azure?**

R: Permite obtener varios planes de azure los cuales ayudan al cifrado de datos, a la autenticación por roles y la gestión de contraseñas dinámicas para las cuentas lo cual ayuda a tener una mayor seguridad

6. **¿Cómo se realiza la escalabilidad de una aplicación en Azure?**

R:

Coding exercises

Proyecto para Desarrollador Junior: "Tienda en Línea Básica"

Funcionalidad:

- Permitir a los usuarios navegar a través de una lista de productos.
- Permitir a los usuarios agregar productos a un carrito de compras y realizar el proceso de pago ficticio.
- Debe tener una interfaz de usuario simple pero efectiva.

Entregables:

- Código fuente del proyecto.
- Documentación básica del código.
- Pruebas unitarias básicas.

R:

Link drive

https://drive.google.com/drive/folders/1ogeFZ8z-c6ZBq_YEvuP9ypp7w9sJD9V9?usp=sharing

Link de GitHub:

<https://github.com/SantiagoMR02/Prueba-TELEPERFORMANCE>

Nota: Se utilizó el live server de VS code para la visualización correcta de la página.