

Test: Modelo Entidad-Relación en Bases de Datos Relacionales

Instrucciones:

Selecciona la opción correcta en cada pregunta.

1. ¿Qué representa una entidad en el modelo Entidad-Relación?

- a) Una tabla en la base de datos
- b) Un atributo de una tabla
- c) Un valor dentro de una celda
- d) Un tipo de dato específico

2. ¿Cómo se representa gráficamente una entidad en un diagrama Entidad-Relación?

- a) Un rombo
- b) Un rectángulo
- c) Un óvalo
- d) Un hexágono

3. ¿Qué tipo de entidad no depende de otra para existir?

- a) Entidad dependiente
- b) Entidad fuerte
- c) Entidad débil
- d) Entidad secundaria

4. ¿Qué se utiliza para representar una relación entre entidades en un diagrama Entidad-Relación?

- a) Un círculo
- b) Un rectángulo
- c) Un rombo
- d) Un triángulo

5. ¿Cómo se representa gráficamente un atributo en el modelo Entidad-Relación?

- a) Un rectángulo
- b) Un óvalo
- c) Un hexágono
- d) Un triángulo

6. ¿Qué representa una clave primaria en una entidad?

- a) Un atributo opcional
- b) Un identificador único
- c) Una relación con otra entidad
- d) Un tipo de dato

7. ¿Qué representa una relación de cardinalidad 1:N entre dos entidades?

- a) Un elemento de la primera entidad se relaciona con varios de la segunda
- b) Cada elemento de ambas entidades solo se relaciona con uno de la otra
- c) Varios elementos de la primera entidad se relacionan con uno de la segunda
- d) Ninguna de las anteriores

8. ¿Qué caracteriza a una entidad débil en un modelo Entidad-Relación?

- a) No tiene atributos propios
- b) Depende de una entidad fuerte y necesita una clave foránea
- c) Es un atributo compuesto
- d) Solo se usa en bases de datos NoSQL

9. ¿Cómo se representa una clave foránea en un modelo Entidad-Relación?

- a) Con un óvalo
- b) Con un rectángulo
- c) Con un subrayado doble
- d) Con una línea que une entidades

10. ¿Cuál es el propósito principal del modelo Entidad-Relación?

- a) Implementar directamente una base de datos
- b) Representar gráficamente los datos y sus relaciones antes de implementarlos
- c) Analizar el rendimiento de consultas SQL
- d) Crear interfaces gráficas de usuario

Test: Modelo Entidad-Relación Extendido (EER) en Bases de Datos Relacionales

Instrucciones:

Selecciona la opción correcta en cada pregunta.

1. ¿Qué concepto adicional introduce el Modelo Entidad-Relación Extendido (EER) respecto al MER tradicional?

- a) Relaciones binarias
 - b) Cardinalidades
 - c) Generalización y especialización
 - d) Claves primarias
-

2. ¿Qué representa la especialización en el modelo EER?

- a) La agrupación de varias entidades en una sola
 - b) La división de una entidad en subconjuntos más específicos
 - c) La eliminación de atributos en una entidad
 - d) La creación de una clave primaria compuesta
-

3. ¿Cómo se representa gráficamente la generalización en un diagrama EER?

- a) Con un rectángulo conectado a un rombo
 - b) Con un círculo que conecta una entidad general con sus subentidades
 - c) Con una línea punteada entre entidades
 - d) Con una flecha doble entre relaciones
-

4. ¿Cuál de las siguientes afirmaciones es verdadera sobre la generalización y especialización?

- a) La especialización combina varias entidades en una sola
- b) La generalización descompone una entidad en entidades más específicas
- c) La generalización crea una entidad padre a partir de entidades hijas
- d) La especialización no permite heredar atributos

5. ¿Cuál es la herencia en el modelo EER?

- a) Un mecanismo para compartir atributos y relaciones entre entidades
- b) La combinación de varias claves primarias en una sola
- c) Un tipo de relación específica entre dos entidades
- d) La eliminación de datos redundantes en una tabla

6. ¿Qué tipo de restricción de especialización permite que una instancia pertenezca a múltiples subclases?

- a) Exclusiva
- b) Superpuesta
- c) Disjunta
- d) Binaria

7. ¿Qué representa una entidad supertipo en un modelo EER?

- a) Una entidad que agrupa atributos comunes para varias subentidades
- b) Una entidad que solo almacena claves primarias
- c) Una entidad que no tiene atributos propios
- d) Una entidad que solo se usa en relaciones ternarias

8. ¿Cómo se denomina una restricción de especialización en la que cada instancia de la entidad padre solo puede pertenecer a una subclase?

- a) Completa
- b) Disjunta
- c) Generalizada
- d) Jerárquica

9. ¿Qué significa que una especialización sea total en un modelo EER?

- a) Que toda instancia de la entidad padre debe pertenecer al menos a una subentidad
- b) Que una entidad puede existir sin pertenecer a ninguna subentidad

- c) Que la especialización permite herencia múltiple
 - d) Que se deben eliminar relaciones redundantes
-

10. ¿Cuál de las siguientes características es exclusiva del modelo EER y no está en el MER tradicional?

- a) Relaciones entre entidades
- b) Atributos multivaluados
- c) Generalización y especialización
- d) Cardinalidad 1:N

Test: Transformación del Modelo Entidad-Relación al Modelo Relacional

Instrucciones:

Selecciona la opción correcta en cada pregunta.

1. ¿Qué representa una entidad en el modelo relacional después de la transformación?

- a) Un atributo en una tabla
 - b) Una clave primaria
 - c) Una fila en una tabla
 - d) Una tabla en la base de datos
-

2. ¿Cómo se convierten los atributos de una entidad en el modelo relacional?

- a) Se convierten en columnas dentro de una tabla
 - b) Se transforman en claves foráneas
 - c) Se eliminan al pasar al modelo relacional
 - d) Se convierten en entidades separadas
-

3. ¿Qué sucede con la clave primaria de una entidad al transformarla en una tabla relacional?

- a) Se convierte en una clave foránea
 - b) Se mantiene como clave primaria en la tabla
 - c) Se elimina porque no es necesaria
 - d) Se duplica en varias tablas
-

4. ¿Cómo se representan las relaciones 1:N en el modelo relacional?

- a) Mediante una tabla intermedia
 - b) A través de la repetición de registros
 - c) Mediante una clave foránea en la tabla del lado N
 - d) Se eliminan en la transformación
-

5. ¿Cómo se transforman las relaciones N:M en el modelo relacional?

- a) Se crean como una clave foránea en una de las tablas
 - b) Se eliminan porque no son compatibles con el modelo relacional
 - c) Se crean mediante una tabla intermedia con claves foráneas de ambas entidades
 - d) Se combinan ambas entidades en una sola tabla
-

6. ¿Qué ocurre con los atributos multivaluados en la transformación al modelo relacional?

- a) Se convierten en columnas adicionales en la tabla original
 - b) Se crean como una tabla separada con una relación 1:N
 - c) Se eliminan porque no son compatibles con el modelo relacional
 - d) Se almacenan como una lista dentro de un solo campo
-

7. ¿Cómo se representan las entidades débiles en el modelo relacional?

- a) Como una tabla con clave primaria propia sin relación con otras tablas
- b) Como una tabla que incluye la clave primaria de la entidad fuerte como parte de su clave primaria
- c) Como una clave foránea en la tabla principal
- d) No necesitan ser representadas en el modelo relacional

8. ¿Cómo se representan las relaciones de especialización/generalización en el modelo relacional?

- a) Mediante una sola tabla que incluye todos los atributos y un campo de tipo
- b) Mediante una tabla para la entidad padre y tablas separadas para cada subentidad
- c) Se eliminan en la transformación al modelo relacional
- d) Se convierten en una relación N:M

9. ¿Qué pasa con los atributos compuestos cuando se transforman al modelo relacional?

- a) Se descomponen en atributos individuales dentro de la tabla
- b) Se almacenan como un solo campo de texto
- c) Se eliminan porque no son compatibles
- d) Se representan como claves foráneas

10. ¿Cuál es el propósito principal de transformar un modelo Entidad-Relación en un modelo relacional?

- a) Mejorar la velocidad de consulta
- b) Asegurar que la base de datos esté normalizada y lista para implementación
- c) Convertir datos en formato JSON
- d) Reducir el tamaño de la base de datos

Test: Normalización en Bases de Datos Relacionales

Instrucciones:

Selecciona la opción correcta en cada pregunta.

1. ¿Cuál es el propósito principal de la normalización en bases de datos?

- a) Mejorar la velocidad de consulta eliminando índices
- b) Reducir la redundancia y mejorar la integridad de los datos
- c) Aumentar el número de tablas para complicar el diseño
- d) Reemplazar claves primarias por claves foráneas

2. ¿Qué se debe evitar en una base de datos normalizada?

- a) La presencia de claves primarias
- b) La duplicación y redundancia de datos
- c) El uso de claves foráneas
- d) La creación de múltiples relaciones

3. ¿Cuál es la primera forma normal (1FN)?

- a) Los atributos deben ser atómicos y la tabla debe tener una clave primaria
- b) No debe haber dependencias transitivas
- c) Cada tabla debe tener al menos una clave foránea
- d) La base de datos debe eliminar todas las claves primarias

4. ¿Cuál de los siguientes problemas resuelve la segunda forma normal (2FN)?

- a) Dependencias funcionales parciales
- b) Atributos multivaluados
- c) Dependencias transitivas
- d) Eliminación de claves foráneas

5. Para que una tabla esté en segunda forma normal (2FN), ¿qué requisito debe cumplir?

- a) Tener una clave primaria y no tener dependencias funcionales parciales
- b) No debe tener claves primarias compuestas
- c) Todos los atributos deben depender directamente de la clave foránea
- d) No debe haber claves foráneas en la tabla

6. ¿Qué problema corrige la tercera forma normal (3FN)?

- a) La existencia de claves primarias compuestas
- b) Las dependencias funcionales transitivas
- c) La normalización excesiva de las tablas
- d) La eliminación de claves foráneas

7. ¿Qué condición adicional se debe cumplir para que una tabla esté en tercera forma normal (3FN)?

- a) Todo atributo no clave debe depender únicamente de la clave primaria
 - b) Los atributos deben ser multivaluados
 - c) Todas las claves foráneas deben convertirse en claves primarias
 - d) Se deben fusionar todas las tablas en una sola
-

8. ¿Qué forma normal introduce el concepto de "dependencias de multivalor"?

- a) Segunda forma normal (2FN)
 - b) Tercera forma normal (3FN)
 - c) Cuarta forma normal (4FN)
 - d) Quinta forma normal (5FN)
-

9. ¿Qué sucede si una base de datos está sobre-normalizada?

- a) Se mejora el rendimiento en todas las consultas
 - b) Se eliminan completamente las claves foráneas
 - c) Se incrementa la complejidad de las consultas y puede afectar el rendimiento
 - d) No hay ningún impacto en el rendimiento
-

10. ¿Cuándo es recomendable dejar de normalizar una base de datos?

- a) Cuando la base de datos ya no tiene redundancia alguna
- b) Cuando la normalización comienza a afectar el rendimiento de las consultas
- c) Cuando se han eliminado todas las claves primarias
- d) Cuando todas las tablas contienen un solo atributo

Test: DDL (Lenguaje de Definición de Datos) y DQL (Lenguaje de Consulta de Datos) en Bases de Datos Relacionales

Instrucciones:

Selecciona la opción correcta en cada pregunta.

1. ¿Qué significa DDL en bases de datos relacionales?

- a) Data Definition Language (Lenguaje de Definición de Datos)
- b) Data Data Language (Lenguaje de Datos de Datos)
- c) Data Distribution Language (Lenguaje de Distribución de Datos)
- d) Data Design Language (Lenguaje de Diseño de Datos)

2. ¿Cuál es la principal función del DDL?

- a) Consultar datos de la base de datos
 - b) Definir la estructura de las tablas, índices y otras estructuras de la base de datos
 - c) Insertar datos en las tablas
 - d) Eliminar datos de la base de datos
-

3. ¿Cuál de las siguientes sentencias es una sentencia DDL?

- a) **SELECT**
 - b) **INSERT**
 - c) **DROP**
 - d) **UPDATE**
-

4. ¿Qué hace la sentencia **CREATE en DDL?**

- a) Modifica la estructura de una tabla
 - b) Crea un nuevo usuario de base de datos
 - c) Crea una nueva tabla, vista o base de datos
 - d) Elimina una tabla o base de datos
-

5. ¿Qué sentencia en DDL se utiliza para eliminar una tabla de la base de datos?

- a) **DELETE**
 - b) **REMOVE**
 - c) **DROP**
 - d) **DESTROY**
-

6. ¿Qué significa DQL en bases de datos relacionales?

- a) Data Query Language (Lenguaje de Consulta de Datos)
 - b) Data Quality Language (Lenguaje de Calidad de Datos)
 - c) Data Query License (Licencia de Consulta de Datos)
 - d) Data Quick Language (Lenguaje de Consulta Rápida de Datos)
-

7. ¿Cuál es la principal función de DQL?

- a) Modificar la estructura de la base de datos
 - b) Consultar o recuperar datos de la base de datos
 - c) Eliminar tablas de la base de datos
 - d) Crear índices para optimizar las consultas
-

8. ¿Cuál de las siguientes sentencias es una sentencia DQL?

- a) **SELECT**
 - b) **ALTER**
 - c) **DROP**
 - d) **CREATE**
-

9. ¿Qué hace la sentencia **SELECT en DQL?**

- a) Modifica una tabla existente
 - b) Elimina una fila de una tabla
 - c) Recupera datos de una o más tablas
 - d) Crea una nueva base de datos
-

10. ¿Qué diferencia existe entre **DELETE y **DROP** en SQL?**

- a) **DELETE** elimina datos, mientras que **DROP** elimina la estructura de la tabla
- b) **DELETE** elimina la tabla, mientras que **DROP** elimina los datos
- c) **DELETE** y **DROP** hacen lo mismo, solo que en diferentes bases de datos
- d) **DELETE** es una sentencia DQL, mientras que **DROP** es una sentencia DML

Test: DML (Lenguaje de Manipulación de Datos) y DCL (Lenguaje de Control de Datos) en Bases de Datos Relacionales

Instrucciones:

Selecciona la opción correcta en cada pregunta.

1. ¿Qué significa DML en bases de datos relacionales?

- a) Data Manipulation Language (Lenguaje de Manipulación de Datos)
 - b) Data Modeling Language (Lenguaje de Modelado de Datos)
 - c) Data Management Language (Lenguaje de Gestión de Datos)
 - d) Data Migration Language (Lenguaje de Migración de Datos)
-

2. ¿Cuál es la principal función de DML?

- a) Consultar datos de la base de datos
 - b) Definir la estructura de las tablas y relaciones
 - c) Manipular o modificar los datos de las tablas
 - d) Controlar el acceso a la base de datos
-

3. ¿Cuál de las siguientes sentencias es una sentencia DML?

- a) **SELECT**
 - b) **INSERT**
 - c) **DROP**
 - d) **CREATE**
-

4. ¿Qué hace la sentencia **INSERT en DML?**

- a) Elimina registros de una tabla
 - b) Recupera registros de una tabla
 - c) Modifica registros existentes
 - d) Agrega nuevos registros a una tabla
-

5. ¿Qué sentencia en DML se utiliza para actualizar registros existentes en una tabla?

- a) **SELECT**
 - b) **DELETE**
 - c) **UPDATE**
 - d) **ALTER**
-

6. ¿Qué significa DCL en bases de datos relacionales?

- a) Data Control Language (Lenguaje de Control de Datos)
- b) Data Creation Language (Lenguaje de Creación de Datos)
- c) Data Classification Language (Lenguaje de Clasificación de Datos)
- d) Data Communication Language (Lenguaje de Comunicación de Datos)

7. ¿Cuál es la principal función de DCL?

- a) Modificar la estructura de la base de datos
- b) Manipular los datos dentro de las tablas
- c) Controlar el acceso y los privilegios sobre los datos
- d) Eliminar tablas y bases de datos

8. ¿Cuál de las siguientes sentencias es una sentencia DCL?

- a) GRANT
- b) DELETE
- c) INSERT
- d) SELECT

9. ¿Qué hace la sentencia GRANT en DCL?

- a) Otorgar permisos de acceso a usuarios o roles sobre la base de datos
- b) Elimina un usuario o rol de la base de datos
- c) Modifica los registros de una tabla
- d) Crea un nuevo índice en la base de datos

10. ¿Qué hace la sentencia REVOKE en DCL?

- a) Revoca permisos previamente otorgados a usuarios o roles
- b) Recupera registros de la base de datos
- c) Crea nuevas tablas
- d) Modificar registros existentes en una tabla

Test: Vistas, Transacciones y Bloqueos, Programación SQL, Cursores, Disparadores y Eventos en Bases de Datos Relacionales

Instrucciones:

Selecciona la opción correcta en cada pregunta.

1. Vistas

1. ¿Qué es una vista en una base de datos relacional?

- a) Una copia de seguridad de la base de datos
- b) Una tabla virtual que muestra datos de una o más tablas
- c) Una tabla física que contiene los datos almacenados
- d) Un índice creado sobre una tabla

2. ¿Cuál es la principal ventaja de usar vistas?

- a) Mejorar el rendimiento de las consultas
- b) Ocultar complejidad y simplificar el acceso a los datos
- c) Mejorar la seguridad del sistema operativo
- d) Almacenar datos duplicados para agilizar el acceso

3. ¿Cuál de las siguientes sentencias se usa para crear una vista en SQL?

- a) `CREATE INDEX`
- b) `CREATE VIEW`
- c) `CREATE TABLE`
- d) `SELECT`

4. ¿Qué tipo de operación no se puede realizar directamente sobre una vista?

- a) Consultar
 - b) Insertar
 - c) Actualizar
 - d) Crear índices
-

2. Transacciones y Bloqueos

5. ¿Qué es una transacción en bases de datos relacionales?

- a) Un proceso de respaldo de los datos
- b) Un conjunto de operaciones que se ejecutan de manera atómica
- c) Un comando SQL que permite crear una tabla
- d) Una consulta que obtiene datos de varias tablas

6. ¿Qué es un bloqueo (lock) en el contexto de bases de datos?

- a) Un mecanismo para proteger la base de datos de accesos no autorizados
- b) Una técnica para mejorar el rendimiento de las consultas
- c) Un control de acceso que impide que varios usuarios modifiquen los mismos datos simultáneamente
- d) Un procedimiento que genera copias de seguridad automáticas

7. ¿Cuál de los siguientes es un tipo de aislamiento de transacciones?

- a) Serialización
- b) Borrado lógico
- c) Comprobación de integridad
- d) Sincronización

8. ¿Qué sucede cuando se realiza un COMMIT en una transacción?

- a) Los cambios realizados en la transacción se revierten
 - b) Los cambios realizados en la transacción se guardan permanentemente en la base de datos
 - c) Los cambios no se guardan hasta que se cierra la sesión
 - d) La transacción se cancela automáticamente
-

3. Programación SQL. Funciones y Procedimientos

9. ¿Qué es una función en SQL?

- a) Un bloque de código que se ejecuta solo una vez
- b) Un conjunto de instrucciones que devuelven un valor y se pueden reutilizar
- c) Un tipo de índice creado para optimizar las consultas
- d) Un procedimiento que permite modificar datos

10. ¿Qué es un procedimiento almacenado en SQL?

- a) Una operación matemática que devuelve un valor
- b) Un conjunto de consultas que se ejecutan en un solo bloque de código
- c) Un bloque de código que no devuelve valores
- d) Una tabla que almacena las funciones disponibles en la base de datos

11. ¿Cuál de las siguientes sentencias se usa para crear un procedimiento almacenado en SQL?

- a) CREATE PROCEDURE
- b) CREATE FUNCTION
- c) CREATE DATABASE
- d) EXECUTE

12. ¿Qué hace una función SQL que devuelve un valor?

- a) Modifica una tabla de la base de datos
 - b) Recuperar y manipular datos, devolviendo un solo valor
 - c) Genera una vista de la base de datos
 - d) Crea nuevas tablas en la base de datos
-

4. Cursores

13. ¿Qué es un cursor en SQL?

- a) Un tipo de índice para mejorar el rendimiento
- b) Un objeto que permite recorrer y manipular los resultados de una consulta fila por fila
- c) Un tipo de transacción que garantiza que los datos sean coherentes
- d) Un procedimiento utilizado para modificar los datos de una tabla

14. ¿Qué sentencia se usa para declarar un cursor en SQL?

- a) DECLARE
- b) CREATE
- c) OPEN
- d) EXECUTE

15. ¿Cuál es la función principal de un cursor?

- a) Optimizar la ejecución de consultas complejas
 - b) Ejecutar una consulta sin devolver resultados
 - c) Iterar sobre los resultados de una consulta, procesándolos fila por fila
 - d) Crear una tabla temporal con los resultados de una consulta
-

5. Disparadores y Eventos

16. ¿Qué es un disparador (trigger) en bases de datos?

- a) Una operación que se ejecuta en respuesta a una consulta SELECT
- b) Un mecanismo que se activa automáticamente en respuesta a ciertos eventos en la base de datos
- c) Una función que mejora el rendimiento de las consultas
- d) Un tipo de vista que se crea para ejecutar acciones automáticamente

17. ¿Cuál de los siguientes es un evento que puede activar un disparador?

- a) Una actualización de los datos de una tabla
- b) La creación de una nueva vista
- c) La consulta de datos con SELECT
- d) La modificación de la estructura de una tabla

18. ¿Cuál de las siguientes acciones puede realizar un disparador?

- a) Actualizar datos en otra tabla automáticamente
- b) Modificar la estructura de una tabla
- c) Eliminar una tabla entera
- d) Crear nuevas vistas basadas en los datos modificados

19. ¿Qué es un evento en SQL?

- a) Una transacción que se ejecuta de manera programada
- b) Una acción automática que ocurre en un momento específico en la base de datos
- c) Un tipo de índice creado para optimizar las consultas
- d) Una función que devuelve un conjunto de resultados específicos

20. ¿Cómo se define un disparador en SQL?

- a) Usando la sentencia `CREATE TRIGGER`
- b) Usando la sentencia `DECLARE TRIGGER`
- c) Usando la sentencia `CREATE EVENT`
- d) Usando la sentencia `CREATE PROCEDURE`

1. Introducción a las Clases y Objetos

1. ¿Qué es una clase en programación orientada a objetos?

- a) Un objeto que contiene funciones y variables
- b) Un modelo o plantilla que define las características y comportamientos de los objetos
- c) Una estructura de datos simple
- d) Un tipo de función que retorna un objeto

2. ¿Qué es un objeto en programación orientada a objetos?

- a) Una variable que almacena un valor
- b) Una instancia de una clase
- c) Una función dentro de una clase
- d) Un tipo de base de datos relacional

3. ¿Cómo se crea un objeto a partir de una clase en Java?

- a) Usando la palabra clave `new` seguida del nombre de la clase
- b) Declarando la clase como `final`

- c) Definiendo el constructor de la clase
- d) Usando la palabra clave `object`

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

- a) La capacidad de una clase de heredar propiedades de otra
 - b) El proceso de ocultar los detalles internos de una clase y exponer sólo lo necesario
 - c) La capacidad de un objeto de ser reutilizado en diferentes clases
 - d) El proceso de compartir funciones entre diferentes objetos
-

2. El Desfase Objeto-Relacional. Herramientas de Mapeado (JPA y ORMLite)

5. ¿Qué es el desfase objeto-relacional (ORM)?

- a) La conversión de datos de un formato XML a un formato JSON
- b) La conversión entre objetos de programación y tablas de bases de datos relacionales
- c) El proceso de crear objetos dentro de la base de datos
- d) La creación de relaciones entre objetos de diferentes bases de datos

6. ¿Qué herramienta se utiliza comúnmente para resolver el desfase objeto-relacional en Java?

- a) SQL
- b) JPA (Java Persistence API)
- c) REST API
- d) MongoDB

7. ¿Qué es JPA (Java Persistence API)?

- a) Un framework para crear aplicaciones web
- b) Un API que permite interactuar con bases de datos relacionales mediante objetos Java
- c) Un tipo de base de datos NoSQL
- d) Un lenguaje de programación para bases de datos

8. ¿Qué es ORMLite?

- a) Una base de datos relacional
- b) Un framework ligero para mapeo objeto-relacional en Java
- c) Una herramienta de administración de bases de datos NoSQL
- d) Una API de comunicación entre bases de datos relacionales

9. ¿Cuál es la ventaja de usar herramientas ORM como JPA o ORMLite?

- a) Evitar la necesidad de crear tablas de base de datos
- b) Simplificar la conversión de objetos en código Java a datos en bases de datos

- c) Acelerar la creación de interfaces de usuario
 - d) Eliminar la necesidad de relaciones entre tablas
-

3. Bases de Datos NoSQL (MongoDB)

10. ¿Qué tipo de base de datos es MongoDB?

- a) Relacional
- b) NoSQL
- c) Distribuida
- d) In-memory

11. ¿Qué modelo de datos utiliza MongoDB?

- a) Modelo basado en tablas
- b) Modelo basado en columnas
- c) Modelo basado en documentos
- d) Modelo jerárquico

12. ¿Qué es un documento en MongoDB?

- a) Una colección de datos estructurados como JSON o BSON
- b) Una tabla que contiene filas de datos
- c) Una base de datos relacional en formato binario
- d) Un archivo de texto plano que contiene datos de la base de datos

13. ¿Cómo se llama a una colección de documentos en MongoDB?

- a) Tabla
- b) Entidad
- c) Collection
- d) Schema

14. ¿Qué operación se usa para insertar un documento en una colección de MongoDB?

- a) `INSERT INTO`
 - b) `CREATE`
 - c) `INSERT`
 - d) `db.collection.insertOne()`
-

4. Bases de Datos en la Nube (Firebase)

15. ¿Qué es Firebase?

- a) Un sistema de gestión de bases de datos relacionales
- b) Un servicio de almacenamiento de archivos en la nube
- c) Una plataforma de desarrollo de aplicaciones que incluye bases de datos en tiempo real
- d) Un motor de bases de datos NoSQL local

16. ¿Cuál de las siguientes características es propia de Firebase?

- a) Soporta solo bases de datos SQL
- b) Es una plataforma exclusivamente para aplicaciones móviles
- c) Permite sincronización en tiempo real de datos entre clientes
- d) Sólo permite almacenar imágenes y archivos en la nube

17. ¿Qué tipo de base de datos ofrece Firebase para almacenamiento de datos?

- a) Relacional
- b) NoSQL en tiempo real
- c) SQL tradicional
- d) Documentos en JSON

18. ¿Qué función cumple Firebase Authentication?

- a) Gestiona los pagos dentro de las aplicaciones
- b) Permite la integración con bases de datos NoSQL
- c) Facilita la autenticación y gestión de usuarios en las aplicaciones
- d) Ofrece almacenamiento en la nube

19. ¿Cuál de las siguientes operaciones puedes realizar con Firebase Firestore?

- a) Consultar y almacenar documentos de manera estructurada
- b) Crear tablas SQL
- c) Realizar backup de bases de datos locales
- d) Convertir datos a formato JSON

20. ¿Cuál es una ventaja de usar Firebase para el desarrollo de aplicaciones móviles?

- a) Permite almacenar sólo datos en formato de texto
- b) Ofrece un sistema de bases de datos relacionales
- c) Soporta sincronización de datos en tiempo real sin necesidad de servidores adicionales
- d) Solo es compatible con aplicaciones Android