**INVESTIGACION CIENCIA DE DATOS**

Autores:

Dayan Berrio – Juan David Suarez – Andres Felipe Monsalve – Santiago Gallego

Institución: SENA

Ficha:2795599

Fecha: 15 Febrero 2025

**Investigar:**

1. **Entorno Personal de Aprendizaje**
2. **Métodos de Aprendizaje**
3. **Tipos de Aprendizaje**
4. **¿Qué es una Base de Datos?**
5. **¿Cuáles son los Tipos de Base de Datos?**
6. **¿Qué es la Ciencia de Datos?**
7. **¿Qué es Python – Historia?**
8. **Bases de Datos Relacionales y No Relacionales**
9. **Fabricantes de Bases de Datos**
   * ¿Qué son?
   * ¿Cómo se integran?
   * ¿Para qué se utilizan?
10. **Mapa Conceptual**
11. **Replit + GitHub**

**1. Entorno Personal de Aprendizaje**

Hace referencia al conjunto de recursos, herramientas, estrategias y personas que cada individuo utiliza para adquirir conocimiento. Esto incluye: recursos digitales, redes sociales, comunidades de aprendizaje, métodos de estudio y fuentes de información como libros, revistas y experiencias prácticas.

**2. Métodos de Aprendizaje**

* **Aprendizaje Basado en Problemas**: Resuelve problemas del mundo real. Fomenta el pensamiento crítico.
* **Aprendizaje Cooperativo**: Se trabaja en equipo para aprender. Promueve la comunicación y el respeto mutuo.
* **Aprendizaje Experiencial**: Aprender haciendo y reflexionando. Aplicar la teoría en la práctica.
* **Aprendizaje Autodidacta**: Se aprende de forma independiente a través de la investigación y la práctica.
* **Aprendizaje Activo**: El estudiante se involucra activamente en proyectos, debates y experimentos.

**3. Tipos de Aprendizaje**

* **Visual**: Aprendes mejor con imágenes y gráficos. Prefieres los mapas mentales y diagramas.  
  **Estrategias**: Usa videos, presentaciones visuales, y toma notas con dibujos y símbolos.
* **Auditivo**: Aprendes mejor escuchando y hablando. Prefieres debates y discusiones.  
  **Estrategias**: Escucha podcasts, audiolibros, participa en grupos de estudio y explica conceptos.
* **Kinestésico**: Aprendes mejor moviéndote y tocando. Prefieres actividades prácticas y experimentos.  
  **Estrategias**: Realiza experimentos, proyectos, y utiliza modelos y simulaciones.
* **Memorístico**:  
  **Repetición**: Repite la información varias veces.  
  **Asociación**: Conecta la nueva información con algo conocido.  
  **Mnemotecnia**: Crea acrónimos y rimas.

**4. ¿Qué es una Base de Datos?**

Una base de datos es un conjunto organizado de datos estructurados que se almacenan electrónicamente en un sistema informático. Permite almacenar, acceder, gestionar y actualizar datos de manera eficiente.  
**Propósito**: Facilitar el acceso, gestión y actualización de grandes volúmenes de información. Se utilizan en diversas aplicaciones, desde gestión de inventarios hasta administración de clientes.

**5. Tipos de Bases de Datos**

* **Relacionales**: Organizan los datos en tablas con filas y columnas, usando claves primarias y foráneas para establecer relaciones entre ellas. Ejemplos: MySQL, PostgreSQL.
* **No Relacionales (NoSQL)**: Almacenan datos en formatos diferentes a las tablas, como documentos, grafos o clave-valor. Son más flexibles y escalables. Ejemplos: MongoDB, Cassandra.

**6. ¿Qué es Ciencia de Datos?**

* **Estadística**: Utiliza métodos estadísticos para analizar datos y extraer patrones significativos.
* **Algoritmos**: Se emplean algoritmos de aprendizaje automático para predecir resultados y tomar decisiones basadas en los datos.
* **Bases de Datos**: Es fundamental tener acceso a datos almacenados en bases de datos para realizar análisis y modelos.

**7. ¿Qué es Python – Historia?**

Python es un lenguaje de programación de alto nivel, interpretado, de propósito general y fácil de aprender.  
**Versatilidad**: Se utiliza en una amplia gama de aplicaciones, desde desarrollo web hasta inteligencia artificial y ciencia de datos.  
**Comunidad activa**: Posee una comunidad de desarrolladores que contribuyen con bibliotecas y herramientas.

**Historia de Python**:

* **Nacimiento**: Fue creado por Guido van Rossum, quien quería un sucesor del lenguaje ABC, usado en el centro de investigación CWI. Su creación comenzó en diciembre de 1989 como un proyecto “hobby”.
* **Inspiración**: Se inspiró en Monty Python, lo que da origen al nombre. Quería un lenguaje fácil de usar, corto, único y algo misterioso.

**Evolución**:

* **Python 1.x (1994-2000)**: Introducción de características como comprensión de listas y soporte para módulos de extensión.
* **Python 2.x (2000-2010)**: Mejoras significativas y la inclusión del sistema de recolección de basura y el módulo unittest.
* **Python 3.x (desde 2008)**: Introducción de cambios importantes, como el uso de print() como función, la división de enteros y el soporte Unicode.

**8. Bases de Datos Relacionales y No Relacionales**

**1. Bases de Datos Relacionales (SQL)**

* **¿Qué son?**: Son bases de datos organizadas en tablas con filas y columnas. Se gestionan usando SQL (Structured Query Language).
* **¿Cómo se integran?**: Se conectan a aplicaciones a través de controladores y lenguajes como Python, Java, PHP o Node.js.
* **¿Para qué se utilizan?**: Son ideales para aplicaciones que requieren integridad y relaciones bien definidas, como sistemas bancarios o ERPs.

**Ejemplos**:

* MySQL
* PostgreSQL
* Microsoft SQL Server
* Oracle Database

**2. Bases de Datos No Relacionales (NoSQL)**

* **¿Qué son?**: Almacenan datos en formatos como documentos, grafos o clave-valor, siendo más flexibles y escalables.
* **¿Cómo se integran?**: Se accede a ellas mediante APIs y drivers específicos, utilizando lenguajes como JavaScript, Python o Go.
* **¿Para qué se utilizan?**: Son ideales para aplicaciones que manejan grandes volúmenes de datos o requieren escalabilidad horizontal.

**Ejemplos**:

* MongoDB
* Redis
* Cassandra
* Neo4j

**Diferencias Clave entre SQL y NoSQL**:

| **Característica** | **SQL (Relacionales)** | **NoSQL (No Relacionales)** |
| --- | --- | --- |
| **Modelo de datos** | Basado en tablas | Flexible (documentos, grafos, clave-valor) |
| **Lenguaje** | SQL | JSON, BSON, APIs propietarias |
| **Escalabilidad** | Vertical (aumento de recursos del servidor) | Horizontal (distribución en varios servidores) |
| **Transacciones** | ACID (Atomicidad, Consistencia, Aislamiento, Durabilidad) | Eventualmente consistente, CAP Theorem |
| **Uso recomendado** | Finanzas, ERP, CRM, comercio electrónico | Big Data, IoT, redes sociales, aplicaciones en tiempo real |

**9. Mapa Conceptual**

Un mapa conceptual es una herramienta visual que organiza información y muestra las relaciones jerárquicas entre conceptos clave. Se compone de nodos (conceptos) y líneas o enlaces que los conectan.

**Características**:

* **Jerárquico**: Va de lo general a lo específico.
* **Conectado**: Muestra relaciones a través de enlaces.
* **Claro y conciso**: Debe ser fácil de entender.
* **Visualmente estructurado**: Usa colores y tamaños para resaltar ideas.

**10. Replit + GitHub**

* **¿Qué es Replit?**: Es un entorno de desarrollo en línea (IDE en la nube) compatible con más de 50 lenguajes de programación.
* **¿Qué es GitHub?**: Es una plataforma basada en Git para almacenar y gestionar código.

**Integración entre Replit y GitHub**:

* Clona repositorios desde GitHub a Replit.
* Sincroniza proyectos entre ambos.
* Colabora con otros desarrolladores.

**Ventajas**:

* Desarrollo en la nube.
* Colaboración en tiempo real.
* Integración fácil.
* Compatible con múltiples lenguajes.