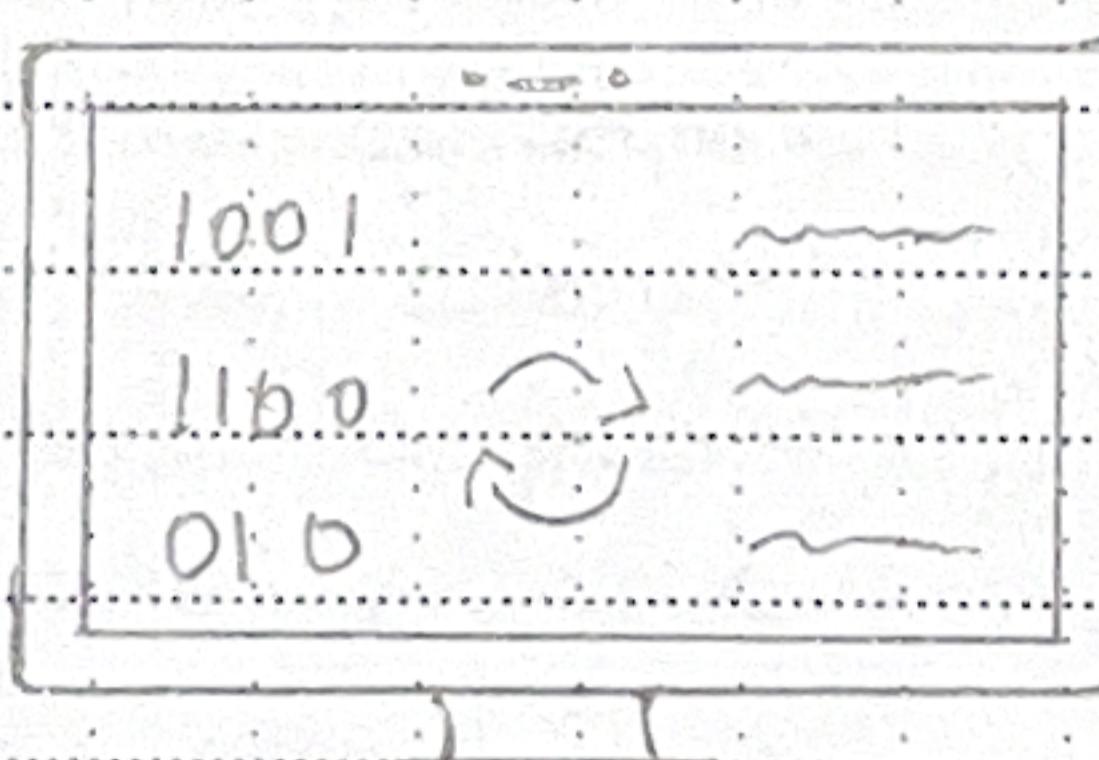
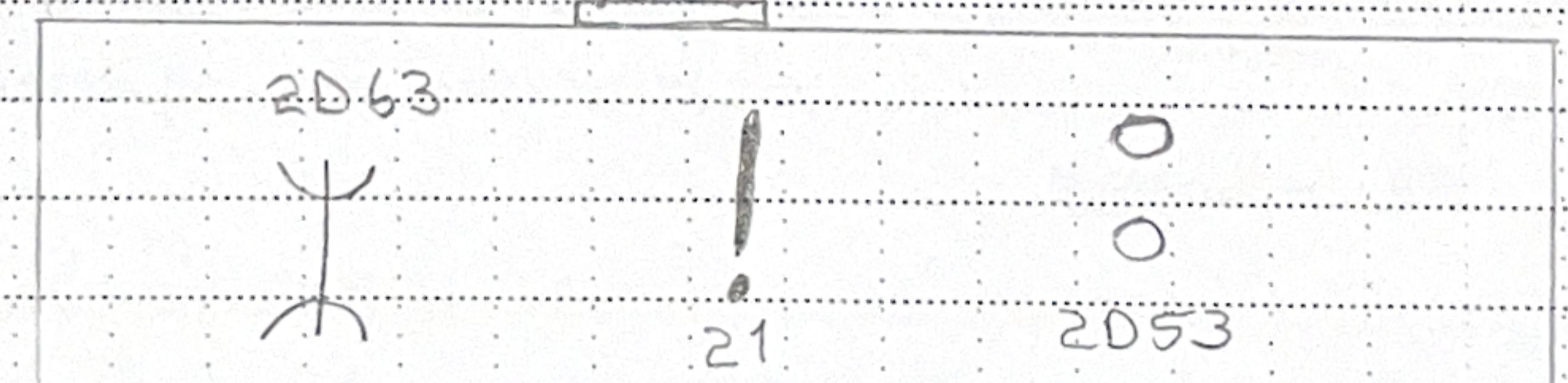


| NAME | CLASS | SPEAKER | DATE & TIME |
|-------------------|-------|-----------------------------|-------------|
| Yanet Encarnacion | 1-6 | Programación Mecatrónica | 19/10/25 |

Title Sistemas de codificación de caracteres

| Keyword | Topic |
|-----------------|--|
| -Sistema | |
| -Códigos | Un sistema de codificación de caracteres (character encoding system) es un esquema asociativo entre un carácter y un determinado código numérico. Es un sistema artificial, creada por convención, que permite transformar de manera inequívoca una secuencia de números o ítems electrónicos en un valor alfanumérico, sea un carácter propio de un alfabeto u otra clase de signo o valor. |
| -Alfanuméricico | |
| -Signos | |
| Questions |   |

Summary: En conclusión, estos permiten representar letras, símbolos y números mediante códigos numéricos. Así las computadoras pueden entender, guardar y mostrar texto correctamente.

| NAME | CLASS | SPEAKER | DATE & TIME |
|-------------------------------|-------|-----------------------------|-------------|
| Yanet Caroline Zeneahacion | 2 - 6 | Programación Mecatrónica | 19/10/25 |

Title Sistemas de codificación de caracteres

| Keyword | Topic |
|--|---|
| - ASA - ANSI - 7 bits - 128 caracteres - Estandar - código - sistema | ASCII (American Standard Code for Information Interchange) es un sistema de codificación de caracteres estandarizado basado en el alfabeto latino. Fue creado en 1963 por el ASA (American Standards Association). ASCII usa 7 bits (a veces 8) para representar 128 caracteres: letras, números, signos de puntuación y caracteres de control. |
| Questions | ¿Qué tipo de caracteres puede representar el código ASCII? |
| | Notas: Fue el primer sistema usado para la comunicación entre computadoras y todavía sirve de base para otros sistemas modernos. |
| | Ejemplos: A → 65 a → 97 nota: diferencia de 32 entre mayúsculas y minúsculas. Espacio → 32 |

Summary: ASCII fue el primer sistema de codificación de caracteres. Usa 7 bits para representar letras, números y signos del alfabeto latino. Es la base para los sistemas modernos.

| NAME | CLASS | SPEAKER | DATE & TIME |
|----------------------------|-------|--------------------------|-------------|
| Yanet Caroline Encarnacion | 3 - 6 | Programación Recatrónica | 19/10/25 |

Title Sistemas de codificación de caracteres

Keyword

-Unicode

-bytes

-compatibilidad

-Eficiencia

Topic UTF-8

Unicode Transformation Format - 8 bits

Puede usar entre 1 y 4 bytes para representar cada carácter.

Es el formato más usado en internet porque es compatible con ASCII.

Ventajas:

- Representa caracteres de cualquier idioma.
- Compatible con sistemas antiguos.
- Muy eficiente para textos comunes.

Questions

Ejemplos:

A → 01000001 (1 byte)

ñ → 11000011 10110001 (2 bytes)



→ 11110000 10011111 10011000 10000000
(4 bytes)

Summary:

UTF-8 es una codificación moderna basada en Unicode. Puede usar de 1 a 4 bytes para representar caracteres de cualquier idioma. Es el formato más usado en internet por su compatibilidad con ASCII.

| NAME | CLASS | SPEAKER | DATE & TIME |
|----------------------------|-------|--------------------------|-------------|
| Yanet Caroline Encarnacion | 4 - 6 | Programación Recatrónica | 19/10/25 |

Title Sistemas de codificación de caracteres

Keyword

- unicode
- 16 bits
- Java
- Windows

Topic

UTF-16

Unicode Transformation Format -16 bits

UTF-16 también pertenece al sistema Unicode.

Este usa 2 o 4 bytes para representar cada carácter. Es común en sistemas operativos como Windows y en lenguajes de programación como Java y C#.

Ventajas:

Questions

• Representa la mayoría de los caracteres con solo 2 bytes.

• Adequado para textos multilingües.

Ejemplos:

A → 0041

ñ → 00F1

ü → D83D DE00

Summary:

Este es útil para textos que combinan varios idiomas. Usa 2 o 4 bytes por carácter y es común en Java y Windows.

| NAME | CLASS | SPEAKER | DATE & TIME |
|----------------------------|-------|--------------------------|-------------|
| Yanet Caroline Encarnacion | 5 - 6 | Programación Electrónica | 19/10/25 |

Title Tipos de archivos

Keyword

- Javascript
- Clave-valor
- ligero
- APIs
- Intercambio de datos

Topic

JSON

JavaScript Object Notation

JSON es un formato de texto ligero y fácil de leer. Se usa para intercambiar datos entre aplicaciones. Organiza la información en pares de "clave: valor".

Ventajas:

- Sencillo y rápido.
- Compatible con casi todos los lenguajes.
- Muy usado en páginas web y APIs.

Ejemplo de JSON:

{

 "nombre": "Yanet",
 "edad": 20,
 "País": "República Dominicana"

}

Summary:

JSON es un formato de texto ligero y fácil de leer. Usa pares de "clave: valor" para organizar la información. Es muy usado en aplicaciones web y comunicación entre programas.

| NAME | CLASS | SPEAKER | DATE & TIME |
|-------------------------------|-------|-----------------------------|-------------|
| Yanet Caroline Enearmacion | 6-6 | Programación Electrónica | 19/10/25 |

Title Tipos de archivos

Keyword

- Lenguaje
- Etiquetas
- Estructura
- W3C
- Transporte
- legible

Topic

XML

extensible Markup Language

XML es un lenguaje de marcado creado por el W3C (World Wide Web Consortium), la organización internacional que se encarga de desarrollar normas y estándares para que la web funcione correctamente.

XML sirve para almacenar y transportar información. Usa etiquetas personalizadas (parecidas a HTML).

Questions

ventajas:

Ejemplo:

• Flexible y fácil de entender.

<Persona>

<nombre> Yanet </nombre>

• Compatible con muchos programas.

<edad> 20 </edad>

<País> República

• Se usa en bases de datos y documentos.

Dominicana </País>

</Persona>

Summary: En conclusión, este es un lenguaje marcado que usa etiquetas personalizadas. Sirve para almacenar y transportar información estructurada. Aunque es más largo que JSON, es muy flexible y compatible.