

1. Que es un paradigma de POO

es el estilo o forma en la cual se desarrolla un programa para fines especificos, el objetivo es resolver problemas.

2. Que es una clase, un objeto, un atributo y un metodo

La clase es una "plantilla" donde se definen conjuntos de objetos que tienen características similares.

Objeto es cualquier cosa que se pueda imaginar, ya sea tangible o intangible, se define por características propias del objeto llamadas atributos y operación, permitiendo modificar los atributos, se obtiene de las clases.

CS Escaneado con CamScanner

Atributo son las características abstractas de un objeto, digamos que tenemos al objeto "Persona", al abstraer la idea, se ve que tiene cédula, nombre y edad estas características representan los atributos del objeto.

Metodos funciones o procedimientos que se asocian a una clase para definir el comportamiento de los objetos creados a partir de esa clase, Tienen parámetros con sus tipos respectivamente, pueden tener algún tipo de retorno.

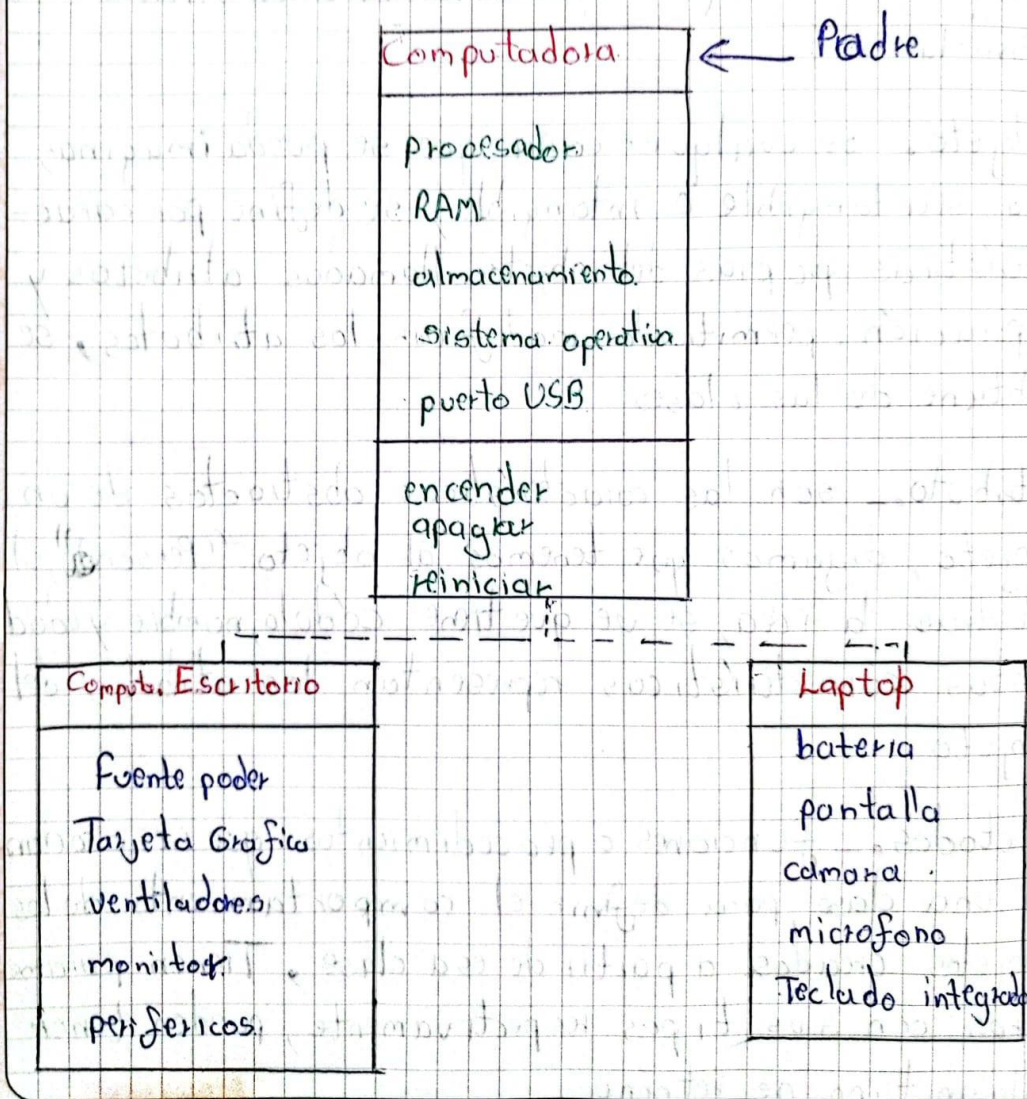
GAMA

CS Escaneado con CamScanner

3. Que es y para que sirve un sistema de control de versionamiento. (SCV).

Son las herramientas y metodos utilizados para administrar o gestionar los cambios que se dan en el software o en conjunto de ficheros, manteniendo un historial de cambios.

4. UML de dos objetos hijos y 1 padre.



Tipo de datos Referencia

Class

plantilla que permite definir objetos, una clase contiene atributos, metodos para determinar el objeto.

```
Class persona {  
    String nombre;  
    int edad;  
}
```

Arrays

Es una colección de elementos del mismo tipo, es como un bloque de memoria que almacena elementos.

```
Int[] numeros = {1, 2, 3, 4};
```

```
Int[] nombres = {Juan, Carlos, Pedro};
```

Interface

tipo abstracto que define un conjunto de metodos que una clase debe implementar, no almacenan datos por si mismos, deben ser llamados.

```
Interface Animal {  
    void hacerSonido();  
}
```

Collections

Es una estructura de datos dinámicos, permite almacenar y manipular grupos de elementos.

```
List<Integer> numeros = new ArrayList<>();  
numeros.add(10);  
numeros.add(20);
```

Genericos

Son parametrizados, permiten trabajar con datos de cualquier tipo. Son muy usados en colecciones y estructuras genericas.

```
List<String> nombres = new ArrayList<>();  
nombres.add("Juan");  
nombres.add("Ana");
```


1. Tipos de datos primitivos

Byte

Representa un dato de 8 bits, puede almacenar valores numéricos del -128 a 127

Short

Representa un dato de 16 bits, almacena valores de -32,768 a 32,767 (decimales)

Int

Dato de 32 bits, para almacenar datos enteros de -2,147,483,648 hasta 2,147,483,647

Long

Dato de 64 bits, almacena valores entre -2^{63} a $2^{63}-1$

Float

Sirve para almacenar números como flotante con precisión de (32 bits)

double

almacena números como flotante con doble precisión (64 bits)

boolean

define valores de tipo booleanos (True o False) ocupa 1 Bit de Información

Char

representa un carácter o texto de 16 bits