

## Informe Programación Orientada a Objetos

|                       |  |                                   |  |
|-----------------------|--|-----------------------------------|--|
| Para:                 | Cristian Henao   |                                   |  |
| Ficha:                | 2278879  |                                   |  |
| Creado por:           | Cristian David Amador Mancera  | Actualizado por:                  |  |
| Fecha de la creación: | 14/02/2021   | Fecha de la última actualización: |  |
| Descripción:          | Informe escrito sobre cada concepto investigado, el informe deberá ser construido por cada persona definiendo los conceptos con sus propias palabras según lo que se entendió de los conceptos y donde se evidencie la apropiación del conocimiento. |                                   |  |

### 1. Investigar sobre el concepto de Programación Orientada a Objetos.

Es una forma de programación que nos deja crear o representar cosas de la vida real junto a sus acciones, nos permite ver claramente las interacciones que tiene una cosa con otra, para así hacer más entendible el código, en esta forma de programación el software o programa que codifiquemos se vuelve un conjunto de objetos que tienen que ver entre sí.

Para poder crear uno de estos objetos primero que nada se necesita la estructura, es decir la clase, para saber que es el objeto, que tiene y que hace

### 2. Investigar sobre el uso de constructores Implícitos y Explícitos (desarrollar laboratorio 13 del material de apoyo donde se explica el uso de constructores).

El constructor implícito es el que nos permite crear el objeto de esa clase, con los atributos que tiene por defecto y sin posibilidad de cambiarlos directamente ahí, esta construcción se hace automáticamente cuando se instancia, mientras que el constructor explícito requiere de una intervención, hay que asignarle valores a los datos mínimos que requiere la clase para crearle un objeto.

### 3. Investigar sobre el uso del foreach.

El for each funciona para arreglos, básicamente facilita la iteración en un arreglo, la variable que se asigna al principio es donde se va a almacenar el valor de cada posición del arreglo.

### 4. Investigar sobre el concepto de encapsulación.

Es una forma de limitar el alcance o acceso de los datos que tiene una clase mediante las palabras reservadas, private, public y protected, si no se le asigna ninguno tendrá un alcance default. Normalmente se asignan los atributos de una clase private en este caso se podría modificar o acceder a los atributos con los métodos get y set, deberían ser públicos estos son para su traducción literal al español.

## **5. Investigar sobre el concepto de herencia en Java.**

La herencia es un mecanismo que permite hacer una clase en base a otra, de esta manera le cede sus métodos y atributos, los métodos constructores no se heredan, esto simplifica la reutilización de código de una clase a otra solo usando la palabra `extends` y el nombre de la super clase..

## **6. Investigar sobre el concepto de sobrecarga de métodos (Overloading).**

Nos permite crear un método con el mismo nombre pero con diferentes métodos, solo si tienen diferentes parámetros. Esto sirve para permitir la realización de un mismo tipo de cosa, requiriendo diferentes herramientas para hacer cada una de ellas.

## **7. Investigar sobre el concepto de sobrescritura de métodos (Overriding).**

Es la forma que tiene una clase hijo de no regirse estrictamente de cómo el papa realiza sus métodos, el hijo tiene la oportunidad de hacer la misma cosa pero de manera diferente cuando lo quiera, aunque siempre el método sobrescrito debe llamarse igual al del padre y debe pedir los mismos argumentos además de el tipo de retorno, para que el compilador reconozca la sobrescritura se debe poner `@override` por convención y facilidad.

## **8. Investigar sobre manejo de excepciones en java (uso del Try Catch).**

Las excepciones son heredadas por todos los objetos por su superclase, nos permiten ver que hay algo que no está bien en la ejecución, algunas veces dichas excepciones son intencionales, para poder prever las cosas obvias y poder controlarlas sin que se detenga el programa existe el `try - catch - finally`, nos permiten visibilizar las excepciones y prever que el programa se detenga cuando no debería.

## **9. Desarrollar el laboratorio “Caso\_estudio\_Herencia\_en\_Java.pdf” que encuentra en la sección de actividades, así como las actividades planteadas en dicho documento.**

## **10. Investigar sobre clases abstractas en Java.**

La abstracción nos deja ver principalmente lo más importante de algo, estas clases nos dejan crear métodos con comportamiento común para sus herederos concretos ignorando sus atributos y funciones, no se pueden crear objetos de estas clases, sólo se puede crear métodos abstractos en una clase abstracta así mismo no se puede dar funcionalidad a un método abstracto en una clase abstracta.