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

POO(La programación orientada a objeto) representa acciones o casos de la vida real basados en objetos(personas, un perro, etc).

La programación orientada a objetos consta de 2 divisiones principales que son las clases y los métodos. Las clases son una representación de los objetos(autos, personas,perros vehículos, etc.) y los métodos son las acciones( dormir,correr, acelerar, jugar, frenar, etc…)

la poo se trabaja vinculando diferentes conceptos como clases, objetos, métodos, propiedades, estados, herencia,encapsulacion,etc…

1. Investigar sobre el uso de constructores Implícitos y Explícitos

los constructores implícitos son los que se le asigna valores y los constructores explícitos son los que no se le predetermina un valor si no `por el contrario se le asigna los valores por parámetro o otras formas

Ejemplo

class Foo{

private char grupo;

private int edad;

*}*

*\*constructor implicito\**

Foo(){

grupo=”A”;

edad=”16”;

}

*\*constructor explicito\**

Foo(grupo,edad)

{

this.grupo=grupo;

this.edad=edad;

}

1. Investigar sobre el uso del foreach

El uso del foeach nos permite recorrer arreglos, estructuras de datos, listas. Pero este cuenta con:

for(Tipo de datos del arreglo nombre de la variable: nombre del arreglo){

}

1. Investigar sobre el concepto de Encapsulación

la encapsulacion es la forma que limita el acceso a nuestras variables de nuestras clases de tal forma que podamos tener un mayor control sobre ellas, normalmente utilizando los métodos set/get.

Es una forma de programar de proteger nuestros datos y se tiene diferentes formas de acceso.

private: se utiliza dentro de la misma clase y se utiliza los metodos set/get.

Public: se puede acceder desde toda la aplicación

Default: este permite el acceso desde todo el paquete pero no a otro paquete

1. Investigar sobre el concepto de Herencia en Java

Permite la definición de una clase a partir de la definición de otra clase ya existente. permite compartir métodos y datos en clases, subclases y objetos(se trata de heredar por medio dela palabra extends):

public class marcas extends carros{

}

1. Investigar sobre el concepto de sobre-carga de métodos (Overloading)

La sobrecarga del es la creación de varios métodos con un mismo nombre pero con diferente lista de tipos de parámetros. Cuando se va a llamar el método el sistema define cual llamar dependiendo de los parámetros.

1. Investigar sobre el concepto de sobre-escritura de métodos (Overriding)

Es la forma por el cual una clase que hereda puede definir los métodos de su clase padre, de esta manera se pueden crear nuevos métodos con el mismo nombre de su superclase.

1. Investigar sobre manejo de excepciones en java (uso del Try Catch)

Una excepción y sus subclases indicas situaciones donde la aplicación debería tratar de resolver un “error” de manera transparente al usuario lanzando advertencias.

Cuando se produce una excepción “try” termina y “catch” revise como argumento el objeto THROWABLE por ejemplo:

*\*Bloque 1\**

try{

*\*Bloque 2\**

}catch (excepcion error){

*\*Bloque 3\**

}

*\*Bloque 4\**

*}*

*una excepción se puede dar por ejemplo cuando pide una letra e he ingresan un numero.*

1. Investigar sobre clases Abstractas en Java

Una clase se define como abstracta cuando representa una entidad que no debería ser insustancialidad hay un método abstracto toda la clase debe ser abstracta pero esta clase puede tener métodos concretos abstractos.

public abstract class animal{

public abstract void metodo();

}

public class perro extends animal{

public void metodo(){

System.ot.println(“los animales);

}

}