HERENCIA

La empresa informática “IPM Tech” necesita llevar un registro de todos sus empleados que se encuentran en la oficina central, para eso ha creado un diagrama de clases que debe incluir lo siguiente:   
  
**1) Empleado**  
  
Atributos:  
  
- nombre: tipo cadena (Debe ser nombre y apellido)   
- cedula: tipo cadena   
- edad : entero (Rango entre 18 y 45 años)   
- casado: boolean   
- salario: tipo numérico doble  
  
Métodos:  
- Constructor con y sin parámetros de entrada   
- Método que permita mostrar la clasificación según la edad de acuerdo al siguiente algoritmo:   
Si edad es menor o igual a 21, Principiante   
Si edad es >=22 y <=35, Intermedio   
Si edad es >35, Senior.-  
  
- Imprimir los datos del empleado por pantalla (utilizar salto de línea \n para separar los atributos.  
  
- Un método que permita aumentar el salario en un porcentaje que sería pasado como parámetro al método.  
  
**2. Programador** (Especialización de Empleado). Clase que hereda de Empleado todos los atributos y métodos.  
  
- Atributos:   
  
- lineasDeCodigoPorHora : tipo entero   
  
- lenguajeDominante: tipo cadena   
  
- Metodos:   
  
- Constructor con y sin parámetos de entrada.  
  
**SOLUCION:**  
  
Primero debe definirse la clase Empleado con los atributos señalados en, se definen los métodos con las acciones pertinentes. Posteriormente se define la clase Programador como hija de la clase Empleado, con sus atributos y métodos correspondientes. Cuando se hace referencia a la clase padre se antepone al atributo la palabra super. La palabra this sirve para referenciar el objeto actual.

**/\*\***

**\* ExamnIPM.java**

**\*/**

**class Empleado{**

**String nombre;**

**String cedula;**

**int edad;**

**boolean casado;**

**double salario;**

**Empleado(){**

**}**

**Empleado(String nombre, String cedula, int edad, boolean casado, double salario) {**

**this.nombre = nombre;**

**this.cedula = cedula;**

**this.edad = edad;**

**this.casado = casado;**

**this.salario = salario;**

**}**

**public String Clasificacion(){**

**String c="";**

**if (edad <= 21){**

**c = "Principiante";**

**}**

**if ((edad >=22) & (edad <= 35)){**

**c = "Intermedio";**

**}**

**if (edad > 35){**

**c = "Senior";**

**}**

**return c;**

**}**

**public void ImprimirEmpleado(){**

**System.out.println(this.nombre);**

**System.out.println(this.cedula);**

**System.out.println(this.edad);**

**if (casado) {**

**System.out.println("Casado");**

**}**

**else**

**{**

**System.out.println("Soltero");**

**}**

**System.out.println(this.salario);**

**}**

**public void AumentarSalario(double incremento){**

**salario = salario \* (1 + incremento);**

**}**

**}**

**class Programador extends Empleado{**

**int lineasDeCodigoPorHora;**

**String lenguajeDominante;**

**Programador(String nombre, String cedula, int edad, boolean casado, double salario, int lineasDeCodigoPorHora, String lenguajeDominante){**

**super.nombre = nombre;**

**super.cedula = cedula;**

**super.edad = edad;**

**super.casado = casado;**

**super.salario = salario;**

**this.lineasDeCodigoPorHora = lineasDeCodigoPorHora;**

**this.lenguajeDominante = lenguajeDominante;**

**}**

**Programador(){**

**}**

**}**

**public class ExamnIPM {**

**public static void main(String[] args){**

**Empleado e = new Empleado("Orlando Arróliga","12345",17,true,2000);**

**Programador p1 = new Programador("Flavio","3456",20,false,1000,123,"Visual Basic");**

**p1.nombre = "Flavio";**

**p1.salario = 1000;**

**p1.AumentarSalario(0.25);**

**p1.ImprimirEmpleado();**

**}**

**}**