Prompt 1 : **Quiero que desarrolles un sistema de clases en Java para gestionar el control del gasto público**

**en las ciudades de un país. El país estará compuesto por provincias, y cada provincia tendrá**

**diversas ciudades.**

**Cada pais debera tener las siguientes propiedades: nombre, capital, continente y una lista de**

**sus provincias.**

**Cada provincia debera tener las siguientes propiedades: nombre, capital, pais al que pertenece**

**y una lista de sus ciudades.**

**Cada ciudad deberá contar con un nombre, una población, ingresos provenientes de cinco tipos**

**de impuestos (impuesto1, impuesto2, impuesto3, impuesto4, impuesto5) y gastos en**

**mantenimiento. El sistema deberá ser capaz de identificar qué ciudades gastan más de lo que**

**recaudan y cuáles provincias tienen más de la mitad de sus ciudades en situación de déficit.**

**Asegúrate de seguir las mejores prácticas de programación orientada a objetos, incluyendo un**

**adecuado encapsulamiento y la implementación de métodos getter y setter cuando sea**

**necesario**

Codigo:

Para crear un sistema de clases en Java que gestione el control del gasto público en las ciudades de un país, podemos diseñar las clases Pais, Provincia y Ciudad siguiendo las mejores prácticas de la programación orientada a objetos. A continuación, se presenta una implementación que incluye encapsulamiento, métodos getter y setter, y lógica para identificar ciudades y provincias en situación de déficit.

Clase Pais

java

Copiar código

import java.util.ArrayList;

import java.util.List;

public class Pais {

private String nombre;

private String capital;

private String continente;

private List<Provincia> provincias;

public Pais(String nombre, String capital, String continente) {

this.nombre = nombre;

this.capital = capital;

this.continente = continente;

this.provincias = new ArrayList<>();

}

public String getNombre() {

return nombre;

}

public void setNombre(String nombre) {

this.nombre = nombre;

}

public String getCapital() {

return capital;

}

public void setCapital(String capital) {

this.capital = capital;

}

public String getContinente() {

return continente;

}

public void setContinente(String continente) {

this.continente = continente;

}

public List<Provincia> getProvincias() {

return provincias;

}

public void addProvincia(Provincia provincia) {

provincias.add(provincia);

}

}

Clase Provincia

java

Copiar código

import java.util.ArrayList;

import java.util.List;

public class Provincia {

private String nombre;

private String capital;

private Pais pais;

private List<Ciudad> ciudades;

public Provincia(String nombre, String capital, Pais pais) {

this.nombre = nombre;

this.capital = capital;

this.pais = pais;

this.ciudades = new ArrayList<>();

}

public String getNombre() {

return nombre;

}

public void setNombre(String nombre) {

this.nombre = nombre;

}

public String getCapital() {

return capital;

}

public void setCapital(String capital) {

this.capital = capital;

}

public Pais getPais() {

return pais;

}

public List<Ciudad> getCiudades() {

return ciudades;

}

public void addCiudad(Ciudad ciudad) {

ciudades.add(ciudad);

}

public boolean isInDeficit() {

int ciudadesEnDeficit = 0;

for (Ciudad ciudad : ciudades) {

if (ciudad.isInDeficit()) {

ciudadesEnDeficit++;

}

}

return ciudadesEnDeficit > (ciudades.size() / 2);

}

}

Clase Ciudad

java

Copiar código

public class Ciudad {

private String nombre;

private int poblacion;

private double impuesto1;

private double impuesto2;

private double impuesto3;

private double impuesto4;

private double impuesto5;

private double gastosMantenimiento;

public Ciudad(String nombre, int poblacion, double impuesto1, double impuesto2, double impuesto3, double impuesto4, double impuesto5, double gastosMantenimiento) {

this.nombre = nombre;

this.poblacion = poblacion;

this.impuesto1 = impuesto1;

this.impuesto2 = impuesto2;

this.impuesto3 = impuesto3;

this.impuesto4 = impuesto4;

this.impuesto5 = impuesto5;

this.gastosMantenimiento = gastosMantenimiento;

}

public String getNombre() {

return nombre;

}

public void setNombre(String nombre) {

this.nombre = nombre;

}

public int getPoblacion() {

return poblacion;

}

public void setPoblacion(int poblacion) {

this.poblacion = poblacion;

}

public double getImpuesto1() {

return impuesto1;

}

public void setImpuesto1(double impuesto1) {

this.impuesto1 = impuesto1;

}

public double getImpuesto2() {

return impuesto2;

}

public void setImpuesto2(double impuesto2) {

this.impuesto2 = impuesto2;

}

public double getImpuesto3() {

return impuesto3;

}

public void setImpuesto3(double impuesto3) {

this.impuesto3 = impuesto3;

}

public double getImpuesto4() {

return impuesto4;

}

public void setImpuesto4(double impuesto4) {

this.impuesto4 = impuesto4;

}

public double getImpuesto5() {

return impuesto5;

}

public void setImpuesto5(double impuesto5) {

this.impuesto5 = impuesto5;

}

public double getGastosMantenimiento() {

return gastosMantenimiento;

}

public void setGastosMantenimiento(double gastosMantenimiento) {

this.gastosMantenimiento = gastosMantenimiento;

}

public double getTotalImpuestos() {

return impuesto1 + impuesto2 + impuesto3 + impuesto4 + impuesto5;

}

public boolean isInDeficit() {

return getTotalImpuestos() < gastosMantenimiento;

}

}

Ejemplo de uso

java

Copiar código

public class Main {

public static void main(String[] args) {

// Crear un país

Pais pais = new Pais("País Ejemplo", "Capital Ejemplo", "Continente Ejemplo");

// Crear provincias

Provincia provincia1 = new Provincia("Provincia 1", "Capital Provincia 1", pais);

Provincia provincia2 = new Provincia("Provincia 2", "Capital Provincia 2", pais);

// Agregar provincias al país

pais.addProvincia(provincia1);

pais.addProvincia(provincia2);

// Crear ciudades

Ciudad ciudad1 = new Ciudad("Ciudad 1", 500000, 1000, 2000, 3000, 4000, 5000, 15000);

Ciudad ciudad2 = new Ciudad("Ciudad 2", 300000, 2000, 3000, 4000, 5000, 6000, 20000);

Ciudad ciudad3 = new Ciudad("Ciudad 3", 100000, 500, 1000, 1500, 2000, 2500, 6000);

// Agregar ciudades a las provincias

provincia1.addCiudad(ciudad1);

provincia1.addCiudad(ciudad2);

provincia2.addCiudad(ciudad3);

// Verificar y mostrar ciudades en déficit

for (Provincia provincia : pais.getProvincias()) {

for (Ciudad ciudad : provincia.getCiudades()) {

if (ciudad.isInDeficit()) {

System.out.println("La ciudad " + ciudad.getNombre() + " está en déficit.");

}

}

if (provincia.isInDeficit()) {

System.out.println("La provincia " + provincia.getNombre() + " tiene más de la mitad de sus ciudades en déficit.");

}

}

}

}

Este sistema permite crear un país con sus provincias y ciudades, calcular los ingresos por impuestos de cada ciudad, comparar estos ingresos con los gastos de mantenimiento y determinar si una ciudad o provincia está en déficit.

Informe del código generado :

**El código generado por la IA esta bien estructurado con las clases y métodos correspondientes . Puede verificar si la Ciudad esta en déficit entre otras . Lo único que le corregiría es que puso cada impuesto como atributos double en vez de hacer una clase Impuestos y asociarla a la ciudad**

Prompt 2 :

**Quiero un sistema de clases en Java que modele las especificaciones de un "Aeródromo". De**

**acuerdo con las siguientes especificaciones:**

**Una clase avión con los atributos: id, numeroMatricula, hangar, fechaAdquisicion y propietario.**

**Un a clase TipoAvion, con los atributos: modelo, capacidad y peso.**

**Una clase Hangar, con los atributos: id, capacidad y ubicacion.**

**Una clase Servicio con los atributos fecha, tipoDeCarga.**

**Una clase Persona con los atributos dni, nombre, direccion y telefono.**

**Una clase Mecanico que heredará de Persona, con los atributos salario, turno y**

**tipoDeAvionAutorizado.**

**Una clase Piloto que heredará de Persona, con los atributos salario, tipoAvionAutorizado.**

**Todas las clases con sus getters y setters necesarios.**

**Asegúrate de implementar correctamente la relación entre las clases y mantener el**

**encapsulamiento adecuado de los datos.**

**Código Generado:**

Para modelar un sistema de clases en Java que gestione un aeródromo siguiendo las especificaciones proporcionadas, se pueden crear las clases Avion, TipoAvion, Hangar, Servicio, Persona, Mecanico y Piloto. Estas clases deben estar bien encapsuladas y utilizar los principios de la programación orientada a objetos.

A continuación, se muestra una posible implementación para estas clases:

Clase Avion

java

Copiar código

import java.util.Date;

public class Avion {

private int id;

private String numeroMatricula;

private Hangar hangar;

private Date fechaAdquisicion;

private Persona propietario;

public Avion(int id, String numeroMatricula, Hangar hangar, Date fechaAdquisicion, Persona propietario) {

this.id = id;

this.numeroMatricula = numeroMatricula;

this.hangar = hangar;

this.fechaAdquisicion = fechaAdquisicion;

this.propietario = propietario;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getNumeroMatricula() {

return numeroMatricula;

}

public void setNumeroMatricula(String numeroMatricula) {

this.numeroMatricula = numeroMatricula;

}

public Hangar getHangar() {

return hangar;

}

public void setHangar(Hangar hangar) {

this.hangar = hangar;

}

public Date getFechaAdquisicion() {

return fechaAdquisicion;

}

public void setFechaAdquisicion(Date fechaAdquisicion) {

this.fechaAdquisicion = fechaAdquisicion;

}

public Persona getPropietario() {

return propietario;

}

public void setPropietario(Persona propietario) {

this.propietario = propietario;

}

}

Clase TipoAvion

java

Copiar código

public class TipoAvion {

private String modelo;

private int capacidad;

private double peso;

public TipoAvion(String modelo, int capacidad, double peso) {

this.modelo = modelo;

this.capacidad = capacidad;

this.peso = peso;

}

public String getModelo() {

return modelo;

}

public void setModelo(String modelo) {

this.modelo = modelo;

}

public int getCapacidad() {

return capacidad;

}

public void setCapacidad(int capacidad) {

this.capacidad = capacidad;

}

public double getPeso() {

return peso;

}

public void setPeso(double peso) {

this.peso = peso;

}

}

Clase Hangar

java

Copiar código

public class Hangar {

private int id;

private int capacidad;

private String ubicacion;

public Hangar(int id, int capacidad, String ubicacion) {

this.id = id;

this.capacidad = capacidad;

this.ubicacion = ubicacion;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public int getCapacidad() {

return capacidad;

}

public void setCapacidad(int capacidad) {

this.capacidad = capacidad;

}

public String getUbicacion() {

return ubicacion;

}

public void setUbicacion(String ubicacion) {

this.ubicacion = ubicacion;

}

}

Clase Servicio

java

Copiar código

import java.util.Date;

public class Servicio {

private Date fecha;

private String tipoDeCarga;

public Servicio(Date fecha, String tipoDeCarga) {

this.fecha = fecha;

this.tipoDeCarga = tipoDeCarga;

}

public Date getFecha() {

return fecha;

}

public void setFecha(Date fecha) {

this.fecha = fecha;

}

public String getTipoDeCarga() {

return tipoDeCarga;

}

public void setTipoDeCarga(String tipoDeCarga) {

this.tipoDeCarga = tipoDeCarga;

}

}

Clase Persona

java

Copiar código

public class Persona {

private String dni;

private String nombre;

private String direccion;

private String telefono;

public Persona(String dni, String nombre, String direccion, String telefono) {

this.dni = dni;

this.nombre = nombre;

this.direccion = direccion;

this.telefono = telefono;

}

public String getDni() {

return dni;

}

public void setDni(String dni) {

this.dni = dni;

}

public String getNombre() {

return nombre;

}

public void setNombre(String nombre) {

this.nombre = nombre;

}

public String getDireccion() {

return direccion;

}

public void setDireccion(String direccion) {

this.direccion = direccion;

}

public String getTelefono() {

return telefono;

}

public void setTelefono(String telefono) {

this.telefono = telefono;

}

}

Clase Mecanico

java

Copiar código

public class Mecanico extends Persona {

private double salario;

private String turno;

private TipoAvion tipoDeAvionAutorizado;

public Mecanico(String dni, String nombre, String direccion, String telefono, double salario, String turno, TipoAvion tipoDeAvionAutorizado) {

super(dni, nombre, direccion, telefono);

this.salario = salario;

this.turno = turno;

this.tipoDeAvionAutorizado = tipoDeAvionAutorizado;

}

public double getSalario() {

return salario;

}

public void setSalario(double salario) {

this.salario = salario;

}

public String getTurno() {

return turno;

}

public void setTurno(String turno) {

this.turno = turno;

}

public TipoAvion getTipoDeAvionAutorizado() {

return tipoDeAvionAutorizado;

}

public void setTipoDeAvionAutorizado(TipoAvion tipoDeAvionAutorizado) {

this.tipoDeAvionAutorizado = tipoDeAvionAutorizado;

}

}

Clase Piloto

java

Copiar código

public class Piloto extends Persona {

private double salario;

private TipoAvion tipoAvionAutorizado;

public Piloto(String dni, String nombre, String direccion, String telefono, double salario, TipoAvion tipoAvionAutorizado) {

super(dni, nombre, direccion, telefono);

this.salario = salario;

this.tipoAvionAutorizado = tipoAvionAutorizado;

}

public double getSalario() {

return salario;

}

public void setSalario(double salario) {

this.salario = salario;

}

public TipoAvion getTipoAvionAutorizado() {

return tipoAvionAutorizado;

}

public void setTipoAvionAutorizado(TipoAvion tipoAvionAutorizado) {

this.tipoAvionAutorizado = tipoAvionAutorizado;

}

}

Ejemplo de uso

java

Copiar código

import java.util.Date;

public class Main {

public static void main(String[] args) {

// Crear tipo de avión

TipoAvion tipoAvion = new TipoAvion("Boeing 747", 660, 183500);

// Crear hangar

Hangar hangar = new Hangar(1, 50, "Sector A");

// Crear propietario (persona)

Persona propietario = new Persona("12345678A", "Juan Pérez", "Calle Falsa 123", "555-1234");

// Crear avión

Avion avion = new Avion(1, "XYZ123", hangar, new Date(), propietario);

// Crear mecánico

Mecanico mecanico = new Mecanico("87654321B", "Carlos López", "Avenida Siempre Viva 456", "555-5678", 35000, "Nocturno", tipoAvion);

// Crear piloto

Piloto piloto = new Piloto("11223344C", "Ana Martínez", "Plaza Mayor 789", "555-9876", 45000, tipoAvion);

// Crear servicio

Servicio servicio = new Servicio(new Date(), "Carga de combustible");

// Mostrar información

System.out.println("Avión ID: " + avion.getId());

System.out.println("Matrícula: " + avion.getNumeroMatricula());

System.out.println("Propietario: " + avion.getPropietario().getNombre());

System.out.println("Mecánico: " + mecanico.getNombre() + ", Turno: " + mecanico.getTurno());

System.out.println("Piloto: " + piloto.getNombre() + ", Tipo de Avión Autorizado: " + piloto.getTipoAvionAutorizado().getModelo());

System.out.println("Servicio: " + servicio.getTipoDeCarga() + " en la fecha: " + servicio.getFecha());

}

}

**Informe del código : El código esta bien hecho , funciona sin errores y la IA genero un ejemplo donde carga datos la única corrección que le haría es que pondría la clase persona en abstracta ya que después se extiende en piloto, mecánico , etc pero es intrascendente .**