1- Escribe la clase MarcaPagina, que ayuda a llevar el control de la lectura de un libro. Deberá disponer de métodos para incrementar la página leída, para obtener información de la última página que se ha leído y para comenzar desde el principio una nueva lectura del mismo libro.

**package** main;

**import** java.util.Scanner;

**public** **class** Marcapaginas {

**private** **int** paginas;

**private** **int** paginasLeidas;

**public** Marcapaginas(**int** paginas, **int** paginasLeidas) {

**this**.paginas = paginas;

**this**.paginasLeidas = paginasLeidas;

}

**public** **int** getPaginasLeidas() {

**return** paginasLeidas;

}

**public** **void** setPaginasLeidas(**int** paginasLeidas) {

**this**.paginasLeidas = paginasLeidas;

}

**public** **int** getPaginas() {

**return** paginas;

}

**public** **void** setPaginas(**int** paginas) {

**this**.paginas = paginas;

}

**public** **void** incrementarPagina() {

**boolean** paginaLeida = **false**;

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Has leído la página?");

String resp = sc.nextLine();

**if** (resp.equals("si") && paginasLeidas < paginas) {

paginaLeida = **true**;

paginasLeidas++;

System.***out***.println("Incrementando página: " + paginasLeidas);

} **else** **if** (resp.equals("no") && paginasLeidas < paginas) {

System.***out***.println("Vas por la página " + paginasLeidas);

}

}

**public** **void** ultimaPaginaLeida() {

System.***out***.println("La última página leída es la: " + paginasLeidas);

}

**public** **void** comenzarLibro() {

**if** (paginasLeidas >= paginas) {

paginasLeidas = 1;

System.***out***.println("Has llegado al final. Volviendo a empezar... ");

}

}}

**package** main;

**public** **class** Main {

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

Marcapaginas mp = **new** Marcapaginas(10, 9);

// incrementar página

mp.incrementarPagina();

// mostrar última página leída

mp.ultimaPaginaLeida();

// empezar libro de nuevo

mp.comenzarLibro();

}

}

2- Implementa una clase que permita resolver ecuaciones de segundo grado. Los coeficientes pueden indicarse en el constructor y modificarse a posteriori. Es fundamental que la clase disponga de un método que devuelva las distintas soluciones y de un método que nos informe si el discriminante es positivo.

**package** main;

**public** **class** ResolvedorEcuacion {

**private** **int** a;

**private** **int** b;

**private** **int** c;

**public** ResolvedorEcuacion(**int** a, **int** b, **int** c) {

**this**.a = a;

**this**.b = b;

**this**.c = c;

}

**public** **int** getA() {

**return** a;

}

**public** **void** setA(**int** a) {

**this**.a = a;

}

**public** **int** getB() {

**return** b;

}

**public** **void** setB(**int** b) {

**this**.b = b;

}

**public** **int** getC() {

**return** c;

}

**public** **void** setC(**int** c) {

**this**.c = c;

}

**public** **void** solucion() {

**int** discriminante = b \* b - (4 \* a \* c);

**if** (discriminante > 0) {

// Dos soluciones reales

**double** x1 = (-b + Math.*sqrt*(discriminante)) / (2 \* a);

**double** x2 = (-b - Math.*sqrt*(discriminante)) / (2 \* a);

System.***out***.println("Las soluciones son reales y diferentes:");

System.***out***.println("x1 = " + x1);

System.***out***.println("x2 = " + x2);

} **else** **if** (discriminante == 0) {

// Una solución real

**double** x = -b / (2 \* a);

System.***out***.println("La solución es real y única:");

System.***out***.println("x = " + x);

} **else** {

System.***out***.println("No hay soluciones reales");

}

}

**public** **void** discPos() {

**int** discriminante = b \* b - (4 \* a \* c);

**if** (discriminante > 0) {

System.***out***.println("El discriminante es positivo");

} **else** {

System.***out***.println("El discriminante es negativo");

}

}

}

**package** main;

**public** **class** Main {

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

ResolvedorEcuacion re = **new** ResolvedorEcuacion(1, -5, 6);

re.solucion();

re.discPos();

}

}

3- Diseña la clase Colores, que alberga por defecto una serie de colores (mediante una cadena), aunque es posible añadir tantos como necesitemos. La clase tendrá un método que devuelve una tabla con los n colores que necesitemos elegidos al azar sin repeticiones.

**package** main;

**import** java.util.Arrays;

**import** java.util.Random;

**import** java.util.Scanner;

**public** **class** Colores {

**private** String serieColores = "Rojo, Verde, Azul, Naranja, Rosa, Amarillo, Blanco, Negro, Gris, Morado, Marrón";

**public** Colores(String serieColores) {

**this**.serieColores = serieColores;

}

**public** String getSerieColores() {

**return** serieColores;

}

**public** String[] tablaColores(**int** n) { // n guarda número de colores que queremos

Random rand = **new** Random();

String [] arrColores = serieColores.split(","); // separamos el string por la coma

String [] resultado = **new** String [n]; // guardará los colores no repes

**int** lonColores = arrColores.length;

**if** (n > lonColores) { // comprobamos que n no excede la longitud del array

n = lonColores;

}

**for** (**int** i = 0; i < n; i++) {

**int** indice = rand.nextInt(lonColores); // escoger un índice aleatorio de la longitud del array

resultado[i] = arrColores[indice]; // a cada posición del nuevo array le pasamos un color al azar del array original

}

**return** resultado;

}

}

**package** main;

**public** **class** Main {

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

Colores col = **new** Colores("Rojo, Verde, Azul, Naranja, Rosa, Amarillo, Blanco, Negro, Gris, Morado, Marrón");

String[] tColores = col.tablaColores(4);

**for** (String color: tColores) {

System.***out***.println("Colores seleccionados: " + color);

}

}

}

4- Diseña la clase Calendario que representa una fecha concreta (año, mes y dia). La clase debe disponer de los métodos: • Calendario(int año, int mes, int dial: que crea un objeto con los datos pasados como parámetros, siempre y cuando, la fecha que representensea correcta. • void incrementarDia(): que incrementa en un día la fecha del calendario. • void incrementarMes(): que incrementa en un mes la fecha del calendario. • void incrementarAño(int cantidad): que incrementa la fecha del calendario en el número de años especificados. Ten en cuenta que el año 0 no existió. • void mostrar(): muestra la fecha por consola. • boolean iguales(Calendario otraFecha): que determina si la fecha invocante y la que se pasa como parámetro son iguales o distintas. Por simplicidad, solo tendremos en consideración que existen meses con distinto número de días, pero no tendremos en cuenta los años bisiestos.

**package** main;

**public** **class** Calendario {

**private** **int** dia;

**private** **int** mes;

**private** **int** ano;

**public** Calendario(**int** dia, **int** mes, **int** ano) {

**this**.dia = dia;

**this**.mes = mes;

**this**.ano = ano;

}

**public** **int** getDia() {

**return** dia;

}

**public** **void** setDia(**int** dia) {

**this**.dia = dia;

}

**public** **int** getMes() {

**return** mes;

}

**public** **void** setMes(**int** mes) {

**this**.mes = mes;

}

**public** **int** getAno() {

**return** ano;

}

**public** **void** setAno(**int** ano) {

**this**.ano = ano;

}

**public** **void** incrementarDia() {

dia++;

System.***out***.println("El día siguiente es: " + dia);

}

**public** **void** incrementarMes() {

**int** diasEnMes;

**if** (mes == 1 || mes == 3 || mes == 5 || mes == 7 || mes == 8 || mes == 10 || mes == 12) {

diasEnMes = 31;

} **else** **if** (mes == 4 || mes == 6 || mes == 9 || mes == 11) {

diasEnMes = 30;

} **else** {

diasEnMes = 28;

}

**if** (dia > diasEnMes) {

mes++;

dia = 1;

}

**if** (mes == 12) {

mes = 1;

}

System.***out***.println("El mes siguiente es: " + mes);

}

**public** **void** incrementarAno(**int** cantidad) {

**int** anoIncr = ano + cantidad;

**if** (ano == 0) {

System.***out***.println("El año 0 no existe");

} **else** {

anoIncr = ano + cantidad;

}

System.***out***.println("Si al año " + ano + " le sumas " + cantidad + " queda " + anoIncr);

}

**public** **void** mostrar() {

System.***out***.println("La fecha introducida es: " + **this**.dia + " / " + **this**.mes + " / " + **this**.ano);

}

**public** **boolean** iguales(Calendario otraFecha) { // pasa por parámetro otro objeto con otraFecha como parámetro

**if** (**this**.dia == otraFecha.dia && **this**.mes == otraFecha.mes && **this**.ano == otraFecha.ano) { // compara esta instancia a la propiedad correspondiente del otro objeto

System.***out***.println("Las fechas son iguales");

**return** **true**;

} **else** {

System.***out***.println("Las fechas son distintas");

**return** **false**;

}

}

}

**package** main;

**public** **class** Main {

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

Calendario cal = **new** Calendario(19, 11, 2024);

cal.incrementarDia();

cal.incrementarMes();

cal.incrementarAno(15);

cal.mostrar();

Calendario fechaComparar = **new** Calendario(20, 11, 2024); // creamos otro objeto para poder llamar al método iguales, que recibe un objeto como parámetro

cal.iguales(fechaComparar); // llamamos el método iguales sobre el primer objeto y le pasamos el nuevo objeto para compararlos

}

}

5- Escribe la clase Punto que representa un punto en el plano (con un componente x y un componente y), con los métodos: • Punto(double x, double y): construye un objeto con los datos pasados como parámetros. • void desplazax(double dx): incrementa el componente x en la cantidad dx. • void desplazaY(double dy): incrementa el componente y en la cantidad dy. • void desplaza(double dx, double dy): desplaza ambos componentes según las cantidades dx (en el eje x) y dy (en el componente y). • doubledistanciaEuclidea(Punto otro): calcula y devuelve la distancia euclídea entre el punto invocante y el punto otro. • void muestra (): muestra por consola la información relativa al punto.

**package** main;

**public** **class** Main {

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

Punto pun = **new** Punto (5, 7);

Punto p2 = **new** Punto (4, 10);

pun.desplazaX(5);

pun.desplazaY(7);

pun.desplaza(5, 7);

pun.distanciaEuclidea(p2);

pun.muestra();

}

}

**package** main;

**public** **class** Punto {

**double** x;

**double** y;

**public** Punto (**double** x, **double** y) {

**this**.x = x;

**this**.y = y;

}

**public** **double** getX() {

**return** x;

}

**public** **void** setX(**double** x) {

**this**.x = x;

}

**public** **double** getY() {

**return** y;

}

**public** **void** setY(**double** y) {

**this**.y = y;

}

**public** **void** desplazaX (**double** dx) {

x += dx;

System.***out***.println("Incrementando X: " + x);

}

**public** **void** desplazaY (**double** dy) {

y += dy;

System.***out***.println("Incrementando Y: " + y);

}

**public** **void** desplaza (**double** dx, **double** dy) {

x += dx;

y += dy;

System.***out***.println("Incrementando X e Y: " + x);

System.***out***.println("Incrementando X e Y: " + y);

}

**public** **double** distanciaEuclidea (Punto otro) {

**double** x1 = **this**.getX();

**double** y1 = **this**.getY();

**double** x2 = otro.getX();

**double** y2 = otro.getY();

**double** distanciaXY = Math.*sqrt*(Math.*pow*(x2 - x1, 2) + Math.*pow*(y2 - y1, 2));

**return** distanciaXY;

}

**public** **void** muestra () {

System.***out***.println("Mostrando valor de x " + **this**.getX());

System.***out***.println("Mostrando valor de y " + **this**.getY());

}

}

6- Realizar un programa en Java que permita calcular el pedido que realiza un cliente en un restaurante. Los pedidos de un restaurante están conformados por las siguientes partes: • Un primer plato. • Un segundo plato. • Una bebida. • Un postre. Cada uno de dichas partes tiene un nombre y un valor. Se requiere definir métodos sobrecargados para calcular el valor del pedido dependiendo si el cliente solicita: • Un primer plato y una bebida. • Un primer plato, un segundo plato y una bebida. • Un primer plato, un segundo plato, una bebida y un postre. Implementar un método main que utilice los tres métodos sobrecargados en tres diferentes pedidos.

**package** main;

**public** **class** CalculoPedido {

**private** String primero;

**private** String segundo;

**private** String bebida;

**private** String postre;

**public** CalculoPedido(String primero, String segundo, String bebida, String postre) {

**this**.primero = primero;

**this**.segundo = segundo;

**this**.bebida = bebida;

**this**.postre = postre;

}

**public** String getPrimero() {

**return** primero;

}

**public** **void** setPrimero(String primero) {

**this**.primero = primero;

}

**public** String getSegundo() {

**return** segundo;

}

**public** **void** setSegundo(String segundo) {

**this**.segundo = segundo;

}

**public** String getBebida() {

**return** bebida;

}

**public** **void** setBebida(String bebida) {

**this**.bebida = bebida;

}

**public** String getPostre() {

**return** postre;

}

**public** **void** setPostre(String postre) {

**this**.postre = postre;

}

**public** **void** precio(String primero, String bebida) {

**int** precioBebida = 3;

**int** precioPrimero = 10;

**int** precio = precioBebida + precioPrimero;

System.***out***.println("La bebida y el primer plato cuestan: " + precio);

}

**public** **void** precio(String primero, String segundo, String bebida) {

**int** precioBebida = 3;

**int** precioPrimero = 10;

**int** precioSegundo = 10;

**int** precio = precioBebida + precioPrimero + precioSegundo;

System.***out***.println("La bebida, el primero y el segundo cuestan: " + precio);

}

**public** **void** precio(String primero, String segundo, String bebida, String postre) {

**int** precioBebida = 3;

**int** precioPrimero = 10;

**int** precioSegundo = 10;

**int** precioPostre = 5;

**int** precio = precioBebida + precioPrimero + precioSegundo + precioPostre;

System.***out***.println("El menú completo cuesta : "+ precio);

}

}