0, eje Loco Pilar

**public** **class** Coche {

**private** String ciudad; //Ciudad de cada objeto taxi

**private** String matricula; //Matrícula de cada objeto taxi

**private** String provincia; //Provincia asignado a cada objeto taxi

**private** **int** tipoMotor; //Tipo de motor asignado a cada objeto taxi. 0 = desconocido, 1 = gasolina, 2 = diesel

//Constructor: cuando se cree un objeto taxi se ejecutará el código que incluyamos en el constructor

**public** Coche(String ciudad, String matricula, String provincia,

**int** tipoMotor) {

**this**.ciudad = ciudad;

**this**.matricula = matricula;

**this**.provincia = provincia;

**this**.tipoMotor = tipoMotor;

}

**public** Coche(String ciudad) {

**this**.ciudad = ciudad;

}

**public** Coche (String valorMatricula, String valorProvincia, **int** valorTipoMotor) {

ciudad = "Madrid";

matricula = valorMatricula;

provincia = valorProvincia;

tipoMotor = valorTipoMotor;

} //Cierre del constructor

**public** String getCiudad () {

**return** ciudad;

}

//Método para obtener la matrícula del objeto coche

**public** String getMatricula () {

**return** matricula;

} //Cierre del método

//Método para obtener la provincia del objeto coche

**public** String getProvincia () {

**return** provincia;

} //Cierre del método

//Método para obtener el tipo de motor del objeto coche

**public** **int** getTipoMotor () {

**return** tipoMotor;

} //Cierre del método

}

-------------------------------------------main--------------------------------------

**public** **class** Taxi {

**private** Coche taxi1;//declaración global del objeto taxi1 de la clase Coche

**private** Coche taxi2; ;//declaración global del objeto taxi2 de la clase Coche

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

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

Taxi obj=**new** Taxi();// para no declarar los metodos y variables globales static

obj.cargarDatos();

obj.muestraDatos();

}

**private** **void** cargarDatos(){

System.***out***.print("\nIntroduce un ciudad: ");

String ciudad =teclado.next();

System.***out***.print("\nIntroduce los matricula: ");

String matricula=teclado.next();

System.***out***.print("\nIntroduce provincia");

String provincia=teclado.next();

System.***out***.print("\n Introduce tipo de motor");

**int** tipoMotor=teclado.nextInt();

taxi1=**new** Coche(ciudad,matricula,provincia, tipoMotor);

taxi2=**new** Coche(matricula,provincia, tipoMotor);

}

**public** **void** muestraDatos(){

System.***out***.print("Los datos de taxi 1 son : ");

System.***out***.println("\nla Ciudad : "+taxi1.getCiudad()+".");

System.***out***.println("la matricula :"+taxi1.getMatricula()+".");

System.***out***.println("la provincia : "+taxi1.getProvincia()+".");

System.***out***.println("El tipo de motor : "+taxi1.getTipoMotor()+".");

System.***out***.print("Los datos de taxi 2 son : ");

System.***out***.println("\nla Ciudad : "+taxi2.getCiudad()+".");

System.***out***.println("la matricula :"+taxi2.getMatricula()+".");

System.***out***.println("la provincia : "+taxi2.getProvincia()+".");

System.***out***.println("El tipo de motor : "+taxi2.getTipoMotor()+".");

}

}

1. Crear una aplicación para simular un cajero automático.

Tenemos la **clase CuentaCorriente**, con los atributos: cliente y cantidad.

* + - Crea dos constructores: uno inicializando el cliente y cantidad por parametros. El otro constructor solo introduciremos el nombre cliente y su saldo será cantidad= 0.

Tenemos tres métodos:

* + - Ingresar(), introduciremos la cantidad a ingresar como parámetro y se lo sumaremos al saldo anterior
    - Retirar(), introduciremos la cantidad a retirar y decrementaremos su saldo.
    - Gettes y Settes.

Tenemos la **clase Sucursal.** Tenemos un ArrayList de tipo Cuenta (clase anterior).

* + - Tenemos un constructor donde introduciremos tres clientes.

Tenemos los siguientes métodos:

* + - Operar() Nos preguntará si quiero consultar un cliente
    - obtenerCliente(), comprobamos si el cliente existe, si es así llamará al método menú()
    - menú()
      * 1.-Ingresar
      * 2.-Retirar
      * 3.-Mostrar Saldo
      * 4.-Salir

Para el anterior menú tendremos los siguientes métodos.

* + - ingresar()
    - extraer()
    - mostrarSaldo()

Una vez introducido el nombre del cliente (que en este caso identifica cada cuenta) se dará acceso a un menú para poder realizar las operaciones básicas sobre la cuenta corriente.

**public** **class** CuentaCorriente {

**private** String cliente;

**private** **float** cantidad;

CuentaCorriente(String cliente){

**this**.cliente=cliente;

**this**.cantidad=0;

}

CuentaCorriente(String cliente,**float** cantidad){

**this**.cliente=cliente;

**this**.cantidad=cantidad;

}

**public** String retornarCliente(){

**return** cliente;

}

**public** **void** extraer(**float** cantidad){

**this**.cantidad=cantidad-cantidad;

}

**public** **void** ingresar(**float** cantidad){

**this**.cantidad=cantidad+cantidad;

}

**public** **float** retornarCantidad(){

**return** cantidad;

}

}

**public** **class** Sucursal {

**private** ArrayList<CuentaCorriente> cuentas = **new** ArrayList<CuentaCorriente>();

Scanner teclado;

**private** String nombre;

**private** **int** numCuenta;

Sucursal() {

//CuentaCorriente cliente1=new CuentaCorriente(“Sara”);

//CuentaCorriente cliente2=new CuentaCorriente("Andrea", 100);

//CuentaCorriente cliente3=new CuentaCorriente(“Carlos”);

//cuentas.add(cliente1);

//cuentas.add(cliente2);

//cuentas.add(cliente3);

//Se trabaja con el arrayList

cuentas.add( **new** CuentaCorriente("Sara"));

cuentas.add( **new** CuentaCorriente("Andrea", 100));

cuentas.add( **new** CuentaCorriente("Carlos"));

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

numCuenta = -1;

System.***out***.println("\*\*\*\*\*\*\*\*\*\*BIENVENIDO AL BANCO\*\*\*\*\*\*\*\*\*\*\n");

}

**void** operar() {

System.***out***.println("Quieres consultar un nuevo clientes");

String opc=teclado.next();

**if** (opc.equalsIgnoreCase("si")){

System.***out***.println("\n\*\*\*\*\*\*\*\*\*\*BIENVENIDO AL BANCO\*\*\*\*\*\*\*\*\*\*\n");

teclado.nextLine();//vaciado buffer

obtenerCliente();

}

**else**{

System.***out***.println("Adios, ten un buen día");

System.*exit*(0);

}

}

**private** **void** obtenerCliente() {

System.***out***.print("\nNOMBRE CLIENTE: ");

nombre = teclado.nextLine();

**for** (**int** i = 0; i < cuentas.size(); i++) {

**if** (nombre.equals(cuentas.get(i).retornarCliente())) {//metodo clase anterior

numCuenta = i;//igualas a la posicion

menu();

**break**;

}

}

**if** (numCuenta == -1) { //la posicion es -1 por no existe

System.***out***.print("\nEl cliente no existe.\n");

operar();

}

}

**private** **void** menu() {

System.***out***.println("\n\*\*\*\*\*\*\*\*\*MENÚ\*\*\*\*\*\*\*\*\*");

System.***out***.println("1> ingresar");

System.***out***.println("2> extraer");

System.***out***.println("3> mostrar saldo");

System.***out***.println("4> salir");

System.***out***.print("Introduzca opción: ");

**int** opcion=teclado.nextInt();

teclado.nextLine();

**switch** (opcion) {

**case** 1:

ingresarCantidad();

**break**;

**case** 2:

extraerCantidad();

**break**;

**case** 3:

mostrarSaldo();

**break**;

**case** 4:

System.***out***.println("\n\*\*\*\*\*GRACIAS POR SU VISITA " +

nombre + "\*\*\*\*\*\*\n");

}

numCuenta = -1;

operar();

}

**private** **void** ingresarCantidad() {

System.***out***.print("CANTIDAD A INGRESAR: ");

**float** cantidad = teclado.nextFloat();

teclado.nextLine();

cuentas.get(numCuenta).ingresar(cantidad);//accedo al AL

System.***out***.println("\nOPERACIÓN REALIZADA CON ÉXITO\n");

System.***out***.println("\nPulse intro para continuar...");

teclado.nextLine();

menu();

}

**private** **void** extraerCantidad() {

System.***out***.print("CANTIDAD A RETIRAR: ");

**float** cantidad = teclado.nextFloat();

teclado.nextLine();

cuentas.get(numCuenta).extraer(cantidad);

System.***out***.println("\nOPERACIÓN REALIZADA CON ÉXITO");

System.***out***.println("\nPulse intro para continuar...");

teclado.nextLine();

menu();

}

**private** **void** mostrarSaldo() {

System.***out***.println("\nSALDO: " +

cuentas.get(numCuenta).retornarCantidad()+ " €");

System.***out***.println("\nPulse intro para continuar...");

teclado.nextLine();

menu();

}

---------------------main-------------------------------------

**public** **class** SucursalPrincipal {

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

Sucursal miSucursal = **new** Sucursal();

miSucursal.operar();

}

}

1. Crea una clase llamada **Biblioteca**, que guarde la información de cada uno de los libros de una biblioteca. La clase debe guardar el título del libro, autor, número de ejemplares del libro y número de ejemplares prestados. La clase contendrá los siguientes métodos:
   * Constructor por defecto.
   * Constructor con parámetros, todos los posibles constructores.
   * Métodos Setters/getters

* + Método *préstamo* que incremente el atributo correspondiente cada vez que se realice un préstamo del libro. No se podrán prestar libros de los que no queden ejemplares disponibles para prestar. Devuelve true si se ha podido realizar la operación y false en caso contrario.

* + Método *devolución* que decremente el atributo correspondiente cuando se produzca la devolución de un libro. No se podrán devolver libros que no se hayan prestado. Devuelve true si se ha podido realizar la operación y false en caso contrario.
  + Método *Mostrar,* para mostrar los datos de los libros.

Crear otra clase llamada **Principal,** donde crearemos cuatro objetos Biblioteca.

Para crear los dos primeros objetos, pediremos por teclado los datos y utilizaremos los métodos setters.

Para crear los dos restantes objetos, utilizaremos constructores.

Visualizar los datos de los libros con getters.

Realizar préstamos y devoluciones, hasta que no queramos realizar más acciones, visualizar como están los objeros al final.

**public** **class** Libro {

**private** String titulo;

**private** String autor;

**private** **int** ejemplares;

**private** **int** prestados;

//constructor por defecto

**public** Libro() {

}

//constructor con parámetros

**public** Libro(String titulo, String autor, **int** ejemplares, **int** prestados) {

**this**.titulo = titulo;

**this**.autor = autor;

**this**.ejemplares = ejemplares;

**this**.prestados = prestados;

}

//faltan constructores

//getters y setters

**public** String getAutor() {

**return** autor;

}

**public** **void** setAutor(String autor) {

**this**.autor = autor;

}

**public** **int** getEjemplares() {

**return** ejemplares;

}

**public** **void** setEjemplares(**int** ejemplares) {

**this**.ejemplares = ejemplares;

}

**public** **int** getPrestados() {

**return** prestados;

}

**public** **void** setPrestados(**int** prestados) {

**this**.prestados = prestados;

}

**public** String getTitulo() {

**return** titulo;

}

**public** **void** setTitulo(String titulo) {

**this**.titulo = titulo;

}

//método para realizar el prestamo de un libro

**public** **boolean** prestamo() {

**if** (prestados < ejemplares) {

prestados++; //pasa saber del contador

**return** **true**;

} **else** {

**return** **false**;

}

}

//método para realizar la devolución de un libro

**public** **boolean** devolucion() {

**if** (prestados == 0) {

**return** **false**;

} **else** {

prestados--;

**return** **true**;

}

}

//método toString sobrescrito para mostrar los datos de la clase Libro

//se puede utilizar metodos get

**public** String toString() {

**return** "titulo: " + titulo + "\nautor: " + autor +

"\nejemplares: " + ejemplares + "\nprestados: " + prestados;

}

**public** Libro(String titulo, String autor, **int** ejemplares) {

**super**();

**this**.titulo = titulo;

**this**.autor = autor;

**this**.ejemplares = ejemplares;

}

}

--------------------------------------main-----------------------------------------

**public** **class** Principal {

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

**try** (Scanner sc = **new** Scanner(System.***in***)) {

String titulo, autor;

**int** ejemplares;

//se crea el objeto libro1 utilizando el constructor con parámetros

Libro libro1 = **new** Libro("El quijote", "Cervantes", 1, 0);

//se crea el objeto libro2 utilizando el constructor por defecto

Libro libro3 = **new** Libro("El quijote", "Julio", 3);

//introducir por teclado los set, faltan dos objetos

System.***out***.print("Introduce titulo: ");

titulo = sc.nextLine();

System.***out***.print("Introduce autor: ");

autor = sc.nextLine();

System.***out***.print("Numero de ejemplares: ");

ejemplares = sc.nextInt();

//se asigna a libro2 los datos pedidos por teclado.

//para ello se utilizan los métodos setters

Libro libro2 = **new** Libro();

libro2.setTitulo(titulo);

libro2.setAutor(autor);

libro2.setEjemplares(ejemplares);

//se muestran por pantalla los datos del objeto libro1

//se utilizan los métodos getters para acceder al valor de los atributos

//ejemplo de se ha prestado el libro

System.***out***.println("Libro 1:");

System.***out***.println("Titulo: " + libro1.getTitulo());

System.***out***.println("Autor: " + libro1.getAutor());

System.***out***.println("Ejemplares: " + libro1.getEjemplares());

System.***out***.println("Prestados: " + libro1.getPrestados());

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

//se realiza un préstamo de libro1. El método devuelve true si se ha podido

//realizar el préstamo y false en caso contrario

**if** (libro1.prestamo()) {

System.***out***.println("Se ha prestado el libro " + libro1.getTitulo());

} **else** {

System.***out***.println("No quedan ejemplares del libro " + libro1.getTitulo() + " para prestar");

}

//se realiza una devolución de libro1. El método devuelve true si se ha podido

//realizar la devolución y false en caso contrario

**if** (libro1.devolucion()) {

System.***out***.println("Se ha devuelto el libro " + libro1.getTitulo());

} **else** {

System.***out***.println("No hay ejemplares del libro " + libro1.getTitulo() + " prestados");

}

//se realiza otro préstamo de libro1

**if** (libro1.prestamo()) {

System.***out***.println("Se ha prestado el libro " + libro1.getTitulo());

} **else** {

System.***out***.println("No quedan ejemplares del libro " + libro1.getTitulo() + " para prestar");

}

//se realiza otro préstamo de libro1. En este caso no se podrá realizar ya que

//solo hay un ejemplar de este libro y ya está prestado. Se mostrará por

//pantalla el mensaje No quedan ejemplares del libro…

**if** (libro1.prestamo()) {

System.***out***.println("Se ha prestado el libro " + libro1.getTitulo());

} **else** {

System.***out***.println("No quedan ejemplares del libro " + libro1.getTitulo() + " para prestar");

}

//mostrar los datos del objeto libro1

System.***out***.println("Libro 1:");

System.***out***.println("Titulo: " + libro1.getTitulo());

System.***out***.println("Autor: " + libro1.getAutor());

System.***out***.println("Ejemplares: " + libro1.getEjemplares());

System.***out***.println("Prestados: " + libro1.getPrestados());

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

//mostrar los datos del objeto libro2

System.***out***.println("Libro 2:");

System.***out***.println("Titulo: " + libro2.getTitulo());

System.***out***.println("Autor: " + libro2.getAutor());

System.***out***.println("Ejemplares: " + libro2.getEjemplares());

System.***out***.println("Prestados: " + libro2.getPrestados());

}

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

}

}

1. Escribe una clase Cuenta para representar una cuenta bancaria. Los datos de la cuenta son: nombre del cliente (String), número de cuenta (String), tipo de interés (double) y saldo (double).

La clase contendrá los siguientes métodos:

* + - Constructor por defecto
    - Constructor con todos los parámetros
    - Métodos setters/getters para asignar y obtener los datos de la cuenta.
    - Métodos *ingreso* . Un ingreso consiste en aumentar el saldo en la cantidad que se indique. Esa cantidad no puede ser negativa.
    - Métodos *reintegro.* Un reintegro consiste en disminuir el saldo en una cantidad pero antes se debe comprobar que hay saldo suficiente. La cantidad no puede ser negativa.
    - Los métodos ingreso y reintegro devuelven true si la operación se ha podido realizar o false en caso contrario.
    - Método *transferencia* que permita pasar dinero de una cuenta a otra siempre que en la cuenta de origen haya dinero suficiente para poder hacerla. Ejemplo de uso del método transferencia:
    - cuentaOrigen.transferencia(cuentaDestino, importe);

Indica que queremos hacer una transferencia desde cuentaOrigen a cuentaDestino del importe indicado.

**public** **class** Cuenta {

**private** String nombre;

**private** String numeroCuenta;

**private** **double** tipoInteres;

**private** **double** saldo;

//Constructor por defecto

**public** Cuenta() {

}

//Constructor con parámetros

**public** Cuenta(String nombre, String numeroCuenta, **double** tipoInteres, **double** saldo) {

**this**.nombre = nombre;

**this**.numeroCuenta = numeroCuenta;

**this**.tipoInteres = tipoInteres;

**this**.saldo = saldo;

}

//Constructor copia

**public** Cuenta(**final** Cuenta c) {

nombre = c.nombre;

numeroCuenta = c.numeroCuenta;

tipoInteres = c.tipoInteres;

saldo = c.saldo;

}

//getters y setters

**public** **void** setNombre(String s) {

nombre = s;

}

**public** **void** setNumeroCuenta(String s) {

numeroCuenta = s;

}

**public** **void** setTipoInteres(**double** n) {

tipoInteres = n;

}

**public** **void** setSaldo(**double** n) {

saldo = n;

}

**public** String getNombre() {

**return** nombre;

}

**public** String getNumeroCuenta() {

**return** numeroCuenta;

}

**public** **double** getTipoInteres() {

**return** tipoInteres;

}

**public** **double** getSaldo() {

**return** saldo;

}

//método ingreso

**public** **boolean** ingreso(**double** n) {

**boolean** ingresoCorrecto = **true**;

**if** (n < 0) {

ingresoCorrecto = **false**;

} **else** {

saldo = saldo + n;

}

**return** ingresoCorrecto;

}

//método reintegro

**public** **boolean** reintegro(**double** n) {

**boolean** reintegroCorrecto = **true**;

**if** (n < 0) {

reintegroCorrecto = **false**;

} **else** **if** (saldo >= n) {

saldo -= n;

} **else** {

reintegroCorrecto = **false**;

}

**return** reintegroCorrecto;

}

//método transferencia

**public** **boolean** transferencia(Cuenta c, **double** n) {

**boolean** correcto = **true**;

**if** (n < 0) {

correcto = **false**;

} **else** **if** (saldo >= n) {

reintegro(n);

c.ingreso(n);

} **else** {

correcto = **false**;

}

**return** correcto;

}

}

------------------------------------main-------------------------------

**public** **class** CuentaPrincipal {

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

**try** (Scanner sc = **new** Scanner(System.***in***)) {

String nombre, numero;

**double** tipo, importe;

//se crea objeto cuenta1 sin parámetros

//se ejecuta el constructor por defecto

Cuenta cuenta1 = **new** Cuenta();

System.***out***.print("Nombre : ");

nombre = sc.nextLine();

System.***out***.print("Número de cuenta : ");

numero = sc.nextLine();

System.***out***.print("Tipo de interes : ");

tipo = sc.nextDouble();

System.***out***.print("Saldo: ");

importe = sc.nextDouble();

cuenta1.setNombre(nombre);

cuenta1.setNumeroCuenta(numero);

cuenta1.setTipoInteres(tipo);

cuenta1.setSaldo(importe);

//se crea el objeto cuenta2 con los valores leidos por teclado

//se ejecuta el constructor con parámetros

Cuenta cuenta2 = **new** Cuenta("Juan Ferrández Rubio", "12345678901234567890", 1.75, 300);

//se crea cuenta3 como copia de cuenta1

//se ejecuta el constructor copia

Cuenta cuenta3 = **new** Cuenta(cuenta1);

//mostrar los datos de cuenta1

System.***out***.println("Datos de la cuenta 1");

System.***out***.println("Nombre del titular: " + cuenta1.getNombre());

System.***out***.println("Número de cuenta: " + cuenta1.getNumeroCuenta());

System.***out***.println("Tipo de interés: " + cuenta1.getTipoInteres());

System.***out***.println("Saldo: " + cuenta1.getSaldo());

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

//se realiza un ingreso en cuenta1

cuenta1.ingreso(4000);

//mostrar el saldo de cuenta1 después del ingreso

System.***out***.println("Saldo: " + cuenta1.getSaldo());

//mostrar los datos de cuenta2

System.***out***.println("Datos de la cuenta 2");

System.***out***.println("Nombre del titular: " + cuenta2.getNombre());

System.***out***.println("Número de cuenta: " + cuenta2.getNumeroCuenta());

System.***out***.println("Tipo de interés: " + cuenta2.getTipoInteres());

System.***out***.println("Saldo: " + cuenta2.getSaldo());

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

//mostrar los datos de cuenta3

System.***out***.println("Datos de la cuenta 3");

System.***out***.println("Nombre del titular: " + cuenta3.getNombre());

System.***out***.println("Número de cuenta: " + cuenta3.getNumeroCuenta());

System.***out***.println("Tipo de interés: " + cuenta3.getTipoInteres());

System.***out***.println("Saldo: " + cuenta3.getSaldo());

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

//realizar una transferencia de 10€ desde cuenta3 a cuenta2

cuenta3.transferencia(cuenta2, 10);

//mostrar el saldo de cuenta2

System.***out***.println("Saldo de la cuenta 2");

System.***out***.println("Saldo: " + cuenta2.getSaldo());

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

//mostrar el saldo de cuenta3

System.***out***.println("Saldo de la cuenta 3");

System.***out***.println("Saldo: " + cuenta3.getSaldo());

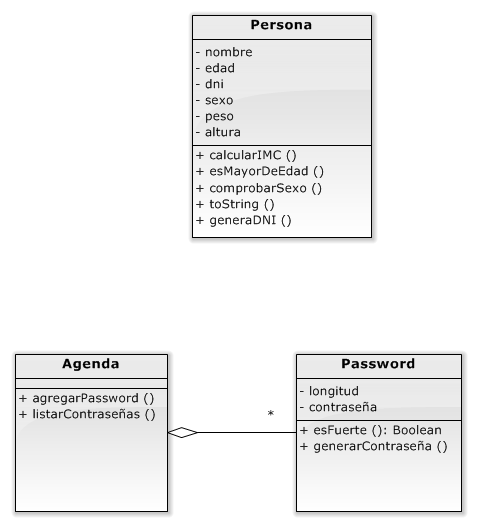
}

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

}

}

1. Haz una clase llamada **Password** que siga las siguientes condiciones:



* Que tenga los atributos**longitud** y **contraseña**.
* Los constructores serán los siguientes:
  + Un constructor por defecto.Por defecto, la longitud será de 8.
  + Un constructor con la longitud que escoja el usuario.Generará una contraseña aleatoria con esa longitud
  + Un constructor con la longitud y la contraseña escogidas por el usuario.
* Los métodos que implementa serán:
  + **esFuerte()**: devuelve un booleano si es fuerte o no, para que sea fuerte debe tener como mínimo 1 1 minúscula y 5 números.
  + **generarPassword()**:  genera la contraseña del objeto con la longitud que tenga.

// Retorna un número aleatorio entre min (incluido) y max (excluido)

Math.random() \* (max - min) + min;

* + Método **get** para contraseña y longitud.
  + Método **set** para longitud.

Ahora, crea una clase **Agenda**:

* Crea un ArrayList de Passwords.
* Crea un bucle que cree un objeto para cada posición del array.
* Indica también por teclado la longitud de los Passwords.
* Crea otro array de booleanos donde se almacene si el password del array de Password es o no fuerte.
* Al final, muestra las contraseñas y si son o no fuertes.

**public** **class** Password {

**private** **int** longitud;

**private** String contraseña;

**public** Password(){

longitud=8;

generarContraseña();

}

**public** Password(**int** longitud){

**this**.longitud=longitud;

generarContraseña();

}

**public** Password(**int** longitud,String contraseña){

**this**.longitud=longitud;

**this**.contraseña=contraseña;

}

**public** **boolean** esFuerte(){

**char** letra;

**int** mayus,minus,nums;

mayus=minus=nums=0;

**for**(**int** i=0;i<contraseña.length();i++){

letra=contraseña.charAt(i);

**if**(letra>='0' && letra<='9') nums++;

**if**(letra>='a' && letra<='z') minus++;

**if**(letra>='A' && letra<='Z') mayus++;

}

**if** (mayus>=2 && minus>=1 && nums>=5) **return** **true**;

**else** **return** **false**;

}

**private** **void** generarContraseña(){

contraseña="";

**for**(**int** i=1;i<=longitud;i++){

**int** opcion=(**int**)((Math.*random*()\*3)+1);

**switch**(opcion){

**case** 1:

contraseña+=(**char**)((**int**)

(Math.*random*()\*(57-48+1)+48));

**break**;

**case** 2:

contraseña+=(**char**)((**int**)

(Math.*random*()\*(90-65+1)+65));

**break**;

**case** 3:

contraseña+=(**char**)((**int**)

(Math.*random*()\*(122-97+1)+97));

**break**;

}

}

}

**public** **void** setLongitud(**int** longitud){

**this**.longitud=longitud;

}

**public** **int** getLongitud(){

**return** longitud;

}

**public** String getContraseña(){

**return** contraseña;

}

}

-----------------main-----------------------------

**public** **class** Agenda {

**private** ArrayList<Password> contraseñas;

**public** Agenda(){

contraseñas=**new** ArrayList();

}

**public** **void** agregarPassword(){

contraseñas.add(**new** Password());

contraseñas.add(**new** Password(10));

contraseñas.add(**new** Password(8,"CON12345"));

contraseñas.add(**new** Password(8,"Con12345"));

}

**public** **void** listarContraseñas(){

**for**(**int** i=0;i<contraseñas.size();i++){

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

System.***out***.println("Longitud: "+contraseñas.get(i).getLongitud());

System.***out***.println("Contraseña: "+contraseñas.get(i).getContraseña());

**if**(contraseñas.get(i).esFuerte())

System.***out***.println("La contraseña es fuerte");

**else** System.***out***.println("La contraseña no es fuerte");

}

}

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

Agenda miAgenda=**new** Agenda() ;

miAgenda.agregarPassword();

miAgenda.listarContraseñas();

}

}