

Examen 2023 2C - Tema 2

Una organización sin fines de lucro desea comenzar a gestionar en un sistema sus donantes y donaciones recibidas. En principio se debe modelar a esta organización con un nombre (la razón social), debe poder registrar donantes y recibir donaciones.

Los donantes se caracterizan por un nombre y apellido, por lo cual no pueden registrarse dos donantes con mismo nombre y apellido (se consideraría que es la misma persona). Al momento de registrar un donante a la organización, se le asigna un número identificador (id) que es único en toda la organización.

Las donaciones se cargan inicialmente en la organización informando quién dona (donante), la fecha y el monto. Similar al caso de donantes, cuando se carga una donación se le asigna un número identificador (id) que es único para cada donación en la organización. Las donaciones tienen un estado asociado que puede ser PENDIENTE, RECHAZADA o COBRADA. Cuando se carga la donación siempre está en estado PENDIENTE.

Se solicita realizar lo siguiente:

- a) Implementar todas las clases necesarias que se describen en el enunciado, incluyendo los métodos necesarios para resolver las funcionalidades solicitadas.

Tips:

- Para definir la fecha de la donación se puede utilizar la clase nativa `LocalDate`
<https://docs.oracle.com/javase/8/docs/api/java/time/LocalDate.html>
- Para definir el estado de la donación se puede utilizar un enum
<https://docs.oracle.com/javase/tutorial/java/javaOO/enum.html>

- b) Implementar la siguiente funcionalidad:

- 1) Crear una **organización vacía**
- 2) **Registrar donantes** en la organización
Dado un nombre y apellido, registrar el donante en la organización, asignándole un nuevo ID, y devolver como retorno el donante registrado. Si el donante ya estaba registrado, se debe devolver a ese donante preexistente.
- 3) **Cargar donaciones** en la organización
Dado un donante previamente registrado, una fecha y un monto, generar una donación en la organización con su estado por defecto (PENDIENTE) y devolver la donación generada.
- 4) **Actualizar donaciones** que fueron generadas
A partir de una donación permitir actualizar su estado a COBRADA o RECHAZADA.
- 5) Mostrar los **donantes registrados**
Mostrar en consola los donantes registrados en la organización. Cada donante debe mostrar su ID, nombre y apellido.
- 6) Mostrar las **donaciones ordenadas por fecha**

Mostrar en consola las donaciones recibidas en la organización ordenadas por fecha de carga. Cada donación debe mostrar su ID, donante, fecha, estado y monto.

7) Mostrar **resultado a la fecha**

Mostrar en consola el estado de resultado de la organización a cierta fecha límite. Esto significa que sólo se utilizan para calcular aquellas **donaciones con fecha de carga menor o igual a la fecha límite**.

La información a mostrar es la siguiente:

- Cantidad de donaciones cobradas
- Cantidad de donaciones rechazadas
- Cantidad de donaciones pendientes

Si existe **al menos una donación cobrada en este resultado**, mostrar:

- Monto total acumulado de donaciones cobradas
- Monto de donación cobrada máximo
- Monto de donación cobrada mínimo
- Monto medio de las donaciones cobradas (total / cobradas)

Forma de Entrega: Archivo zip conteniendo los archivos de código fuente (*.java).

Criterios mínimos de aprobación:

- No deben existir errores de compilación.
- No deben existir errores en tiempo de ejecución (excepciones no tratadas o no deseadas).
- Implementar la estructura básica de la solución contemplando el POO.
- Completar correctamente la funcionalidad requerida de los puntos b.1 a b.6.

Anexo 1: Ejemplo de Main

Se puede utilizar el siguiente método main para probar el ejercicio, donde ya hay un escenario armado para verificar la funcionalidad.

```
public static void main(String[] args) {
    ONG org = new ONG("Codigo Verde");

    // b.2 Registro de donantes
    Donante dt1 = org.registrarDonante("Wiley", "Wesley");
    Donante dt2 = org.registrarDonante("Chandler", "Stephens");
    Donante dt3 = org.registrarDonante("Keeley", "Sherman");
    Donante dt4 = org.registrarDonante("Keiran", "Raines");
    Donante dt5 = org.registrarDonante("Brendon", "Boothman");

    // b.3 Carga de Donaciones
    Donacion dc1 = org.cargarDonacion(dt1, LocalDate.of(2023,10,26), 50);
    Donacion dc2 = org.cargarDonacion(dt2, LocalDate.of(2023,10,22), 100);
    Donacion dc3 = org.cargarDonacion(dt3, LocalDate.of(2023,10,22), 60);
    Donacion dc4 = org.cargarDonacion(dt4, LocalDate.of(2023,10,31), 40);
    Donacion dc5 = org.cargarDonacion(dt5, LocalDate.of(2023,10,15), 30);
    Donacion dc6 = org.cargarDonacion(dt1, LocalDate.of(2023,11,7), 100);
    Donacion dc7 = org.cargarDonacion(dt2, LocalDate.of(2023,11,2), 100);
    Donacion dc8 = org.cargarDonacion(dt3, LocalDate.of(2023,11,9), 60);
    Donacion dc9 = org.cargarDonacion(dt4, LocalDate.of(2023,11,10), 50);
    Donacion dc10 = org.cargarDonacion(dt5, LocalDate.of(2023,11,11), 90);

    // b.4 Actualizacion de Donaciones
    dc1.setCobrada();
    dc2.setCobrada();
    dc3.setCobrada();
    dc4.setRechazada();
    dc5.setCobrada();
    dc6.setCobrada();
    dc7.setRechazada();
    dc9.setCobrada();

    // b.5 Mostrar Donantes
    org.mostrarDonantes();

    // b.6 Mostrar Donaciones ordenadas por fecha
    org.mostrarDonaciones();
}
```

```
// b.7 Mostrar Resultado a la fecha 10/11/2023
org.mostrarResultadoFecha(LocalDate.of(2023,11,10));
}
```

Anexo 2: Ejemplo de salida de consola

Dado el ejemplo del main previo, se podría obtener una salida similar a la siguiente. No es necesario se cumpla con el formato propuesto, simplemente se adjunta para verificar los datos del ejemplo.

b.5 Mostrar Donantes

Listado de donantes deCodigo Verde

- (5) Boothman, Brendon
- (1) Wesley, Wiley
- (3) Sherman, Keeley
- (2) Stephens, Chandler
- (4) Raines, Keiran

b.6 Mostrar Donaciones ordenadas por fecha

Listado de donaciones deCodigo Verde

Donacion 5

- Donante: (5) Boothman, Brendon
- Fecha: 2023-10-15
- Estado: COBRADA
- Monto: 30.0

Donacion 2

- Donante: (2) Stephens, Chandler
- Fecha: 2023-10-22
- Estado: COBRADA
- Monto: 100.0

Donacion 3

- Donante: (3) Sherman, Keeley
- Fecha: 2023-10-22
- Estado: COBRADA
- Monto: 60.0

Donacion 1

- Donante: (1) Wesley, Wiley
- Fecha: 2023-10-26
- Estado: COBRADA
- Monto: 50.0

Donacion 4

- Donante: (4) Raines, Keiran
- Fecha: 2023-10-31
- Estado: RECHAZADA
- Monto: 40.0

Donacion 7

- Donante: (2) Stephens, Chandler
- Fecha: 2023-11-02
- Estado: RECHAZADA
- Monto: 100.0

Donacion 6

- Donante: (1) Wesley, Wiley
- Fecha: 2023-11-07
- Estado: COBRADA
- Monto: 100.0

Donacion 8

- Donante: (3) Sherman, Keeley
- Fecha: 2023-11-09
- Estado: PENDIENTE
- Monto: 60.0

Donacion 9

- Donante: (4) Raines, Keiran
- Fecha: 2023-11-10
- Estado: COBRADA
- Monto: 50.0

Donacion 10

- Donante: (5) Boothman, Brendon
- Fecha: 2023-11-11
- Estado: PENDIENTE
- Monto: 90.0

b.7 Mostrar Resultado a la fecha 10/11/2023

Estado de Resultado deCodigo Verde al 2023-11-10

- Cobradas: 6
- Rechazadas: 2
- Pendientes: 1
- Monto Total: \$ 390.0
- Monto máximo: \$ 100.0
- Monto mínimo: \$ 30.0
- Monto medio: \$ 65.0