






**Descripción:** Descargue el ejercicio *simulador\_bancario* desde la plataforma, en la sección de la semana 4. A partir de este, realice las siguientes tareas:

- 1) Modifique la clase CDT, en el método correspondiente para qué: Si se desea invertir en un nuevo CDT cuando ya hay uno actualmente abierto, tire una excepción con el mensaje “ERROR: SOLO SE PUEDE TENER UN CDT ACTIVO A LA VEZ”
- 2) Cree el método verificarInvariante() de las clases CDT y SimuladorBancario e invóquelo en los métodos que considere pertinentes. Considere las siguientes aserciones y su respectivo mensaje de error:
  - a. Aquellos atributos que deban ser positivos.
  - b. Aquellos objetos que deban existir en todo momento del programa.
  - c. Que el total de dinero a interés (Ahorros + CDT) sea menor a la constante INVERSION\_MÁXIMO
    - i. En caso de que no se cumpla este invariante, retorne el siguiente mensaje de error en el AssertionError: “ERROR: SE SUPERÓ EL MONTO MÁXIMO DE INVERSIÓN”
- 3) Construya los siguientes escenarios y pruebas (en la clase correspondiente) para validar la correcta implementación de los dos puntos anteriores:
  - a. setupEscenario10(): Inicie el escenario2. Abra un nuevo CDT con \$500,000 al 2% de interés mensual. Retire \$200,000 de la cuenta de ahorros
  - b. testPunto1(): Inicie el escenario10. Avance un mes en la simulación. Cierre el CDT. Utilice una aserción para verificar que el saldo total de la simulación sea de \$2,074,000. En caso de no cumplirse la aserción utilice el mensaje “ERROR EN EL PUNTO 1”
  - c. testPunto2a(): Esta prueba ya existe. Ejecútela y corrija los errores en el código que causan las excepciones encontradas. Guíese por los contratos de los métodos afectados.

**PUNTO FINAL (VALE POR RETO 5):** Agregue la siguiente clase al mundo del problema:

Transaccion	
 <code>int</code>	<code>ENTRADA = 0</code>
 <code>int</code>	<code>SALIDA = 1</code>
 <code>int</code>	<code>AHORROS = 0</code>
 <code>int</code>	<code>CORRIENTE = 1</code>
 <code>int</code>	<code>CDT = 2</code>
<code>- consecutivo : int</code> <code>- valor : double</code>	
<code>+ Transaccion(pConsecutivo: int, pValor: double)</code> <code>+ darConsecutivo() : int</code> <code>+ darValor() : double</code> <code>+ darTipoTransaccion(): int</code> <code>+ darTipoCuenta(): int</code>	

Considere que la relación entre el simulador bancario y la nueva clase transacción es de uno a muchos. Es decir, una instancia del simulador bancario puede tener múltiples transacciones. Guarde esta información en un atributo llamado *transacciones*. No olvide crear su respectivo método getter.

Modifique el mundo del problema para que cada vez que se realice un movimiento en los saldos (Consignación, retiro, ingreso de intereses) se cree una nueva transacción y se agregue a la lista de transacciones históricas.

Modifique el método 3 para que retorne el consecutivo asociado a la transacción de mayor valor para un tipo(entrada/salida) y una cuenta(ahorros/corriente/cdt) dados por parámetro.

LOS SIGUIENTES DOS PUNTOS SON OPCIONALES. SI DECIDE REALIZARLOS PUEDE OBTENER UNA BONIFICACIÓN DE HASTA 1.2 UNIDADES SOBRE LA NOTA FINAL DEL CICLO.

**BONO PARTE A (0.6 PUNTOS):** Utilizando TRY-CATCH, modifique el método `invertirCDT` de la clase `SimuladorBancario` para qué, si el string que se recibe es de la forma `##` (`#` representa un número entero) se realice el manejo del error para convertirlo en un doble dónde el número anterior a la coma sea la parte entera y el número posterior a la coma sea la parte decimal. En cualquier otro caso, tire una excepción con el mensaje "ERROR: EL VALOR INGRESADO DEBE SER NUMÉRICO"

**BONO PARTE B (0.6 PUNTOS):** Modifique la interfaz para que en el `PanelTransacciones` muestre el historial de transacciones y al seleccionar una, muestre sus detalles en el `PanelDetalles`.

Replique las acciones asociadas a la última prueba del reto 1 (se muestra a continuación)

cuenta	%	saldo inicial	meses	saldo final	interes	Fuente Avance
Ahorros	0.6%	\$1,000,000.00	4	\$1,024,216.87	\$24,216.87	Método 1 (4)
CDT	1.0%	\$50,000.00	4	\$52,000.00	\$2,000.00	Método 1 (4)
Ahorros	0.6%	\$1,024,216.87	1	\$1,030,362.17	\$6,145.30	AvanzarSimulacion()
Ahorros	0.6%	\$530,362.17	1	\$533,544.34	\$3,182.17	AvanzarSimulacion()
Ahorros	0.6%	\$533,544.34	5	\$549,743.90	\$16,199.56	Método 1 (5)
CDT	5.0%	\$400,000.00	5	\$500,000.00	\$100,000.00	Método 1 (5)

TOTAL	\$151,743.90
-------	--------------

Grabe un video de máximo 10 segundos diciendo su nombre completo en voz alta y mostrando el historial de transacciones asociadas a ese escenario. Finalmente, ejecute el método dos (mediante el botón, desde la aplicación). Suba el video a youtube y envíe el link a través de la plataforma iMaster.