

Objetivos

Que el alumno:

- Valore la reutilización del código, a través de la instanciación de clases **propias** previamente definidas
- Valore la reutilización del código, a través de la instanciación de clases **predefinidas de Java**
- Aprenda a manipular archivos de acceso secuencial y directo - Aprenda a manipular datos de tipo fecha

Atención: en todos los casos, al grabar utilizar los getter's de los atributos de las clases correspondientes.

1. Implementar una clase ejecutable Empresa que solicite los datos de sus empleados, de acuerdo a los atributos de la clase Empleado. El programa debe permitir el ingreso de datos hasta que el operador indique lo contrario. Con los datos ingresados, instanciar cada empleado y mostrarlo en pantalla con el método apropiado.



El programa además debe grabar en un archivo secuencial con el nombre "Empleado.dat" los datos de cada empleado, incluyendo el sueldo neto (utilizar el método definido a tal efecto). Utilizar los métodos apropiados de la clase Calendar para obtener, de la fecha de ingreso de cada empleado, los datos desagregados para grabar en el archivo (Día, Mes, Año). El diseño de registro es el siguiente.

CUIL	Apellido	Nombre	Sueldo básico	Sueldo Neto	Día Ingreso	Mes Ingreso	Año Ingreso
------	----------	--------	---------------	-------------	-------------	-------------	-------------

2. You have to implement an executable class named DatosVip which allows generating a new sequential file, named "EmpleadoVIP.dat". Read the data from the file "Empleado.dat" and record only those employees who had more than 10 years working for the company.



Record them with the same record design. You have to use the method which provides the class Employee.

3. Crear una clase ejecutable para generar el archivo de acceso directo "Laboratorio.dat". Solicitar y grabar los datos con el siguiente diseño de registro:

Código Laboratorio (int)	Nombre (String * 30)	Domicilio (String * 30)	Teléfono (String * 30)	Compra Mínima (int)	Día de Entrega (int)
-----------------------------	-------------------------	----------------------------	---------------------------	------------------------	-------------------------

El programa debe permitir el ingreso de datos hasta que el operador indique lo contrario.

Atencion!! A los efectos de acceder posteriormente en forma directa a los registros, Codigo de Laboratorio debe ser inicializado en 0 (cero) para el primer registro y luego se deberá incrementar para cada nuevo registro.

4. Una fundación desea imprimir las invitaciones del próximo evento que realizará. Los destinatarios de las invitaciones son leídos de un archivo secuencial de nombre "Invitados.dat", cuyo registro responde a los atributos de la clase Persona, y tiene el siguiente diseño:

DNI (int)	Apellido (String)	Nombre (String)	Día Nacimiento (int)	Mes Nacimiento (int)	Año Nacimiento (int)
--------------	----------------------	--------------------	-------------------------	-------------------------	-------------------------

A los invitados de 60 o más años se les realiza un 10% de descuento en la inscripción (utilizar método provisto por la clase para determinar la edad). El formato de la invitación es el siguiente:

Estimado/a: **José Martínez**

La Fundación Educar para Crecer

Invita a Ud. al próximo evento: **Congreso de Cardiopatías Congénitas.**

Las inscripciones se realizan con **10 días de anticipación.**

Las mismas se llevarán a cabo el día: **17/10/2016**

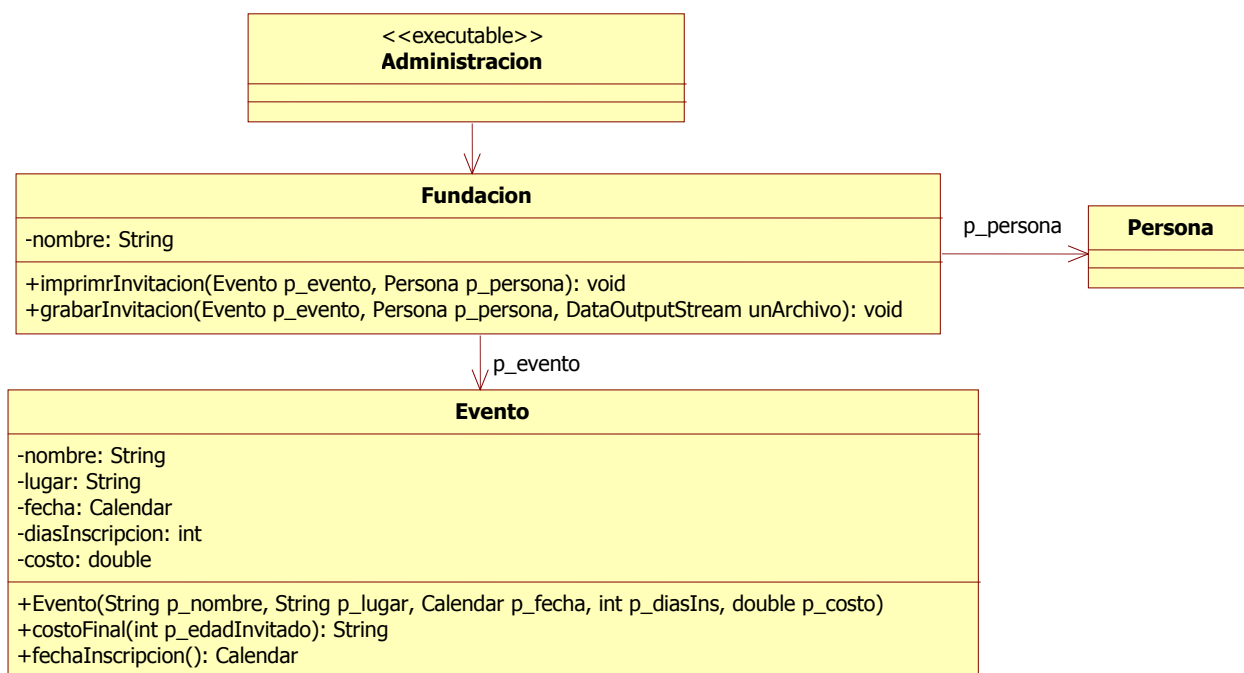
El evento tendrá lugar en: **Facultad Medicina UBA, el día 27 Octubre 2016**

Costo: **\$270 (incluye descuento)** Lo esperamos

Además, se desea grabar los datos en un archivo secuencial con los siguientes datos:

DNI	Apellido y Nombre	Nombre Evento	Lugar	Costo	Mes Evento	Día Evento
(int)	(String)	(String)	(String)	(double)	(int)	(int)

Los requerimientos se modelan en el siguiente diagrama de clases:



El atributo `diasInscripcion` indica cuántos días antes de la fecha del evento se realizará la inscripción.

El método `fechaInscripción()` retorna la fecha correspondiente.

En caso que el invitado se vea beneficiado con un descuento, éste se practicará al costo del evento, y se indicará con el mensaje “incluye descuento”. Esto debe ser implementado en el método `costoFinal()`.

Nota: El archivo “Invitados.dat” que se adjunta en el Aula Virtual contiene 3 registros, de los cuales 2 cumplen la condición para recibir descuento.

6. Una farmacia necesita conocer, al final del día, qué productos comprar y a qué laboratorio hacerlo. Para la compra, establece 2 políticas:
 - a) **Política por Faltante de Stock (FS):** si el stock del producto es igual o menor al mínimo de compra requerido por el laboratorio de dicho producto.
 - b) **Política por Punto de Reposición (PR):** si el stock del producto es menor o igual al punto de reposición del producto y, al día del análisis, faltan más de 5 días (corridos¹) hasta el día de entrega pautado por el laboratorio.

La farmacia posee un archivo de acceso secuencial, donde están registrados todos los productos que maneja la distribuidora, llamado **productos.dat**; y un archivo de acceso directo, con los datos de los Laboratorios, llamado **laboratorio.dat**. (Ver diseño de archivos más abajo).

Al final del día se ejecuta el proceso de solicitudes de compra en el que el sistema verifica la existencia de cada producto y, como resultado de dicho proceso, crea un archivo de acceso secuencial llamado **comprar_AAAAMMDD.txt**, en el que se registran las solicitudes de compras, donde AAAAMMDD es la fecha en la que se ejecuta el proceso.

Diseño de archivos

Solicitudes de compra: (comprar_AAAAMMDD.txt)

¹ Días corridos: considerar los 7 días de la semana. Opuesto: Días laborables (de lunes a viernes).

Código Producto (int)	Código Laboratorio (int)	Cantidad compra (int)	Monto compra (double)	Razón de compra (char)
-----------------------------	--------------------------------	-----------------------------	-----------------------------	------------------------------

Razón de compra: **F** o **P** según se trate de un faltante de stock o de punto de reposición.

Productos: (productos.dat)

Código Producto (int)	Rubro (String *30)	Descripción (String * 30)	Existencia Mínima (int)	% Punto de Reposición (double)	Stock (int)	Precio (double)	Código Laboratorio (int)
--------------------------	-----------------------	------------------------------	----------------------------	--------------------------------------	----------------	--------------------	--------------------------------

Existencia mínima: cantidad mínima del producto que debe existir en el depósito. Valor establecido por la farmacia.

% Punto de Reposición: porcentaje por encima de la existencia mínima que indica que se debe solicitar la compra de dicho producto. Establecido por la farmacia.

Laboratorio: (laboratorio.dat)

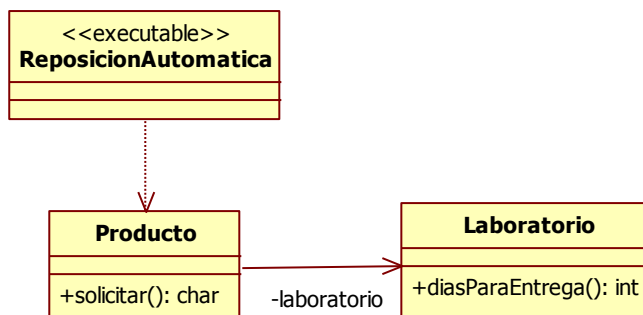
Código Laboratorio (int)	Nombre (String * 30)	Domicilio (String * 30)	Teléfono (String * 30)	Compra Mínima (int)	Día de Entrega (int)
-----------------------------	-------------------------	----------------------------	---------------------------	------------------------	-------------------------

Compra mínima: cantidad mínima del producto que el laboratorio vende en una orden de compra. Valor establecido por el laboratorio.

Día de entrega: Día del mes pautado con el laboratorio para la entrega del producto. Cada laboratorio entrega solo una vez por mes. Ver **Notas Importantes (*)**.

Tener en cuenta que las clases pueden evolucionar para cumplir otro requerimiento. Es posible agregar comportamiento, sin que esto afecte al uso para el cual fue concebida inicialmente. En este caso, el laboratorio necesita conocer la cantidad de días de entrega faltantes, corridos, desde el día actual hasta la próxima entrega del laboratorio. Por ejemplo, si hoy es el día 22 del mes y el laboratorio pauta hacer entregas los días 22 de cada mes, la cantidad de días de entrega faltantes serán de 30. Se debe agregar este comportamiento a la clase modelada previamente.

El siguiente diagrama de Clases representa el diseño de la solución al problema planteado.



En el proceso de solicitud, por cada producto se debe determinar y registrar la razón de su compra, dependiendo de las políticas internas:

- Corresponderá una **F** si la reposición se realiza según los lineamientos de la 1ra. política
- Corresponderá una **P** si la reposición se realiza según los lineamientos de la 2da. Política
- Corresponderá una **N** si el producto no requiere reposición

El punto de reposición en cada producto determina el momento de la reposición, es decir

$$\text{ptoReposición} = \text{Existencia mínima} * (1 + \text{Porcentaje de Reposición} / 100).$$

Simule la gestión de pedidos, mediante una clase ejecutable denominada *ReposicionAutomatica*, donde se realice el armado diario de las solicitudes de compra. Como resultado del proceso se desea obtener el siguiente reporte:

Listado de productos a comprar al 22 de Septiembre de 2016

Producto: Alcohol en gel - biAlcohol Laboratorio:

Colgate S.A.

Domicilio: Scalabrini Ortiz 524 - Teléfono: 54-11 -4239-8447

Cantidad a comprar: 45 paquetes

Día de entrega: 15 del mes

Producto: **Emulsión Corporal Intrahidratante - Hinds** Laboratorio:
Porta Hnos.

Domicilio: **Av.Mitre 3.248** - Teléfono: **54-11 -3925-4700**

Cantidad a comprar: **60** paquetes

Día de entrega: **1** del mes

Número de productos a reponer: **2**

Importe total de la compra planificada: **\$ 1234,56**

Numero de compras por faltante de stock: **1**

Numero de compras por punto de reposición: **1**

Archivo creado: **comprar_20150922.txt**

Notas Importantes (*)

A fin de simplificar la tarea, considerar:

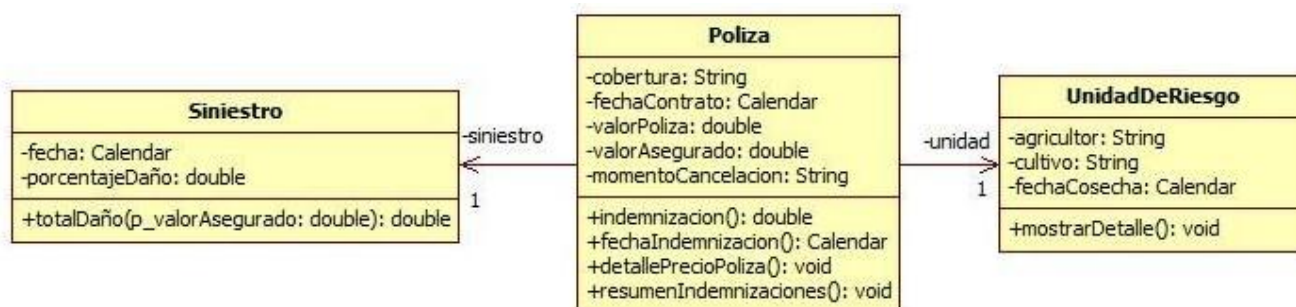
- Que la cantidad a comprar es siempre la compra mínima.
- Que todos los meses tienen 30 días.
- Que todos los días son laborables.

7. Modificar el ejercicio anterior de modo que considere que todos los meses no tienen 30 días. Tomar los días reales de cada mes. Tener en cuenta que Febrero tiene 29 días en año bisiesto.

8. Modificar el ejercicio para que en el cálculo de días para la entrega considere únicamente los días laborables (de lunes a viernes); así, si hoy es miércoles 17 y el laboratorio entrega en 7 días, entonces los días para entrega serían 9 ya que 7 días laborables se convierten en 9 días corridos.

Atención: A los efectos de la entrega para la evaluación en laboratorio, al realizar las modificaciones generar una nueva versión.

9. El Ministerio de Economía y Hacienda conjuntamente al Ministerio de Agricultura administran el Comité de Seguros Agrícolas (COMSA) que se encarga de gestionar todo lo referente a los seguros contratados por los agricultores y los siniestros que se producen con el fin de otorgar futuros subsidios. Un agricultor puede adquirir una póliza de seguro que lo cubra de un eventual siniestro de incendio, granizo o resiembra. Una unidad de riesgo es la porción del campo que el agricultor desea asegurar. El siguiente diagrama de clases modela la solución este enunciado.



Cuando un agricultor contrata una póliza de seguro, debe especificar el cultivo (sorgo, maíz, soja, etc.) que se plantó en la unidad de riesgo. De cada unidad de riesgo se puede conocer un resumen de detalles consistente en:

-Agricultor: Aredo, Rafael
-Cultivo: Trigo

El agricultor puede optar por un momento de cancelación del valor de la póliza que puede ser Inicial, en el momento de comienzo de la siembra, que le asegura un descuento en la contratación, o bien como Final, al término de la siembra. El valor asegurado representa el monto monetario asegurado de la unidad de riesgo.

La póliza contratada puede tener una cobertura de granizo, incendio o resiembra. En cada una de ellas se puede conocer la indemnización en caso de producirse un siniestro. La indemnización depende de la cobertura

contratada. Así, si es granizo, se abona el total del daño producido por el siniestro. Si la cobertura es sobre un incendio se abona el 75% del daño producido por el mismo. En cambio, si se debe realizar una resiembra, se abona el 30% del total del daño producido por el siniestro.

La cobertura también determina la fecha en que el agricultor puede acceder a la indemnización, teniendo en cuenta tanto la fecha del siniestro, como así también, la fecha estipulada de la cosecha que refiere el titular.

	Granizo	Incendio	Resiembra
Total de Daño del siniestro	Indemnización del 100%	Indemnización del 75%	Indemnización del 30%
Fecha de la indemnización	La fecha estipulada de cosecha	20 días después de producido el siniestro	La fecha estipulada de cosecha

El precio de la póliza depende del momento de la cancelación de la misma. Si la cancelación se produce en forma Inicial, el precio de la póliza tiene un 25% de descuento de su valor. En cambio, si la cancelación se produce con carácter de Final, es decir, al finalizar la siembra, se debe abonar el 100% de su valor. El detalle del precio de la póliza será "El abono se realiza al comienzo de la siembra. Tiene un descuento del 25%. Precio Real Abonado: \$412.5" cuando se abona en forma inicial o bien "El abono se realiza al finalizar la siembra. Se abona el 100% de la póliza", cuando se abona al final de la siembra.

Por cada póliza se puede conocer un resumen de la información de la misma.

**** RESUMEN DEL SEGURO CONTRATADO EL DIA 2013/5/4 ****

-----Detalle de forma de pago de la póliza-----

Precio a pagar: \$550.00

El abono se realiza al comienzo de la siembra. Tiene un descuento del 25%. Precio Real Abonado: \$412.5

Valor de cultivo asegurado: \$7500.00

-----Detalle de la unidad de riesgo-----

Agricultor: Aredo, Rafael

Cultivo: Trigo

En cada siniestro se registra la fecha en la que se produjo y el porcentaje de daño que se estima haya ocasionado. Además, es posible determinar el total de daño producido por el mismo teniendo en cuenta el porcentaje de daño estipulado sobre el valor asegurado por el titular.

Cada vez que se realiza el proceso de gestión del comité se graba un registro con los siguientes datos:

Cobertura	Cultivo	Porcentaje de daño	Indemnización
(String)	(String)	(double)	(double)

Simular el proceso de gestión del comité, para obtener el resumen de las indemnizaciones producidas y grabar el archivo correspondiente.