

Orientación a Objetos I

Agenda de la semana
del 23 de septiembre



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA

Actividades de la semana anterior

- Ejercicio 6: Genealogía Salvaje
- Ejercicio 7: Red de Alumbrado
- Ejercicio 8: Method lookup con Empleados

Cuadernillo Semestral de Actividades

Actualizado: 20 de septiembre de 2024

El presente cuadernillo posee un compilado con todos los ejercicios que se usarán durante el semestre en la asignatura. Los ejercicios están organizados en forma secuencial, siguiendo los contenidos que se van viendo en la materia.

Cada semana les indicaremos cuáles son los ejercicios en los que deberían enfocarse para estar al día y algunos de ellos serán discutidos en la explicación de práctica.

Recomendación importante:

Los contenidos de la materia se incorporan y fijan mejor cuando uno intenta aplicarlos - **no alcanza con ver un ejercicio resuelto por alguien más**. Para sacar el máximo provecho de los ejercicios, es importante que asistan a las consultas de práctica habiendo intentado resolverlos (tanto como les sea posible). De esa manera podrán hacer consultas más



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA

Actividades esperadas para esta semana

- Ejercicio 9: Cuenta con ganchos
- Ejercicio 10: Job Scheduler
- Ejercicio 11: El Inversor

Cuadernillo Semestral de Actividades

Actualizado: 20 de septiembre de 2024

El presente cuadernillo posee un compilado con todos los ejercicios que se usarán durante el semestre en la asignatura. Los ejercicios están organizados en forma secuencial, siguiendo los contenidos que se van viendo en la materia.

Cada semana les indicaremos cuáles son los ejercicios en los que deberían enfocarse para estar al día y algunos de ellos serán discutidos en la explicación de práctica.

Recomendación importante:

Los contenidos de la materia se incorporan y fijan mejor cuando uno intenta aplicarlos - **no alcanza con ver un ejercicio resuelto por alguien más**. Para sacar el máximo provecho de los ejercicios, es importante que asistan a las consultas de práctica habiendo intentado resolverlos (tanto como les sea posible). De esa manera podrán hacer consultas más



FACULTAD DE INFORMÁTICA



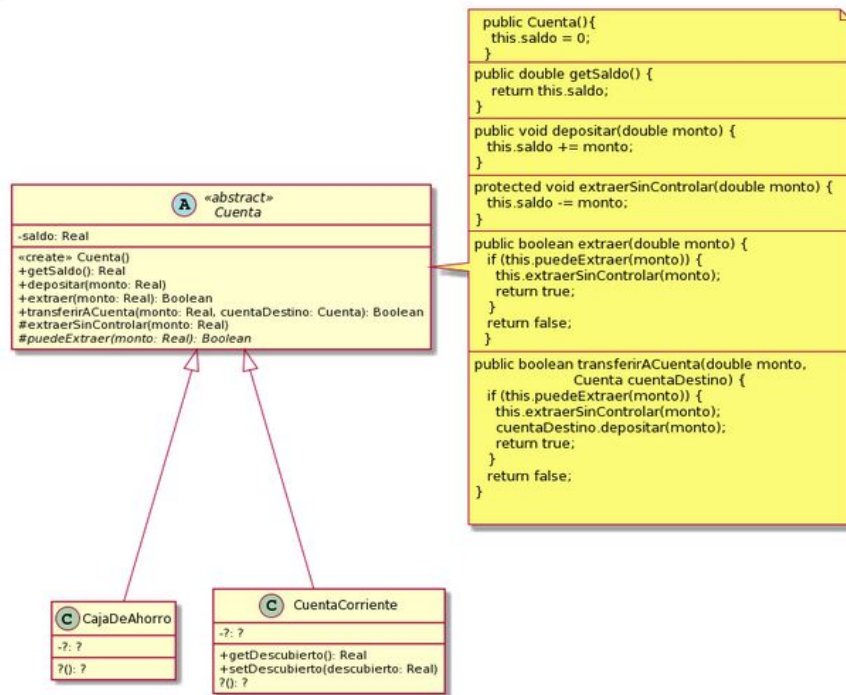
UNIVERSIDAD
NACIONAL
DE LA PLATA

Ejercicio 9: Cuenta con ganchos

Ejercicios de la semana:

- Ejercicio 9: Cuenta con ganchos
- Ejercicio 10: Job Scheduler
- Ejercicio 11: El Inversor

Observe con detenimiento el diseño que se muestra en el siguiente diagrama. La clase cuenta es *abstracta*. El método `puedeExtraer()` es abstracto. Las clases `CajaDeAhorro` y `CuentaCorriente` son concretas y están incompletas.



FACULTAD DE INFORM

Ejercicio 9: Cuenta con ganchos

Ejercicios de la semana:

- **Ejercicio 9: Cuenta con ganchos**
- Ejercicio 10: Job Scheduler
- Ejercicio 11: El Inversor

Tarea A: Complete la implementación de las clases CajaDeAhorro y CuentaCorriente para que se puedan efectuar depósitos, extracciones y transferencias teniendo en cuenta los siguientes criterios.

- 1) Las **cajas de ahorro** solo pueden extraer y transferir cuando cuentan con fondos suficientes.
- 2) Las extracciones, los depósitos y las transferencias desde **cajas de ahorro** tienen un costo adicional de 2% del monto en cuestión (téngalo en cuenta antes de permitir una extracción o transferencia desde caja de ahorro).
- 3) Las **cuentas corrientes** pueden extraer aún cuando el saldo de la cuenta sea insuficiente. Sin embargo, no deben superar cierto límite por debajo del saldo. Dicho límite se conoce como límite de descubierto (algo así como el máximo saldo negativo permitido). Ese límite es diferente para cada cuenta (lo negocia el cliente con la gente del banco).
- 4) Cuando se abre una **cuenta corriente**, su límite descubierto es 0 (no olvide definir el constructor por default).



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA

Ejercicio 9: Cuenta con ganchos

Ejercicios de la semana:

Caja Ahorro
\$ 400

Cuenta Corriente
\$100
límite: \$500

- **Ejercicio 9: Cuenta con ganchos**

depositar(200) ?

?

?

- Ejercicio 10: Job Scheduler

extraer(\$100) ?

?

?

- Ejercicio 11: El Inversor

extraer(\$400) ?

?

?



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA

Ejercicio 9: Cuenta con ganchos

Ejercicios de la semana:

- **Ejercicio 9: Cuenta con ganchos**

- Ejercicio 10: Job Scheduler

- Ejercicio 11: El Inversor

depositar(200) ?

extraer(\$100) ?

extraer(\$400) ?

Caja Ahorro
\$ 400

✓ Saldo \$596

✓ Saldo \$298

✗

Cuenta Corriente
\$100
límite: \$500

✓ Saldo \$300

✓ Saldo \$0

✓ Saldo \$-300



FACULTAD DE INFORMATICA



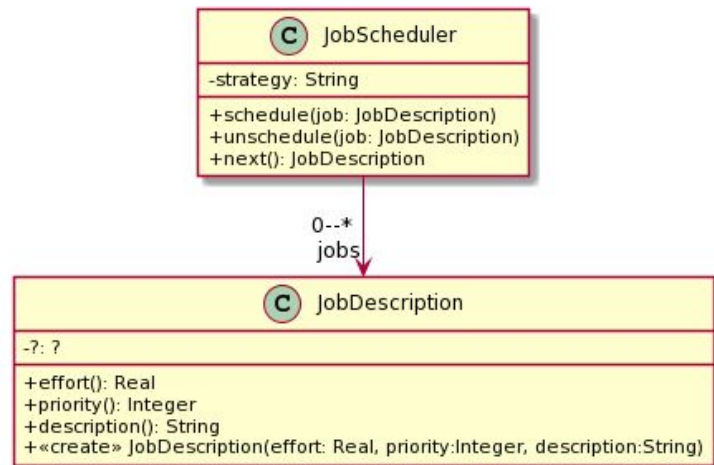
UNIVERSIDAD
NACIONAL
DE LA PLATA

Ejercicio 10: Job Scheduler

Ejercicios de la semana:

- Ejercicio 9: Cuenta con ganchos
- **Ejercicio 10: Job Scheduler**
- Ejercicio 11: El Inversor

El JobScheduler es un objeto cuya responsabilidad es determinar qué trabajo debe resolverse a continuación. El siguiente diseño ayuda a entender cómo funciona la implementación actual del JobScheduler.

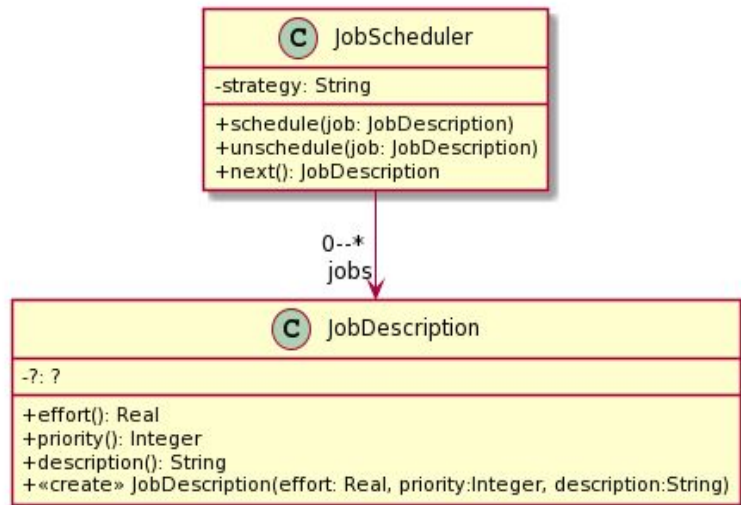


- El mensaje `schedule(job: JobDescription)` recibe un job (trabajo) y lo agrega al final de la colección de trabajos pendientes.
- El mensaje `next()` determina cuál es el siguiente trabajo de la colección que debe ser atendido, lo retorna, y lo quita de la colección.



FACULT

Ejercicio 10: Job Scheduler



FACULTAD

```
public JobDescription next() {
    JobDescription nextJob = null;

    switch (strategy) {
        case "FIFO":
            nextJob = jobs.get(0);
            this.unschedule(nextJob);
            return nextJob;

        case "LIFO":
            nextJob = jobs.get(jobs.size()-1);
            this.unschedule(nextJob);
            return nextJob;

        case "HighestPriority":
            nextJob = jobs.stream()
                .max((j1,j2) -> Double.compare(j1.getPriority(), j2.getPriority()))
                .orElse(null);
            this.unschedule(nextJob);
            return nextJob;

        case "MostEffort":
            nextJob = jobs.stream()
                .max((j1,j2) -> Double.compare(j1.getEffort(), j2.getEffort()))
                .orElse(null);
            this.unschedule(nextJob);
            return nextJob;
    }
    return null;
}
```

Ejercicio 11: El Inversor

Ejercicios de la semana:

- Ejercicio 9: Cuenta con ganchos
- Ejercicio 10: Job Scheduler
- **Ejercicio 11: El Inversor**

Ejercicio 11 - El Inversor

Estamos desarrollando una aplicación móvil para que un inversor pueda conocer el estado de sus inversiones. El sistema permite manejar dos tipos de inversiones: Inversión en acciones e inversión en plazo fijo. Nuestro sistema representa al inversor y a cada uno de los tipos de inversiones con una clase.

- La clase InversionEnAcciones tiene las siguientes variables de instancia:

String nombre;
int cantidad;
double valorUnitario;

- La clase PlazoFijo tiene las siguientes variables de instancia:

LocalDate fechaDeConstitucion;
double montoDepositado;
double porcentajeDelInteresDiario;

- La clase Inversor tiene las siguientes variables de instancia:

String nombre;
List<?> inversiones;



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA

Ejercicio 11: El Inversor

Ejercicio 11 (El inversor)



InversionEnAcciones

- nombre
- cantidad
- valorUnitario

Ejercicio 11 - El Inversor

Estamos desarrollando una aplicación móvil para que un inversor pueda conocer el estado de sus inversiones. El sistema permite manejar dos tipos de inversiones: Inversión en acciones e inversión en plazo fijo. Nuestro sistema representa al inversor y a cada uno de los tipos de inversiones con una clase.

- La clase InversionEnAcciones tiene las siguientes variables de instancia:

```
String nombre;  
int cantidad;  
double valorUnitario;
```

- La clase PlazoFijo tiene las siguientes variables de instancia:

```
LocalDate fechaDeConstitucion;  
double montoDepositado;  
double porcentajeDelInteresDiario;
```

- La clase Inversor tiene las siguientes variables de instancia:

```
String nombre;  
List<?> inversiones;
```



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA

Ejercicio 11: El Inversor

El valor actual de una **InversionEnAcciones** se calcula **multiplicando** el **número de acciones** por el **valor unitario** de las mismas.

meli: InversionEnAcciones

nombre = meli

cantidad = 5

valorUnitario = \$ 1.000



¿Qué retorna meli.valorActual() ?

$$5 * 1.000 = 5.000$$



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA

Ejercicio 11: El Inversor

Ejercicio 11 (El inversor)



InversionEnAcciones

- nombre
- cantidad
- valorUnitario



Plazo fijo

- fechaDeConstitucion
- montoDepositado
- porcentajeDelInteresDiario

Ejercicio 11 - El Inversor

Estamos desarrollando una aplicación móvil para que un inversor pueda conocer el estado de sus inversiones. El sistema permite manejar dos tipos de inversiones: Inversión en acciones e inversión en plazo fijo. Nuestro sistema representa al inversor y a cada uno de los tipos de inversiones con una clase.

- La clase InversionEnAcciones tiene las siguientes variables de instancia:

```
String nombre;  
int cantidad;  
double valorUnitario;
```

- La clase PlazoFijo tiene las siguientes variables de instancia:

```
LocalDate fechaDeConstitucion;  
double montoDepositado;  
double porcentajeDelInteresDiario;
```

- La clase Inversor tiene las siguientes variables de instancia:

```
String nombre;  
List<?> inversiones;
```



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA

Ejercicio 11: El Inversor

El **valor actual** de un plazo fijo equivale al **montoDepositado** incrementado como corresponda por el **porcentaje de interés diario**, desde la fecha de constitución a la fecha actual (la del momento en el que se hace el cálculo).

ahorro: PlazoFijo



fechaDeConstitucion = 01/09/2023
montoDepositado = \$ 3.000
porcentajeDeInteresDiario = 1%

¿Qué retorna ahorro.valorActual() ?

$$3.000 + 24 * 30 = 3.720$$



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA

Ejercicio 11: El Inversor

Ejercicio 11 (El inversor)



InversionEnAcciones

- nombre
- cantidad
- valorUnitario

Plazo fijo

- fechaDeConstitucion
- montoDepositado
- porcentajeDeInteresDiario

inversor.valorActual()



Ejercicio 11 - El Inversor

Estamos desarrollando una aplicación móvil para que un inversor pueda conocer el estado de sus inversiones. El sistema permite manejar dos tipos de inversiones: Inversión en acciones e inversión en plazo fijo. Nuestro sistema representa al inversor y a cada uno de los tipos de inversiones con una clase.

- La clase InversionEnAcciones tiene las siguientes variables de instancia:

```
String nombre;  
int cantidad;  
double valorUnitario;
```

- La clase PlazoFijo tiene las siguientes variables de instancia:

```
LocalDate fechaDeConstitucion;  
double montoDepositado;  
double porcentajeDeInteresDiario;
```

- La clase Inversor tiene las siguientes variables de instancia:

```
String nombre;  
List<?> inversiones;
```



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA

Ejercicio 11: El Inversor

meli: InversionEnAcciones

nombre = meli
cantidad = 5
valorUnitario = \$ 1.000

ahorro: PlazoFijo

fechaDeConstitucion = 01/09/2023
montoDepositado = \$ 3.000
porcentajeDelInteresDiario = 1%

ale: Inversor

nombre = ale

¿Qué retorna ale.valorActual() ?

$$5.000 + 3.720 = 8.720$$



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA

Foros de consulta

Cómo preguntar en el foro

Breve guía para poder sacar el mejor provecho al foro y a la convivencia a través de las preguntas y respuestas.

Cómo preguntar en el foro

Antes de Preguntar: Busca una respuesta por tus propios medios

Elegí el foro específico

Elegí un título apropiado para la pregunta

No envíes una solución para que la corrijan

Describir qué estás intentando hacer

Describir el problema y lo que has intentado para resolverlo

Escribir claro


No solicites respuestas a tu correo

Si no entendés la respuesta


Terminá con una breve nota de conclusión.

Evitá el "Me sumo al pedido"

Foros

 Ejercicio 3 - Presupuestos

Foros

  Ejercicio 5 - Genealogía Salvaje

 Ejercicio 6: Red de Alumbrado

 Ejercicio 7: Method lookup con Empleados



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA