

Práctica de Repaso

Objetivo. Repasar los temas de POO vistos en el módulo. Repasar el uso de Netbeans: crear proyecto, cargar paquete de lectura y exportar proyecto.

Nota: Resuelva cada ejercicio **en un nuevo proyecto y cargue el paquete de lectura**. Finalizado el desarrollo, **exporte/comprima su proyecto (a .zip)**. Puede encontrar las instrucciones en la Guía de uso rápida de Netbeans.

1- La UNLP desea administrar sus proyectos, investigadores y subsidios. Un proyecto tiene: nombre, código, nombre y apellido del director y los investigadores que participan en el proyecto (50 como máximo). De cada investigador se tiene: nombre y apellido, categoría (1 a 5) y su especialidad. Además, cualquier investigador puede pedir hasta un máximo de 5 subsidios. De cada subsidio se conoce: el monto pedido, el motivo y si fue otorgado o no.

i) Implemente el modelo de clases teniendo en cuenta:

- a. Un proyecto sólo debería poder construirse con el nombre y el código.
- b. Un investigador sólo debería poder construirse con nombre y apellido, categoría y especialidad.
- c. Un subsidio sólo debería poder construirse con el monto solicitado y el motivo. Un subsidio siempre se crea en estado no-otorgado.

ii) Implemente los métodos necesarios (en las clases donde corresponda) que permitan:

- a. `void agregarInvestigador(unInvestigador);`
// agrega un investigador a un proyecto.
- b. `void agregarSubsidio(unSubsidio);`
// agrega un subsidio a un investigador.
- c. `double dineroTotalOtorgado();`
// devuelve la cantidad de dinero total de todos los subsidios otorgados a todos los investigadores de un proyecto.
- d. `int cantidadDeSubsidios(String nombre_y_apellido);`
// devuelve la cantidad de subsidios (otorgados o no) solicitados por el investigador llamado "nombre_y_apellido".
- e. `void otorgarTodos(String nombre_y_apellido);`
// otorga todos los subsidios pendientes que tiene el investigador llamado "nombre_y_apellido"
- f. `String toString();`
// Devuelve un string que tiene el nombre del proyecto, su código, el nombre y apellido del director, el total de dinero otorgado y el nombre y apellido de cada investigador. Para cada investigador, además, se debe agregar la categoría del mismo y el dinero de sus subsidios otorgados.

iii) Escriba un programa que instancie un proyecto con un director y dos investigadores. Asigne dos subsidios a cada investigador y otorgue los subsidios del primero de ellos, luego imprima todos los datos del proyecto en pantalla.

Taller de Programación 2022 – Módulo POO

2- Un manager de bandas de música desea un sistema para manejar los recitales que organiza. El manager maneja dos tipos de recitales: eventos ocasionales y giras.

- Todo recital se caracteriza por el nombre de la banda y la lista de temas que tocarán durante el recital.
- Un evento ocasional es un recital que además tiene el motivo (a beneficio, show de TV o show privado), el nombre de la persona que contrata el recital y el día del evento.
- Una gira es el mismo recital que se repite varias veces en distintas ciudades. Una gira tiene un nombre, y las “fechas”, de cada fecha se conoce la ciudad y el día. Además una gira deberá guardar la información de la fecha “*actual*” a tocar.

a) Realice el modelo de clases. Implemente las clases con sus atributos y métodos para obtener/modificar el valor de los mismos.

b) Implemente los constructores en todas las clases. El constructor de recitales recibe el nombre de la banda y la cantidad de temas que tendrá el recital. El constructor de eventos ocasionales además recibe el motivo, la persona que lo contrata y día del evento. El constructor de giras además recibe el nombre de la gira y la cantidad de fechas que tendrá la gira.

c) Incorpore los métodos listados a continuación:

i. Cualquier recital debe saber responder a los mensajes:

- **agregarTema** que recibe el nombre de un tema y lo almacena adecuadamente.
- **actuar** que imprime por consola y para cada tema, la leyenda “y ahora tocaremos...” seguido por el nombre del tema.

ii. La gira debe saber responder a los mensajes:

- **agregarFecha** que recibe la fecha (con ciudad y día) y la almacena adecuadamente.
- La gira debe responder al mensaje **actuar** de manera distinta. Imprime la leyenda “Buenas noches ...” seguido del nombre de la ciudad “actual”. Luego debe imprimir el listado de temas como lo hace cualquier recital. Además debe setearse correctamente el siguiente recital de la gira como el “actual”.

iii. El evento ocasional debe saber responder al mensaje actuar de manera distinta:

- Si es un show de beneficencia se imprime la leyenda “Recuerden colaborar con...” seguido del nombre de quien contrató el evento.
- Si es un show de TV se imprime “Saludos amigos televidentes”
- Si es un show privado se imprime “Un feliz cumpleaños para...” seguido del nombre de quien contrató el evento.

Independientemente del motivo del evento, luego se imprime el listado de temas como lo hace cualquier recital.

iv. Todo recital debe saber responder a los mensajes:

- **finalizado**: Si es un evento ocasional devuelve true si el recital ya se llevó a cabo o false en caso contrario. Si es una gira devuelve true si ya se actuaron todas las fechas o false en caso contrario.

- **calcularCosto:** Si es un evento ocasional devuelve 0 si es a beneficio, 50000 si es un show de TV y 150000 si es privado. Las giras deben devolver 30000 por cada fecha de la misma.

d) Realice un programa que instancie un evento ocasional. Lea desde el teclado el día del evento, el nombre de quien lo contrata, el motivo, la banda y el listado de temas a tocar. Luego imprima el costo del evento e invoque el mensaje actuar del evento.

A continuación instancie una gira, leyendo desde el teclado el nombre de la gira, la banda, el listado de temas y el listado de ciudades con sus correspondientes días. Luego imprima el costo de la gira e invoque al mensaje actuar de la misma hasta finalizar la gira.

3- Una escuela de música arma coros para participar de ciertos eventos. De los coristas se conoce el nombre, el dni, la edad y el tono fundamental (un número entero). Del director se conoce el nombre, el dni, la edad y la antigüedad (un número entero). Los coros poseen un nombre y están formados por un director y una serie de coristas. Asimismo pueden formarse de dos formas: o bien los coristas se colocan en el escenario uno al lado del otro formando un *semicírculo*, o bien conforman *hileras* de la misma cantidad de coristas.



- Implemente el modelo de clases teniendo en cuenta que los coros deberían crearse con un director y sin ningún corista, pero sí sabiendo las dimensiones del coro.
- Implemente métodos (en las clases donde corresponda) que permitan:
 - agregar un corista a un coro.
 - En el coro *semicircular* los coristas se deben ir agregando de izquierda a derecha
 - En el coro *por hileras* los coristas se deben ir agregando de izquierda a derecha completando la hilera antes de pasar a la siguiente, comenzando por la hilera de adelante.
 - determinar si un coro está lleno o no. Devuelve true si el coro tiene a todos sus coristas asignados o false en caso contrario.
 - determinar si un coro (se supone que está lleno) está bien formado. Un coro está bien formado si:
 - En el caso del coro *semicircular*, de izquierda a derecha los coristas están ordenados de mayor a menor en cuanto a tono fundamental.
 - En el caso del coro *por hileras*, desde adelante hacia atrás los coristas están ordenados de mayor a menor en cuanto a tono fundamental y todos los miembros de una misma hilera tienen el mismo tono fundamental.
 - devolver la representación de un coro formada por el nombre del coro, todos los datos del director y todos los datos de todos los coristas.

Taller de Programación 2022 – Módulo POO

- c. Escriba un programa que instancie un coro de cada tipo. Se deberá leer o bien la cantidad de coristas (en el caso del coro *semicircular*) o la cantidad de hileras e integrantes por hilera (en el caso del coro por *hileras*). Luego se deberá crear la cantidad de coristas necesarios, leyendo sus datos, y almacenándolos en el coro. Finalmente imprima toda la información de los coros indicando si están bien formados o no.