Prácticas POO
Curso 08/09

Alejandro Bellogín

Escuela Politécnica Superior
Universidad Autónoma de Madrid
Febrero 2009

http://www.eps.uam.es/~abellogin

Esquema

- Contacto
- Nociones básicas (SO, IDE, …)
- Algunas cosas de Java
- Proyecto
- Organización de las prácticas
- Más cosas de Java
- A practicar
- Extra

Contacto

- Correo electrónico (preferible: asunto '[poo]'):

  alejandro . bellogin @ uam . es
- Despacho: B – 407 - 1
- Hora de tutorías?
  - Tentativo: Jueves de 18:00 a 20:00
- En clase: martes de 14:00 a 16:00
- Teléfono: 91 497 23 58

Nociones básicas

- Sistema Operativo
  - Indiferente (siempre y cuando tenga instalada una Máquina Virtual de Java (JVM))

- IDE: entorno de desarrollo (programar, ejecutar, depurar)
  - NetBeans: desarrollado por Sun, por defecto (v6.5)
  - Eclipse: desarrollado por Eclipse Foundation, muchos plugins
  - JCreator: menos pesado, sólo Win, no ejecuta, no libre

(Pequeña) Introducción a Java

- No es C
- Convenciones (nombres, estilo, …)
  - http://arantxa.ii.uam.es/~poo/practicas/codeconv/CodeConvTOC.doc.html
- Librerías
- Orientado a Objetos

Prácticas

- Cuatro prácticas:
  - P1 (10%, 2 semanas) : introducción (diseño, primeras clases)
  - P2 (35%, 3 sem): clases y herencia (más clases)
  - P3 (35%, 3 sem): interfaces gráficas (ventanas, eventos)
  - P4 (20%, 2 sem): programación distribuida y JSF

    (…en distintos ordenadores!)
- Las prácticas se corrigen mediante inspección de lo entregado y con un *examen presencial* de cada práctica.
- 40% de la nota final

Más información:
http://arantxa.ii.uam.es/~poo/practicas/normas.html

Calendario

Lunes D	**Martes** **F, E**	Miercoles C	Jueves B	Viernes A	Práctica/Explicaciones
23 Feb. Inicio P1	**24 Feb.** **Inicio P1**	25 Feb. Inicio P1	26 Feb. Inicio P1	27 Feb. Inicio P1	P1: introducción a Java; incluye introducción al entorno
02 Mar.	**03 Mar.**	04 Mar.	05 Mar.	06 Mar.	
09 Mar. Entrega P1 Inicio P2	**10 Mar.** **Entrega P1** **Inicio P2**	11 Mar. Entrega P1 Inicio P2	12 Mar. Entrega P1 Inicio P2	13 Mar. Entrega P1 Inicio P2	P2: Clases y Herencia
16 Mar.	**17 Mar.**	18 Mar.	19 Mar. S. José	20 Mar.	
23 Mar.	**24 Mar.**	25 Mar.	26 Mar.	27 Mar.	
30 Mar. Entrega P2 Inicio P3	**31 Mar.** **Parciales 3º**	01 Abr. Parciales 3º	02 Abr. Entrega P2 **(Grupos M, X, J)** Inicio P3	03 Abr. Entrega P2 Inicio P3	P3: Interfaces Gráficas
06 Abr.	**07 Abr.**	08 Abr.	09 Abr.	10 Abr.	Vacaciones de Semana Santa
13 Abr.	**14 Abr.** **Inicio P3**	15 Abr. Inicio P3	16 Abr.	17 Abr.	
20 Abr.	**21 Abr.**	22 Abr.	23 Abr.	24 Abr.	
27 Abr.	**28 Abr.**	29 Abr.	30 Abr.	01 May. Día del trabajo	
04 May. Entrega P3 Inicio P4	**05 May.** **Entrega P3** **Inicio P4**	06 May. Entrega P3 Inicio P4	07 May. Entrega P3 Inicio P4	08 May. Entrega P3 Inicio P4	P4: Programación distribuida y JSF
11 May.	**12 May.**	13 May.	14 May.	15 May. S. Isidro	
18 Abr. Fiesta EPS	**19 May.**	20 May.	21 May.	22 May.	
25 May. Entrega P4	**26 May.** **Entrega P4**	27 May. Entrega P4 **(Grupos X, J, V)**	---	---	Última clase

Normativa

- Para promediar es necesario aprobar independientemente la teoría y las prácticas. Se convalidarán con una nota de 5 las prácticas aprobadas de un año para el siguiente, siempre que se contase con un 3 o más en el examen de teoría correspondiente.
- Los grupos de trabajo serán de 2 personas
- La copia en prácticas es una falta grave que será objeto de sanción (que puede ser extensible al copiado): apertura de **expediente de expulsión** o bien **suspenso automático sin posibilidad de presentarse a Septiembre**.
- El intercambio de *ideas* no se considera copia (es más, se recomienda encarecidamente). El intercambio de código fuente sí, y se castigará como tal. El estudiante es responsable de evitar que su material evaluable (código, problemas, ejercicios, memorias de prácticas, etc.) sea accesible a estudiantes de otros grupos de prácticas.
- Para aprobar la asignatura, es obligatorio haber entregado todas las prácticas.
- Los alumnos que entreguen en **lunes** deben hacerlo **como tarde 2 horas antes** del comienzo de la clase de prácticas. Aquellos alumnos que entreguen **cualquier otro día** de la semana tienen como plazo **las 23:59 del día anterior**. Los retrasos dentro del mismo día de entrega descontarán un 20% de la nota. A partir del primer día de retraso, cada día sucesivo resta otro 10% del total, llegándose al 100% a los 8 días de cumplirse el plazo. No se considera como entrega aquella que sólo contiene código o sólo contiene la memoria.
- La última semana de prácticas se reserva para un examen de prácticas para aquellos alumnos que no han podido asistir a alguna revisión, con entregas que no funcionan, etcétera.
- Las prácticas se enviarán como un único fichero mediante el sistema de entrega de prácticas de la Escuela. Nombre: p<*número de práctica*><*letra de grupo*><*número de pareja, 2 dígitos*>.zip

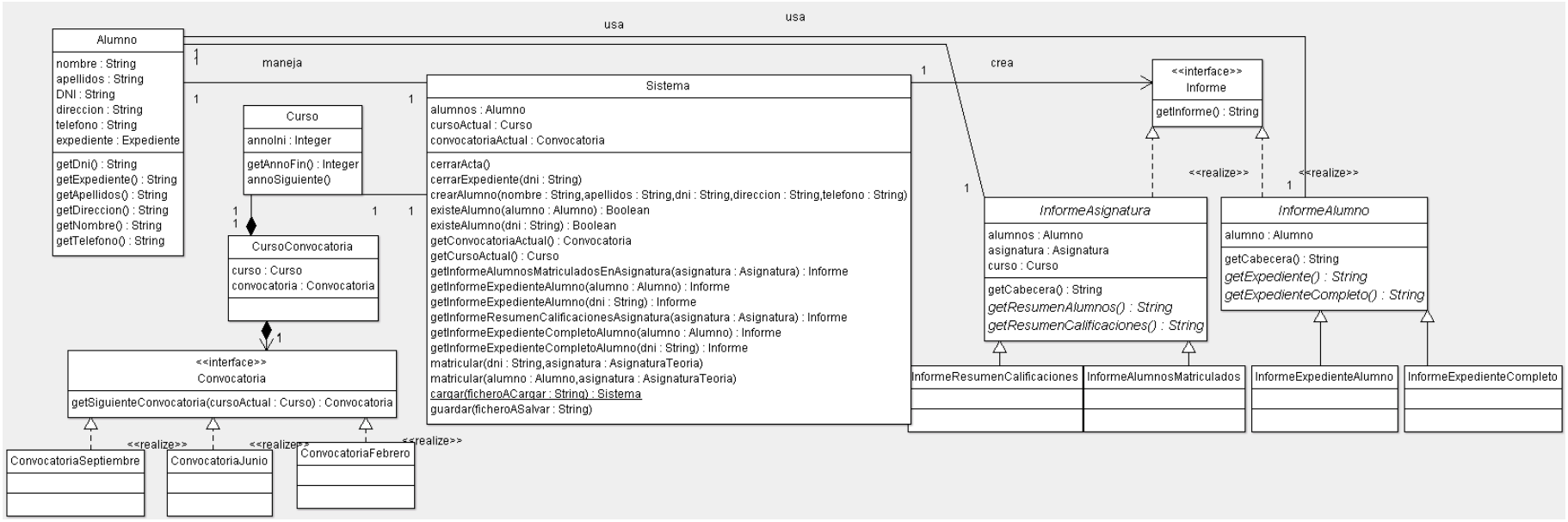
http://arantxa.ii.uam.es/~poo/practicas/normas.html

Proyecto

- Aplicación que:
  - Crea alumnos
  - Matricula un alumno de n asignaturas
  - Cierra expedientes y calcula medias
  - Calcula calificación final de las asignaturas (usando teoría y práctica, si tiene)
  - Cierra actas (para la convocatoria y curso actual, permite ‘No consume’)
  - Genera informe de asignaturas y de alumnos

Práctica 1

- Analizar el problema que se plantea
- Implementar las primeras clases Java



  - Completar diagrama
  - Alumno: getters, setters, toString()

Conceptos necesarios para la P1

- De Orientación a Objetos:
  - Herencia
  - Interfaces

- De Java
  - Noción de paquete (*package*)
  - Atributos, métodos (esp. *toString()*), constructores

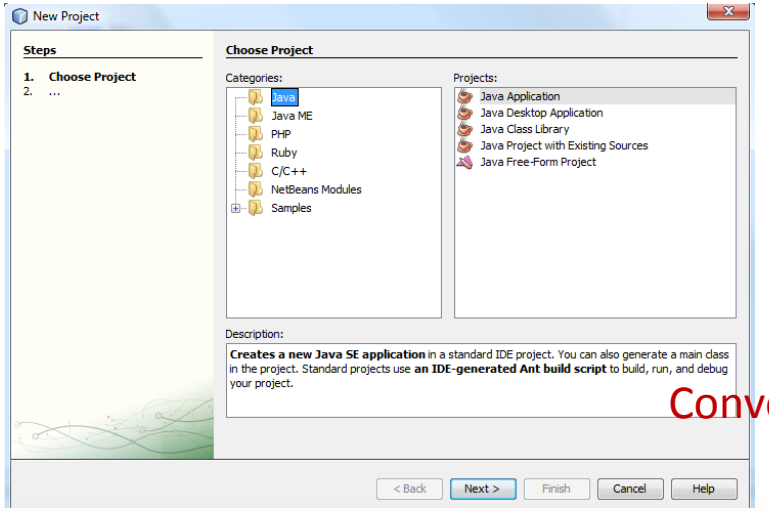
UML (diagrama de clases)

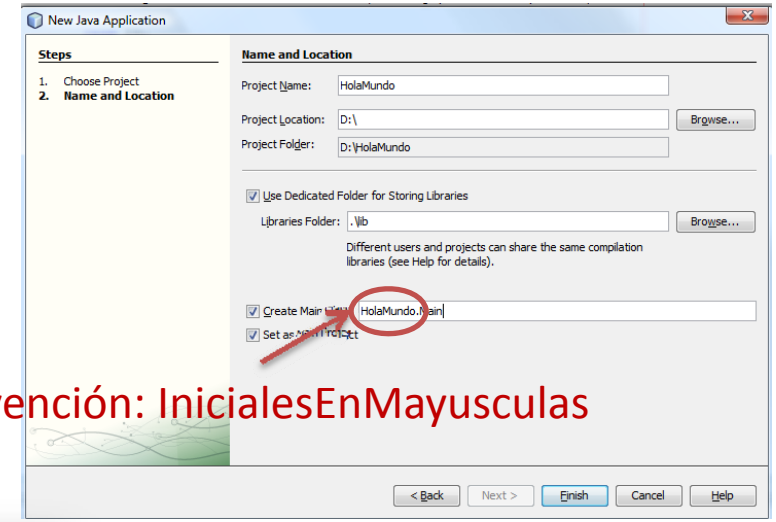
- Relaciones entre clases:
  - Asociación 
  - Agregación (“es parte de …”) 
  - Generalización o herencia (“es un caso particular de …”) 
- Navegabilidad: unidireccional vs bidireccional 
- Pueden tener nombres (roles)

Conceptos Java que veremos en otras prácticas

- Herencia e interfaces
- Clases abstractas
- Manejo de excepciones
- Hilos
- Programación distribuida

Practica: ‘Hola Mundo’ con NetBeans

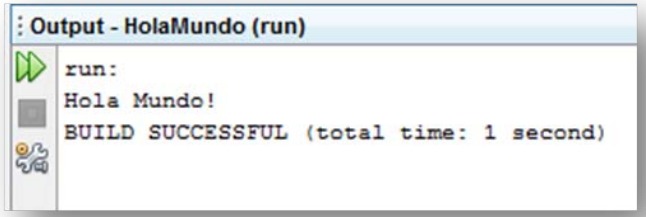




Convención: InicialesEnMayusculas

```
6  package HolaMundo;
7
8  /**
9   *
10  * @author Alejandro
11  */
12 public class Main {
13
14     /**
15      * @param args the command line arguments
16      */
17     public static void main(String[] args) {
18         System.out.println("Hola Mundo!");
19     }
20
21 }
22
```

+ F6 =



Practica: depura con NetBeans

FIN

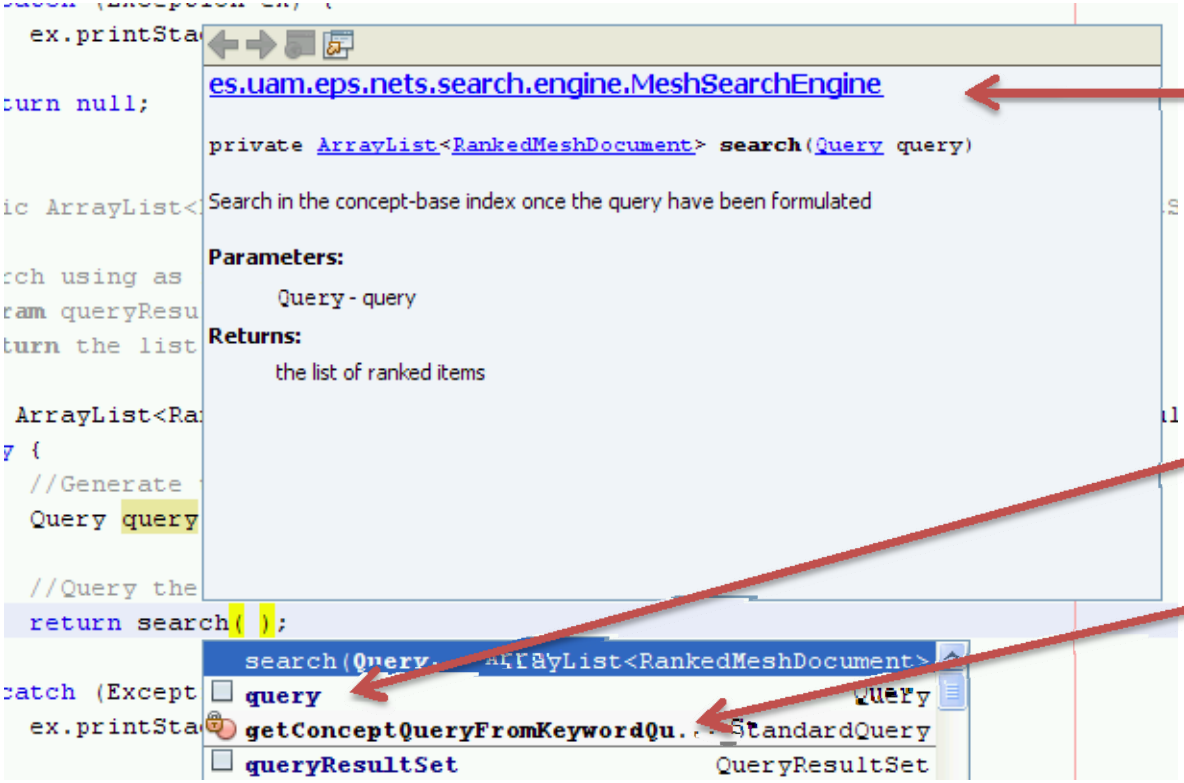
Más cosas de NetBeans

- A veces parece que NetBeans programa solo:
  - Atajos de teclado (combinaciones de teclas, completar código)
  - Ingeniería Inversa
  - …

Alt+Shift+F	Formatear código
Ctrl+R	Renombrar (+ refactor!)
Alt+Shift+C	Comentar
‘sout’ + tab/espacio	‘System.out.println()’
‘re’ + tab/espacio	‘return ‘
…	…

- Integrado un Profiler (similar al Valgrind): analiza memoria y performance
- Viene con servidor de aplicaciones (Tomcat)
- CVS, SVN

Un programa que hace programas!



Documentación en el momento

Variable más adecuada?

Método que devuelve un tipo compatible

Hay que decir que Eclipse tiene una funcionalidad muy parecida…