



Control d'accès



JAVA



Introducció

- Principi: separar el que canvia del que ha de romandre igual.
- Molt important per llibreries de codi. Els programadors “clients” no han de reescriure el seu codi en fer millores al codi de la llibreria.
- JAVA té “especificadors d’accés” per separar la part que està disponible per als “clients” de la que no ho està.
 - public
 - package access
 - protected
 - private

Unitats de compilació

- En crear un nou fitxer de codi font java, dins aquest hi pot haver una classe “public” que ha de tenir necessàriament el mateix nom que el fitxer (sense el suffix “.java”).
- Quan compilem un arxiu de codi font, obtenim un fitxer .class per cada classe definida.
- Un programa JAVA no és més que una col·lecció de fitxers “.class”, empaquetats dins un Java ARchive (JAR). La màquina virtual java executarà directament aquest fitxer.

Package (unitat de llibreria)

- Un “package” conté un grup de classes, organitzats dins un espai de noms (anomenat “namespace”).
 - Exemple: dins el namespace “java.util” hi ha una classe anomenada “ArrayList”.
- Importacions
 - Per evitar escriure el nom llarg de cada classe de les llibreries, podem emprar la sentència “import”.
 - `import “java.util.ArrayList”`
 - `import “java.util.*”`
- Hem d’evitar les col·lisions (noms de classes iguals en diferents paquets).

Package

- Una llibreria és un conjunt de classes. Podem agrupar-les amb la paraula clau “package” de JAVA.
- Estem dient que les classes públiques de les unitats de compilació estaran totes baix el nom especificat en el paquet. Si algú vol usar aquestes classes, haurà d'especificar el seu nom complet o emprar la sentència “import”.
- Serveix per organitzar el codi (conjunt de classes) relacionades.
- Una bona manera de tenir noms de paquets “únicos” és emprar el domini invertit del creador de la classe.

Package

- Exemple de l'anterior:
 - Suposem que publiquem codi JAVA de l'organització "liceu".
 - El nom de domini internet invertit seria: com.liceu
 - Si volem publicar una llibreria per gràfics del liceu, podríem emprar el nom "com.liceu.graphics".
- Si tothom segueix aquest esquema, el codi de les llibreries JAVA queda ordenat d'una manera molt elegant.
- Les llibreries originals de java no segueixen aquesta convenció, per no rompre la compatibilitat amb programes existents.
- Les possibles col·lisions s'arreglen especificant el nom complet de la classe sense ambigüitat.

Especificadors d'accés en JAVA

- Accés de paquet (Package Access)
 - No hi ha keyword per aquest mode.
 - Les classes sense especificador són visibles dins el package, però no des de fora
 - Té sentit: el codi intern de la llibreria (package) accedeix per defecte a tota la resta de classes dins el mateix paquet.
- Accés públic:
 - Emprem la keyword “public” en la declaració de la classe. Només hi pot haver una classe pública per cada fitxer .java. Dins un paquet n’hi pot haver varies.
 - Les classes declarades “public” podran ésser accedides des de fora del paquet.

Especificadors d'accés en JAVA

- **Accés privat:**
 - Emprem la keyword “private”.
 - Els mètodes, atributs i classes anidades declarades com a “private” no són accessibles des de fora de la classe original que els defineix.
 - Exemple: Si declarem un constructor com a private, no podem instanciar un objecte des de fora de la classe amb l'operador “new”.
 - Els mètodes i atributs d'ús intern dins una classe es poden fer “private”, per evitar que s'utilitzin accidentalment per part d'altres classes.
 - Alerta amb les referències: poden apuntar a objectes no-privats.
- **Accés protegit.**
 - Té a veure amb un concepte més complex: l'herència.

Interfície i implementació

- Aquestes tècniques s'anomenen també “ocultació de la implementació” (implementation hiding).
- El control d'accés posa límits als tipus de dades per dues raons:
 - Permetre l'evolució interna del codi del tipus (classe).
 - Separar la interfície de la implementació de manera molt clara.
- Consell: en especificar una classe, posar els membres públics al principi, seguit dels “protected” i després dels “private”.
- El “client” de la llibreria tendrà més fàcil trobar el que li interessa.

Accés a les classes dins una llibreria

- Si volem que la classe estigui disponible per tothom, fins i tot fora de la llibreria, emprem el modificador “public”.
- No podem declarar una classe “private” o “protected”. Això no tindria sentit perquè es denegaria l'accés a tothom manco la pròpia classe.
- Les classes sense modificador d'accés són d'accés només dins el package.
- Les classes poden tenir membres:
 - Sense modificador
 - public
 - protected
 - private