

Lezione 5

Gli IDE

Inizieremo ad utilizzare Visual Studio Code per una serie di motivi:

- Rende molto facile alcune versioni quali l'editing, il beautify, ecc...
- Predisporre alla creazione di progetti
- Supporta più linguaggi

Convenzioni sulla formattazione del codice

- Indentazioni di 2-4 righe
- Commenti POCHI ma buoni:
 - `//` su singola linea
 - `/* */` per commentare sezioni
- Righe non troppo lunghe, se no si va a capo
- Graffe messe nel posto giusto
- Per il resto delle regole vedi le lezioni di laboratorio.

Decomposizione / Modularità

La OOP segue il principio del "divide et impera", in modo da rendere più semplice la risoluzione di un grande problema tramite la soluzioni di suoi sottoproblemi più facili.

Una soluzione è resa facile tramite la **modularità**: se spezziamo un problema in tre parti, una soluzione è modulare se si hanno poche dipendenze esterne (ossia tra parte e parte), ma solo dipendenze interne (ossia all'interno delle parti stesse). Ma cos'è una dipendenza? Una dipendenza tra classi è una **menzione** di una classe B all'interno di un'altra classe A; in questo caso A dipende da B.

C'è una implicazione importante con le dipendenze: se modifico la classe B, da cui la classe A dipende, dovrò modificare anche la classe A stessa.

Incapsulamento

C'è un assioma nella OOP: le classi devono lasciare trasparire solo i metodi e i campi strettamente necessari all'utente, nascondendo quelli invece non necessari. Nel nostro caso:

- Verranno dichiarati `public` quei pochi metodi utili a interagire o creare le istanze delle classi.
- Il resto delle cose verrà reso `private` ; anche **tutti** i campi vanno sempre messi `private` .

Immutabili

Gli oggetti immutabili sono oggetti per cui è garantito che il loro stato iniziale non cambierà mai. Ciò significa che i campi della classe sono `final` , pertanto non potranno essere modificati in corso d'opera.