Analisi esercizio classi astratte ed interfacce

Analisi del problema

Il problema è facilmente risolvibile con l’utilizzo di classi astratte ed interfacce.

Le classi astratte sono classi che servono da “modello” per altre classi: non possono quindi essere instanziate, ma possono contenere metodi (sia definiti che non), attributi e costruttori.

Le interfacce sono classi speciali che contengono sono metodi (non costruttori), e niente attributi (ad eccezione di quelli dichiarati final ed istanziati).

Individuazione delle classi

Le classi richieste sono le seguenti:

* PoligonoRegolare, che sarà la super-classe astratta di:
  + Triangolo;
  + Pentagono;
  + Esagono;
  + Quadrato, che avrà come classe figlia:
    - FiguraQuadrato, che implementa:
* L’interfaccia Figura, la quale sarà la classe madre di:
  + FiguraPiana.

Individuazione di attributi e metodi di ogni classe

abstract PoligonoRegolare:

* Attrbuti:
  + protected final int N\_LATI;
  + protected float lato;
  + protected final float N\_FISSO;
* Metodi:
  + Costruttore (int nLati, float lato, float nFisso);
  + float area();
  + float perimetro();
  + protected final float apotema();

Le classi Triangolo, Quadrato, Pentagono ed Esagono conterranno solo il costruttore (ad eccezione di Quadrato che ridefinirà il metodo area());

interface Figura:

* Metodi:
  + void setNome(String nome);
  + String getNome();

FiguraPiana e FiguraQuadrato conterrnno solo un attributo (private String nome) ed i costruttori.

Impostazione dei metodi costruttori

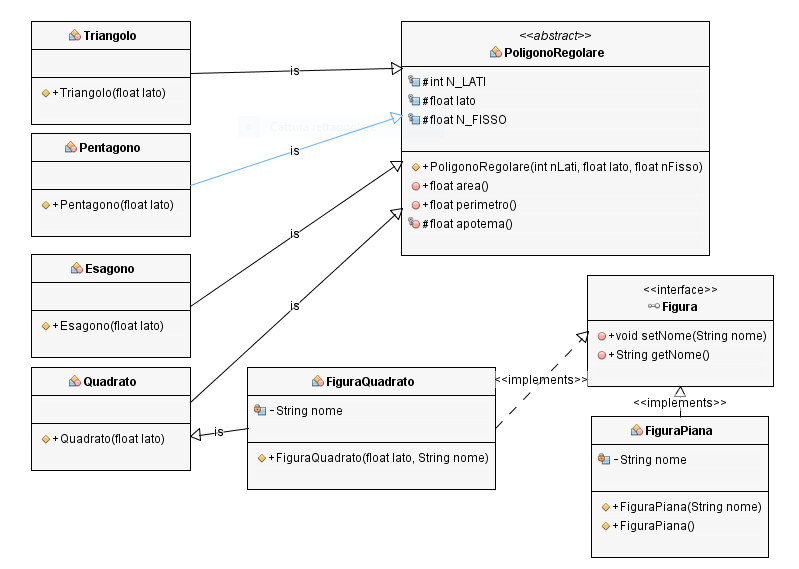
Il metodo costruttore di PoligonoRegolare accetterà come parametri il numero di lati del poligono, la sua lunghezza ed il numero fisso;

Le quattro classi Triangolo, Quadrato, Pentagono ed Esagono avranno solo “float lato” in ingresso (gli altri parametri da dare al costruttore della super-classe verranno inseriti nel codice);

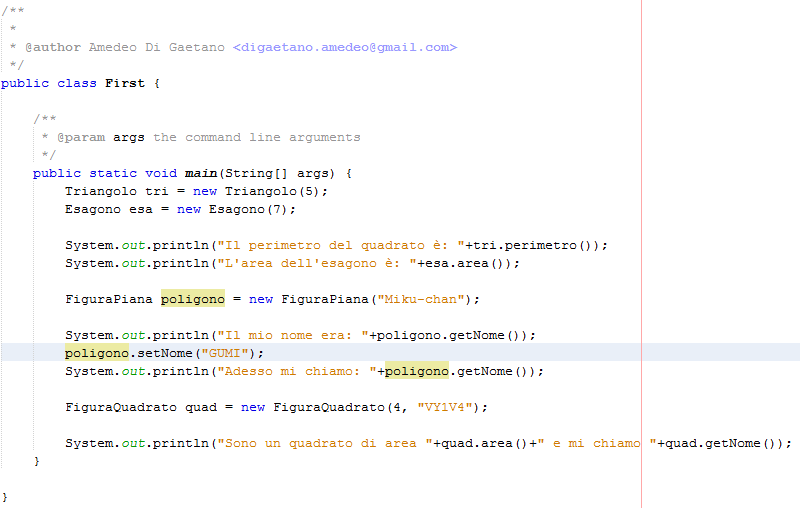
La classe FiguraPiana avrà due costruttori (uno senza parametri; uno con String nome);

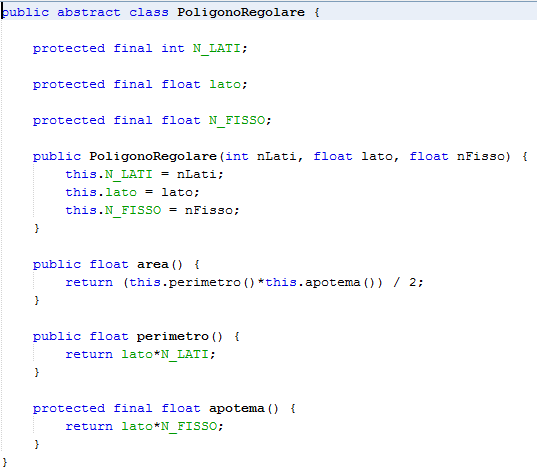
La classe FiguraQuadrato avrà un costruttore con due parametri(float lato, String nome).

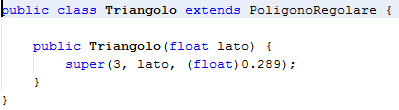
Rappresentazione grafica delle classi (in UML)

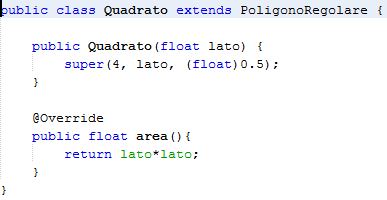


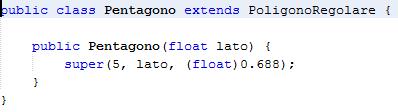
Il codice

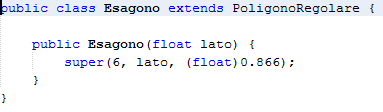


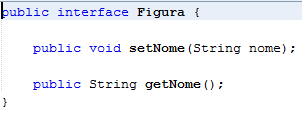


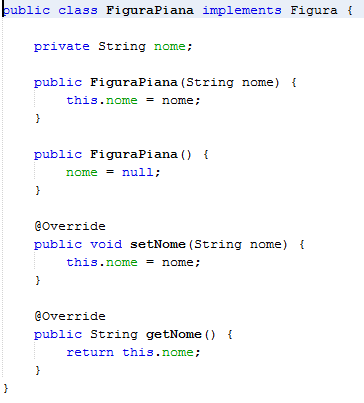


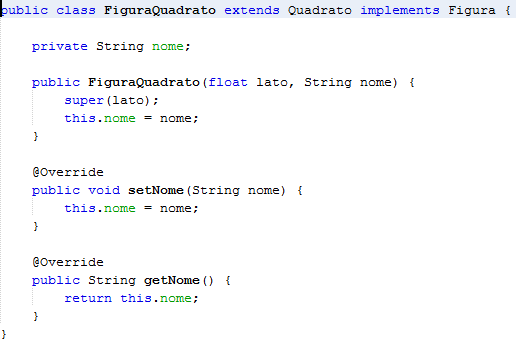












Commenti

1. Una classe che non ridefinisce i metodi astratti della super-classe genera un errore di compilazione poiché questi non possono avere corpo vuoto;
2. Una classe che non ridefinisce i metodi astratti dell’interfaccia implementa genera un errore di compilazione poiché questi non possono avere corpo vuoto;
3. Una classe astratta non può essere istanziata poiché contiene metodi senza corpo, quindi inutilizzabili.
4. Un’interfaccia non può essere istanziata poiché composta metodi senza corpo, quindi inutilizzabili.