

Esercizio

- Progettare una classe **Persona** i cui oggetti rappresentano una persona e ne memorizzano il nome e il cognome
 - La classe espone un metodo **main** che crea un'istanza della Persona
 - La classe espone anche un metodo **stampa** che visualizza nome e cognome della persona

Esercizio

- Progettare una classe **Quadrato**, i cui oggetti sono costruiti con il lato dello stesso
 - La classe è dotata di un metodo **getPerimetro** che restituisce il perimetro
 - E di un metodo **main** che crea un quadrato di lato 4 e ne stampa a video il perimetro

Esercizio

- Progettare una classe **Cerchio** i cui oggetti rappresentano un cerchio
 - La classe è dotata dei metodi **getCirconferenza** e **getArea** (si usi la costante **Math.PI**)
 - La classe espone anche un metodo **main** che crea due cerchi (di raggio 1 e di raggio 5) e ne stampa la circonferenza (per il primo) e l'area (per il secondo)

Esercizio

- Progettare una classe **BarraDiCompletamento** i cui oggetti rappresentano una barra di caricamento
 - Gli oggetti vengono costruiti con la percentuale di partenza
 - La classe espone un metodo **incrementa** che, data una percentuale in input, incrementa la percentuale di partenza con quella fornita in input (ad es. **new**
 BarraDiCompletamento(5).incrementa(10) porta la barra al 15%)
 - Il metodo **toString** dell'oggetto restituisce una stringa contenente la percentuale di completamento arrotondata con **Math.round()**
 - Il metodo **main** che crea una barra di completamento che parte da 0, la incrementa prima di 20 punti percentuale e poi di altri 25 e quindi stampa la rappresentazione stringa della barra

Esercizio

- Progettare una classe **Rettangolo** i cui oggetti rappresentano un rettangolo e sono costruiti a partire dalle coordinate x, y e dalla lunghezza e altezza del rettangolo
- La classe implementa i seguenti metodi:
 - **trasla** che, dati in input due valori x e y, trasla le coordinate del rettangolo dei valori orizzontali e verticali corrispondenti
 - **toString** che restituisce una stringa del tipo “(x1, y1)->(x2, y2)” con i punti degli angoli in alto a sinistra e in basso a destra del rettangolo
- Implementare una classe di test **TestRettangolo** che verifichi il funzionamento della classe **Rettangolo** sul rettangolo in posizione (0, 0) e di lunghezza 20 e altezza 10, traslandolo di 10 verso destra e 5 in basso

Esercizio

Progettare una classe **Colore** i cui oggetti rappresentano un colore in modalità RGB e che sono costruiti a partire da tre valori: R (rosso), G (verde), B (blu), ognuno dei quali ammette un valore intero nell'intervallo 0-255.

- La classe Colore espone anche due costanti BIANCO e NERO
- Fare in modo che ogni Rettangolo (vedere esercizio precedente) abbia associato un Colore di base NERO e che sia possibile impostare il colore di un rettangolo mediante un apposito metodo

Esercizio 1/2

- Progettare una classe **Programmatore** i cui oggetti rappresentano persone che svolgono il lavoro di sviluppo presso un'azienda
- La classe implementa i seguenti metodi:
 - un costruttore con il nome e cognome della persona
 - un metodo **setAzienda** che imposta il nome dell'azienda per cui la persona lavora
 - un metodo **addLinguaggio** che aggiunge un linguaggio di programmazione a quelli inizialmente usati dal programmatore
 - i metodi **getNome**, **getCognome** e **getAzienda** che restituiscono i valori corrispondenti
 - un metodo **getLinguaggi** che restituisce la stringa dei linguaggi (separati da spazio) noti al programmatore

Esercizio 2/2

- Implementare un main all'interno della stessa classe in modo che il seguente codice funzioni correttamente:

```
Programmatore p1 = new Programmatore("Bjarne", "Stroustrup");
Programmatore p2 = new Programmatore("Brian", "Kernighan");
Programmatore p3 = new Programmatore("James", "Gosling");

p1.addLinguaggio("C");
p1.addLinguaggio("C++");
p1.setAzienda("Morgan Stanley");

p2.addLinguaggio("C");
p2.addLinguaggio("AWK");

p3.addLinguaggio("Java");
p3.setAzienda("Oracle");

// stampa: Morgan Stanley
System.out.println(p1.getAzienda());
// stampa: C AWK
System.out.println(p2.getLinguaggi());
```


Esercizio: registratore di cassa

- Progettare una classe che costituisca un modello di registratore di cassa
- La classe deve consentire a un cassiere di **inserire** i **prezzi** degli articoli e, data la **quantità** di **denaro** pagata dal cliente, **calcolare** il **resto** dovuto
- Quale stato (campi) dovete prevedere?
- Quali metodi vi servono?
 - **Registra il prezzo di vendita** per un articolo
 - **Conclude la transazione, calcola il resto** dovuto al cliente nel momento in cui effettua il pagamento e restituendolo in uscita

Esercizio: punti e segmenti

- Progettare una classe **Punto** per la rappresentazione di un punto nello spazio tridimensionale
- E una classe **Segmento** per rappresentare un segmento nello spazio tridimensionale
- Scrivere una **classe di test** che crei:
 - due oggetti della classe Punto con coordinate (1, 3, 8) e (4, 4, 7)
 - un oggetto della classe Segmento che rappresenti il segmento che unisce i due punti di cui sopra