



Esercizi

Su Progettazione e Implementazione di classi

5/11/2004

Laboratorio di Programmazione - Luca Tesei

1

Esercizio 1

- Scrivere un programma che costruisca un oggetto **Rectangle**, lo stampi e successivamente lo sposti (metodo **translate**) e stampi altre tre volte in modo tale che, se ogni volta fosse disegnato, alla fine si avrebbe il seguente disegno:



5/11/2004

Laboratorio di Programmazione - Luca Tesei

2

Esercizio 2

- Consultare le API e trovare tutti i metodi della classe `java.awt.Rectangle`
- Scrivere un programma che crei due rettangoli, li stampi e poi usi il metodo `intersection` per crearne un terzo, risultato dell'intersezione dei due, e stamparlo
- Provare a vedere cosa succede se i due rettangoli hanno intersezione vuota

5/11/2004

Laboratorio di Programmazione - Luca Tesei

3

Esercizio 3

- Aggiungere alla classe **SetNameGreeter** i seguenti metodi
- `/** Stampa un saluto di commiato personalizzato */`
`public void sayGoodbye ()`
- `/** Stampa una frase in cui rifiuta gentilmente di aiutare la persona "name" */`
`public void refuseHelp ()`

5/11/2004

Laboratorio di Programmazione - Luca Tesei

4

Esercizio 4

- Scrivere una classe **SavingsAccount** che abbia le stesse funzionalità di **BankAccount** e in più una variabile istanza **interest** di tipo **double** che rappresenta un tasso di interesse
- Definire i costruttori che si ritengono necessari
- Definire il metodo **addInterest** che, ogni volta che viene chiamato, aggiunge al saldo gli interessi (Su un saldo di 1000 euro, gli interessi del 10% sono di 100 euro)

5/11/2004

Laboratorio di Programmazione - Luca Tesei

5

Esercizio 5

- Implementare una classe **Employee** (dipendente). Ciascun dipendente ha un nome (di tipo **stringa**) e un salario (di tipo **double**).
- Scrivere un costruttore senza parametri e uno con parametri nome e stipendio
- Scrivere metodi `set/get` per leggere e modificare nome e stipendio
- Scrivere un metodo **raiseSalary** (**double byPercent**) che aumenti il salario della percentuale specificata

5/11/2004

Laboratorio di Programmazione - Luca Tesei

6

Esercizio 6

- Implementare una classe **Student** per una applicazione in cui si vogliono conoscere, per ogni studente, solo il nome e un punteggio totale delle risposte a dei quiz
- Fornire un costruttore appropriato
- Fornire il metodo **addQuiz(int score)** per aggiungere il punteggio ottenuto ad un quiz
- Fornire il metodo **getTotalScore()** per conoscere il punteggio totale
- Fornire il metodo **getAverage()** per conoscere la media dei punteggi (Sugg. Per calcolare la media c'è bisogno di tenere traccia del numero dei quiz)

5/11/2004

Laboratorio di Programmazione - Luca Tesi

7

Esercizio 7

- Implementare una classe **Product**. Ciascun prodotto ha un nome e un prezzo come ad esempio **new Product("Tostapane", 29.95)**
- Fornire il metodo **getName()**
- Fornire i metodi set/get per il prezzo
- Scrivere una classe Test che crei due prodotti e ne stampi il nome e il prezzo
- Successivamente diminuire il prezzo di entrambi di 5 euro e ristampare il tutto

5/11/2004

Laboratorio di Programmazione - Luca Tesi

8

Esercizio 8

- Implementare una classe **Circle** che abbia i metodi **getArea()** e **getPerimeter()**
- Nel costruttore indicare il raggio del cerchio
- Si implementi similmente la classe **Square** (quadrato) con gli stessi metodi e indicando, nel costruttore, il lato del quadrato.

5/11/2004

Laboratorio di Programmazione - Luca Tesi

9

Esercizio 9

- Implementare una classe **LattinaDiBirra** con i metodi **getSurfaceArea()** e **getVolume**, indicando nel costruttore il raggio della base e l'altezza della lattina

5/11/2004

Laboratorio di Programmazione - Luca Tesi

10

Esercizio 10

- Implementare una classe **RoachPopulation** (popolazione di scarafaggi)
- Il costruttore riceve la dimensione della popolazione iniziale
- Il metodo **waitForDoubling()** simula un periodo di tempo in cui la popolazione raddoppia
- Il metodo **spray()** simula una spruzzata di insetticida che riduce la popolazione del 10%
- Il metodo **getRoaches()** restituisce il numero attuale di scarafaggi.
- Scrivere una classe Test o Collaudare con BlueJ

5/11/2004

Laboratorio di Programmazione - Luca Tesi

11

Esercizio 11

- Scrivere una classe **RabbitPopulation**
- La popolazione iniziale è sempre di una coppia M/F di conigli
- I conigli sono in grado di accoppiarsi all'età di un mese
- Un mese dopo ogni femmina genera un'altra coppia di conigli
- Ipotizzare che i conigli non muoiano mai e che le femmine generino sempre una coppia M/F ogni mese a partire dal secondo mese

5/11/2004

Laboratorio di Programmazione - Luca Tesi

12

Esercizio 11

- Realizzare un metodo `waitAMonth()` che fa trascorrere un mese
- Realizzare un metodo `getPairs()` che stampa il numero attuale di coppie di conigli
- Collaudare la classe con Bluej
- Suggerimento: usare una variabile istanza per le coppie di conigli neonate e un'altra per quelle che hanno almeno un mese