



Realizzazione di Classi

Laboratorio
01/10/2018



Esempio:

Retribuzione dei dipendenti

○ **Presentazione del problema**

- Modellare un sistema di retribuzione per dipendenti che sono pagati con una tariffa oraria. Il sistema deve riuscire a calcolare la retribuzione di un dipendente sulla base della tariffa oraria e delle ore di lavoro effettuate e deve stampare il nome, le ore e la paga calcolata. I dipendenti che lavorano più di 40 ore ricevono una somma per gli straordinari, pagati una volta e mezzo la loro tariffa salariale normale. Se un dipendente ha 30 o più ore di straordinario nelle ultime due settimane viene emesso un messaggio d'avviso



Retribuzione

○ Scenario d' esempio

Enter employee name: **Gerald Weiss**

Enter employee rate/hour: **20**

Enter Gerald Weiss' s hours for week 1: **30**

Gerald Weiss earned \$600 for week 1

Enter Gerald Weiss' s hours for week 2: **50**

Gerald Weiss earned \$1100 for week 2

Enter Gerald Weiss's hours for week 3: **60**

Gerald Weiss earned \$1400 for week 3

*** Gerald Weiss has worked 30 hours of overtime in the last two weeks.



Esercizio

- Scrivere la classe **SavingsAccount** che è del tutto simile alla classe **BankAccount**, tranne che per una variabile di istanza aggiuntiva, **interestRate** (tasso di interesse annuo). Fornire, oltre a tutti i metodi della classe **BankAccount**
 - Un metodo costruttore che imposti sia il saldo iniziale che il tasso di interesse
 - Un metodo **addInterest** che aggiunge gli interessi trimestrali al conto



Esercizio

- Scrivere una classe, **Name**, che modelli il nome di una persona.
- Comportamento:
 - Ottenere le iniziali come oggetto String
 - Ottenere il nome come oggetto String, nell'ordine nome, cognome
 - Ottenere il nome come oggetto String, nell'ordine cognome, nome
 - Aggiungere o sostituire un titolo (come Signore, Sig.na, Sig.)



Esercizio

- Un distributore di bibite contiene lattine di bibite. Per comprare una bibita il cliente deve inserire un gettone. Quando il gettone è inserito, una lattina cade nel contenitore dove può essere presa dal cliente. Il distributore può essere riempito con ulteriori lattine. Vogliamo essere in grado di determinare quante lattine e gettoni sono presenti nella macchina ad un dato istante.
- Quali metodi sono necessari per la classe **DistributoreBibite**?
- Tradurre le descrizioni informali in Java signatures. (nomi, parametri, e tipi restituiti).
- Quali sono le variabili di istanza che necessitano?
- Implementare e testare la classe ottenuta.



Esercizio

- Esercizio come il precedente rispetto però ad una classe **Studente** dove siamo interessati a registrare i voti degli esami di uno studente e recuperare il voto medio.



Esercizio

- Esercizio come il precedente rispetto però ad una classe **PopolazioneScarafaggi** dove siamo interessati a recuperare il numero totale di scarafaggi e la popolazione può raddoppiare (evoluzione normale) o essere ridotta del 10% (per effetto di pesticidi)



Esercizio 3_4

- Scrivere un programma che prende in input da tastiera 3 numeri in virgola mobile per poi stamparli in ordine crescente.



Esercizio 3_5

- Scrivere un programma che stampi
- “**OK**” se legge “**S**” , “**SI**” , “**OK**” , “**certo**” oppure “**perche’ no?**”.
- “**Fine**” se legge “N” o “No”
- “**Dato non corretto**” altrimenti

- A tal fine scrivere una classe **InputChecker**



Esercizio 3_6

- Scrivere un programma che prende in input da tastiera un certo numero di parole (per esempio 8) e stampa la prima e l'ultima parola secondo l'ordine lessicografico
- Come sopra ignorando la differenza tra maiuscole e minuscole



Esercizio 3_7

- Scrivere un programma che dato il numero di un mese e un anno restituisce il nome del mese, l'anno e il numero di giorni
- Utilizzare
- Classe **Year**
 - con metodo predicativo **isLeapYear()** e
 - metodo **getYear()**
- Classe **Month** con metodi
 - **getDays()** che restituisce i giorni del mese
 - **getMonth()** che restituisce il nome del mese



Esercizio 3_8

- Realizzare una classe per rappresentare una serratura con combinazione di 3 lettere maiuscole
- I metodi sono
 - `void setPosition(String)` // inserisce una lettera del codice
 - `void unlock()` //apre la serratura se il codice passato è quello della serratura
 - `boolean isOpen()` // verifica se la serratura è aperta
 - `void lock()` // chiude la serratura
 - `void newComb(String)` // setta la nuova combinazione se la serratura è aperta



Esercizio 3_9

- Scrivere un programma che valida una nuova password seguendo queste regole:
 - La password deve essere di almeno 8 caratteri
 - La password deve avere almeno un carattere maiuscolo ed uno minuscolo
 - La password deve avere almeno un numero
- Scrivere un programma di test che chiede di inserire una password, poi chiede di confermarla. Se la password non rispetta le regole allora richiede di nuovo l'inserimento.