Îndrumător laborator – LUCRAREA NR. 5

Clase și obiecte

# Exercițiu rezolvat

Creați o clasă Java numită Employee care conține

* Patru membrii privați care rețin un număr al angajatului, numele și prenumele acestuia și salarul angajatului. Numărul asociat angajatului este stocat ca număr întreg.
* Patru metode publice set asociate fiecărui membru privat.
* Patru metode publice get care returnează valorile din cei patru membrii privați.

Salvați fișierul ca Employee.java.

Apoi scrieți o aplicație în care creați cel puțin doi angajați. Creați o metodă care inițializează detaliile angajatului, astfel încât să accepte obiectul Employee nou creat. Salvați aplicația ca DeclareTwoEmployees.java.

public class Employee {

private int empNum;

private String empLastName;

private String empFirstName;

private double empSalary;

//Mutator method (setter)

public void setEmpNum(int emp) {

empNum = emp;

}

//Accessor method (getter)

public int getEmpNum() {

return empNum;

}

public String getEmpLastName() {

return empLastName;

}

public void setEmpLastName(String name) {

empLastName = name;

}

public String getEmpFirstName() {

return empFirstName;

}

public void setEmpFirstName(String name){

empFirstName = name;

}

public double getEmpSalary() {

return empSalary;

}

public void setEmpSalary(double sal){

empSalary = sal;

}

}

import java.util.Scanner; // Import the Scanner class

public class DeclareTwoEmployees {

public static void main(String[] args) {

Employee clerk = new Employee();

Employee driver = new Employee();

clerk = getData(clerk);

driver = getData(driver);

System.out.println("Clerk details:");

System.out.println(clerk.getEmpNum() + " " + clerk.getEmpFirstName() + " " + clerk.getEmpLastName() + " " + clerk.getEmpSalary() + " RON");

System.out.println("Driver details:");

System.out.println(driver.getEmpNum() + " " + driver.getEmpFirstName() + " " + driver.getEmpLastName() + " " + driver.getEmpSalary() + " RON");

}

public static Employee getData(Employee emp){

int num = 0;

String firstName, lastName;

double salary;

Scanner keyboard = new Scanner(System.in);

System.out.print("Enter number >> ");

num = keyboard.nextInt();

//Remove the Enter key that remains in the input buffer after the last numeric entry

keyboard.nextLine();

System.out.print("Enter first name >> ");

firstName = keyboard.nextLine();

System.out.print("Enter last name>> ");

lastName = keyboard.nextLine();

System.out.print("Enter salary >> ");

salary = keyboard.nextDouble();

keyboard.nextLine();

emp.setEmpNum(num);

emp.setEmpFirstName(firstName);

emp.setEmpLastName(lastName);

emp.setEmpSalary(salary);

return emp;

}

}

# Exerciții propuse

1. Jocurile pe calculator conțin adesea diferite personaje sau creaturi. De exemplu, s-ar putea să proiectați un joc în care ființele extraterestre să aibă caracteristici specifice, cum ar fi culoarea, numărul de ochi sau numărul de vieți. Proiectați un personaj pentru un joc, creând o clasă care să conțină cel puțin trei atribute pentru personaj. Includeți metode pentru a obține și seta (getter și setter) fiecare caracteristică a personajului. Salvați fișierul ca MyCharacter.java. Apoi scrieți o aplicație în care creați cel puțin două caractere. La rândul său, transmiteți fiecare caracter la o metodă care afișează atributele personajului. Salvați aplicația ca TwoCharacters.java.
2. Carly’s Catering oferă mese pentru petreceri și evenimente speciale. În lucrarea 2, ați scris o aplicație care solicită utilizatorului numărul de invitați care participă la un eveniment, afișează motto-ul companiei cu un chenar și apoi afișează prețul evenimentului. În lucarea 3 ați modificat programul astfel încât metoda principală să conțină doar trei declarații executabile care apelează fiecare o metodă. Acuma:
3. Creați o clasă care să conțină datele evenimentului pentru Carly's Catering. Clasa conține:

* Două câmpuri public final static care păstrează prețul pe oaspete (35 USD) și valoarea limită pentru un eveniment mare (50 de invitați)
* Trei câmpuri private care rețin numărul evenimentului, numărul de invitați pentru eveniment și prețul. Numărul evenimentului este stocat ca șir, deoarece Carly intenționează să atribuie numere de eveniment, cum ar fi M312.
* Două metode publice set care setează numărul evenimentului (setEventNumber()) și numărul de invitați (setGuests()). Prețul nu are o metodă setată, deoarece metoda setGuests() va calcula prețul ca număr de invitați înmulțit cu prețul pentru fiecare oaspete de fiecare dată când este setat numărul de invitați.
* Trei metode publice get care returnează valorile în cele trei câmpuri nestatice.

Salvați fișierul ca Event.java.

1. Utilizați clasa CarlysEventPriceWithMethods pe care ați creat-o în lucrarea 3 ca punct de plecare pentru un program care demonstrează clasa de eveniment pe care ați creat-o la pasul b, dar efectuați următoarele modificări:

* Aveți deja o metodă care primește un număr de invitați de la un utilizator; acum adăugați o metodă care primește un număr de eveniment. Metoda principală ar trebui să declare un obiect Eveniment, să apeleze la cele două metode de introducere a datelor și să utilizeze valorile returnate pentru a seta câmpurile din obiectul Eveniment.
* Apelați metoda din clasa CarlysEventPriceWithMethods care afișează devizul companiei cu chenarul. Metoda este accesibilă deoarece este publică, dar trebuie să calificați complet numele, deoarece se află într-o altă clasă.
* Revizuiți metoda care afișează detaliile evenimentului, astfel încât să accepte obiectul Eveniment nou creat. Metoda ar trebui să afișeze numărul evenimentului și ar trebui să afișeze în continuare numărul de invitați, prețul pe oaspete, prețul total și dacă evenimentul este un eveniment mare.

Salvați programul ca EventDemo.java.

1. Sammy’s Seashore Supplies închiriază echipamente pe plajă, precum caiace, canoe, scaune de plajă și umbrele pentru turiști. În lucrarea 2, ați scris o aplicație care solicită utilizatorului numărul de minute în care a fost închiriată o piesă de echipament sportiv, afișează motto-ul companiei cu o bordură și afișează prețul pentru închiriere. În lucrarea 3, ați modificat programul astfel încât metoda principală să conțină doar trei declarații executabile care apelează fiecare la o metodă. Acuma:
2. Creați o clasă care să conțină datele de închiriere pentru consumabile pe malul mării Sammy. Clasa conține:

* Două câmpuri public final static care conțin numărul de minute într-o oră și rata de închiriere pe oră (40 USD)
* Patru câmpuri private care rețin un număr de contract, numărul de ore pentru închiriere, numărul de minute pe o oră și prețul. Numărul contractului este stocat ca șir, deoarece Sammy intenționează să atribuie numere de contract, cum ar fi K681.
* Două metode de set publice. Una stabilește numărul contractului (setContractNumber()). Cealaltă poartă numele setHoursAndMinutes() și acceptă numărul de minute pentru închiriere, apoi stabilește orele, minutele suplimentare peste o oră și prețul total. Reamintim de la lucrarea 2 că prețul este de 40 USD pe oră, plus 1 USD pentru fiecare minut în plus.
* Patru metode publice get care returnează valorile din cele patru câmpuri nestatice.

Salvați fișierul ca Rental.java.

1. Utilizați clasa SammysRentalPriceWithMethods pe care ați creat-o în lucrarea 3 ca punct de plecare pentru un program care demonstrează clasa de închiriere pe care ați creat-o la pasul b, dar efectuați următoarele modificări:

* Aveți deja o metodă care primește un număr de minute de la un utilizator; acum adăugați o metodă care primește un număr de contract. Metoda principală ar trebui să declare un obiect de închiriere, să apeleze cele două metode de introducere de date și să utilizeze valorile returnate pentru a seta câmpurile din obiectul de închiriere.
* Din clasa SammysRentalPriceWithMethods, apelați la metoda RentalDemo care afișează devizul companiei cu granița. Metoda este accesibilă deoarece este publică, dar trebuie să calificați complet numele, deoarece se află într-o altă clasă.
* Revizuiți metoda care afișează detaliile de închiriere, astfel încât să accepte obiectul pentru închiriat nou creat. Metoda ar trebui să afișeze numărul contractului și ar trebui să afișeze orele și minutele, tariful pe oră și prețul total.

Salvați programul ca RentalDemo.java.