

Zadanie Interfejsy – program „Firma”

1. Stwórz Program Firma. Program powinien zawierać następujące elementy:
 - a. Interfejs Pracownik wraz z metodami:
 - i. void przyznajPremie(double kwota)
 - ii. double obliczNetto()
 - iii. void dodajOstrzezenie(String tresc)
 - iv. void przyznajKare(double kwota)
 - b. Interfejs Kierowca wraz z metodami:
 - i. void nadajPojazd(String nazwa, int numer)
 - ii. boolean sprawdzBadanieTechniczne()
 - iii. void wyswietlInformacje()
 - c. Interfejs PracownikHali wraz z metodami:
 - i. void nadajStanowsiko()
 - ii. boolean sprawdzBadaniaLekarskie()
 - iii. void sprawdzAkord()
 - iv. void WyswietlInfo
 - d. Klasy Sprzątaczką implementującej interfejs Pracownik. Klasa powinna posiadać ponadto pola:
 - i. private String imie
 - ii. private String nazwisko
 - iii. private String adres
 - iv. private double wypłata
 - v. prywatną listę z ostrzeżeniami
 - vi. Konstruktor
 - vii. Klasa powinna w sensowny sposób implementować metody z interfejsu
 - e. Klasy OperatorPily implementującej interfejsy Pracownik i PracownikHali. Klasa powinna posiadać pola:
 - i. private String imie
 - ii. private String nazwisko
 - iii. private String adres
 - iv. private double wypłata
 - v. String rodzajMaszyny
 - vi. int numerStanowiska

Zadanie Interfejsy – program „Firma”

- vii. boolean badanieLekarskie
 - viii. prywatną listę z ostrzeżeniami
 - ix. Konstruktor
 - x. Klasa powinna w sensowny sposób implementować metody z interfejsu
- f. Klasy KierowcaDostawczaka implementującej interfejsy Pracownik oraz Kierowca. Zaimplementuj klasę podobnie do klasy Sprzątaczką i OperatorPily
- g. Klasy klasy abstrakcyjnej Kierownictwo, posiadającej pola:
- i. String imie
 - ii. String nazwisko
 - iii. String adres
 - iv. double wypłata
 - v. abstrakcyjną metodę ustalWypłatę()
 - vi. abstrakcyjną metodę wyswietlInfo()
 - vii. abstrakcyjną metodę liczPremie()
- h. Klasy KierownikMontażu dziedziczącej po klasie Kierownictwo, który implementuje metody abstrakcyjne oraz dodatkowo posiada pola:
- i. boolean wyrobionaNorma
 - ii. int godzinyPonadNorme
 - iii. metoda liczPremie(), według godzin ponad normę oblicza dodatkową premię i dodaje do wypłaty
- i. Klasy KierownikMarketingu dziedziczącej po klasie Kierownictwo, który implementuje metody abstrakcyjne oraz dodatkowo posiada pola:
- i. Boolean Samochod
 - ii. String numerRejestracyjny
 - iii. String numerSluzbowy
 - iv. Listę obiektów klasy Klient
 - v. Metodę nadajAuto(String numerRejestracyjny)
 - vi. Metodę wyswietlInformacje() która działa w różny sposób, zależnie od posiadania auta służbowego
 - vii. Metodę dodajKlienta()
 - viii. Metodę usunKlienta()
 - ix. metoda liczPremie(), która według ilości klientów oblicza premię i dodaje do wypłaty
- j. BONUS: Stwórz klasę Firma obsługującą przynajmniej część z funkcji.