Readme

Activitatea unei companii de transport

**Clase:**

Fiecare clasa contine getter si setter pentru fiecare variabila si metoda toString.

1. Angajat

Contine date generale despre angajati: nume, numarul de ore suplimentare din luna in curs si jobul pe care il ocupa (jobul este un obiect al clasei Job).

1. Cauciuc

In realitate o companie de transport trebuie sa se ocupe cu grija de starea cauciucurilor masinilor si de asemena trebuie sa le schimbe in functie de sezon. De aceea am o clasa cauciuc cu date specifice: producator, indice de uzura, dimensiuni, cod unic etc.

1. Factura

Fiecare firma trebuie sa tina o evidenta a facturilor, de aceea am obiectul facture care contine un cod unic, un numar de facture, ruta pe care a fost facut transportul si masina folosita. Contine si totalul care se calculeaza automat in functie de tariful specific masinii si distanta parcursa.

1. Job

Contine date standard despre meseriile existente in cadrul firmei: numele jobului, numarul de ore minim pe luna, salariul si comisionul pentru orele suplimentare.

1. Manager

Clasa manager este extinsa din clasa angajat. In plus contine date specific si de interes numai pentru manager: cod bancar, intalniri de afaceri, etc.

1. Masina

E important sa avem o evidenta clara a cat mai multor caracteristici ale masinilor pe care le folosim. Clasa contine date despre marca, caracteristici tehnice ale rotilor, numarul de km, termenul de valabilitate al reviziei(ITP) etc.

1. Oras

Compania de transport asigura transportul intre doar cateva orase. Pentru acestea e important sa cunoastem de exemplu indicele minim de poluare admis pentru o masina (Euro4) ca sa stim ce masini putem sa alocam pentru ruta respective.

1. Revizie

Revizia contine termenul de valabilitate: an,luna, zi. Fiecarei masini ii apartine o revizie.

1. Ruta

Ruta exista intre 2 orase (nu avem rute cu escala), contine informatii despre durata si distanta.

**Repository**:

Pentru fiecare clasa am si o clasa NumeRepository in care am o lista cu fiecare intrare (de exemplu am o lista cu masinile firmei) si metode specific pentru acestea (stergere, adaugare de element nou, o metoda de obtinere a informatiei din lista respective ex: getDateAngajati si o metoda de cautare dupa un identificator unic si cat mai natural(de exemplu pot cauta o masina dupa numarul de inmatriculare care este uni, in loc sa folosesc numarul de sasiu care in realitate ar fi mai greu de utilizat))

**Service**:

Aici am doar functiile pe care vreau ca utilizatorul sa le foloseasca de fapt: adaugari de obiecte, stergeri, cautari(interogari), schimbarea unor anumiti parametrii (poate schimba cauciucurile unei masini, dar nu poate schimba seria de sasiu). Numarul de metode din Service o sa creasc pe masura ce identific si alte nevoi ale aplicatiei.

**Main:**

Aici am folosit cateva din actiunile din service pentru a observa ca functioneaza corect.

//1.Actualizam gradul de uzura al cauciucului cu codul 101.

//2.Adaugam un nou cauciuc

//3.Au fost cumparate cauciucuri noi in locul celor ce aveau gradul de uzura >=4. Sa se actualizeze indicii de uzura.  
//4.Stergem cauciucul adaugat la 2) folosind cautarea dupa cod

//5.Sa se afiseze bonusul pentru orele suplimentare pentru fiecare angajat

//6. Sa se afiseze toate facturile cu valoare mai mare de 15

//7.Sa se modifice comisionul pentru postul de sofer si numarul de ore minim pentru postul de mecanic si sa se sorteze descrescator lista dupa bonus apoi dupa ore

//8.Sa se afiseze masinile cu mai putini de 150000 km

//9.Sa se afiseze toate masinile care pot circula in Viena luan in considerare nivelul de poluare admis in oras

//10. Sa se afiseze toate rutele disponibile care incep sau se termina in Bucuresti

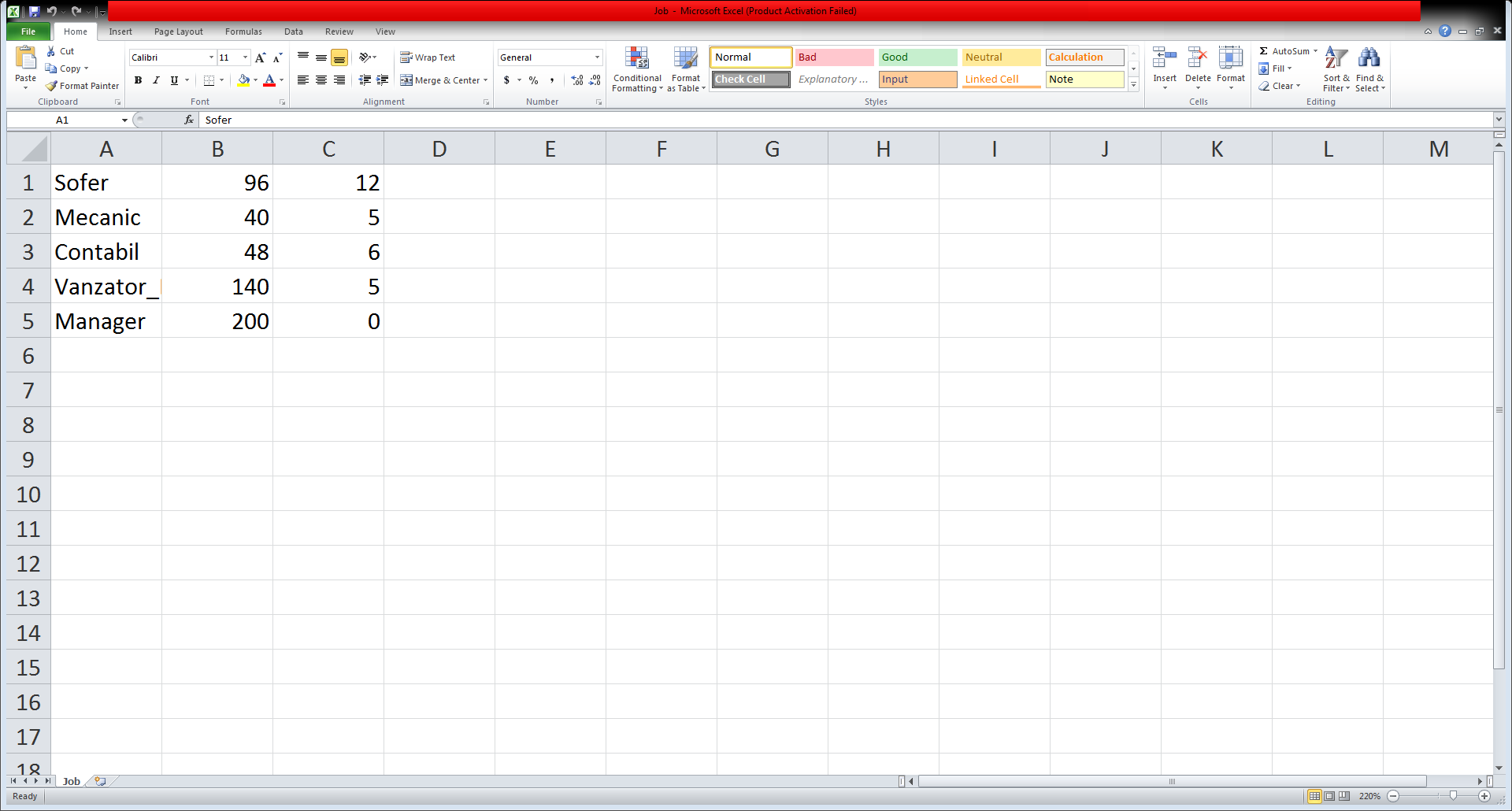
//11. Sa se afiseze datele specifice postului de manager (nu datele despre angajatul care e manager)

**Etapa II:**

Am utilizat fisiere pentru urmatoarele 4 clase: Job, Revizie, Oras, Cauciuc.

Exemplu:

Citesc datele din fisierul cu nume “Job.csv” si afisez datele in “Job2.csv” ca sa putem observa diferentele. In mod normal as afisa tot in “Job.csv” pentru a avea datele actualizate urmatoarea data cand deschid aplicatia si incarc datele. De fiecare data cand se intampla o modificare noua fisierul este reactualizat.

****

Job2.csv

Job.csv

