



Mission Control – Apollo1053 has landed!



Introducere

Bun venit la Institutul International de Cercetare Spațială! Ca inginer software în cadrul institutului, vei contribui la dezvoltare unei soluții software pentru gestionarea viitoarelor misiuni în spațiu.

Institutul nostru deține mai multe tipuri de nave spațiale `Spacecraft`. Misiunile pentru care ai fost alocat/ă constau în manipularea flotei actuale de nave tip `Lander`. În cadrul acestor misiuni, fiecare lander va extrage temperaturile înregistrate pe planeta pe care se află.

Scopul tău este să creezi o soluție care să ajute la gestionarea întregii flote de landere, dar și la stocarea și manipularea în timp real a datelor colectate.



Descrierea mediului de lucru

Good news: există deja un proiect Java creat în vederea implementării acestei soluții, însă acest proiect necesită ajustări. Tot ce au reușit colegii să facă până acum a fost să definească cele 2 clase `Spacecraft` și `Lander` și să extragă atributele necesare fiecărei clase. Astfel, `Spacecraft` are ca atribute numele și greutatea (exprimată în kilograme) unei nave, iar `Lander` conține numele site-ului unde landerul este programat să aterizeze, dar și un contor care va monitoriza dacă landerul a aterizat și, totodată, un vector cu temperaturile înregistrate pe parcursul misiunii.

Colegii noștri de la departamentul de sistem ne-au pregătit un fișier `landers.csv` cu date despre flota actuală de landere a institutului.



Cerințe software

Pentru a duce la bun sfârșit această aplicație software, este nevoie ca următoarele cerințe să fie implementate:

- Pentru început, vrem să ne asigurăm că soluția noastră are o bază solidă în ceea ce privește conceptele de OOP. Ținând cont că `Lander` este un tip de `Spacecraft`, implementează metodele necesare pentru a te asigura că proiectul nostru întrunește principiile de encapsulation și inheritance. Nu uita să testezi aceste metode în `main` !
- Folosindu-te de fișierul `landers.csv`, citește datele și crează obiectele de tip `Lander`
 - Hint: considerăm că landerele nu au aterizat încă
- Implementează o metodă abstractă în clasa `Spacecraft` numită `land()` care va servi la schimbarea valorii câmpului `hasLanded` al clasei `Lander`
- După o discuție avută cu celelalte departamente, este nevoie să adaugi o modalitate prin care se poate calcula temperatura medie înregistrată de un lander până în momentul curent. Avem nevoie de o soluție care îi va face pe inginerii software să implementeze această funcționalitate și pentru celelalte tipuri de nave ale institutului (un fel de "contract"). Așadar, pentru a face acest lucru, folosește-te de o altă modalitate de abstractizare decât cea folosită la punctul anterior 😊

- Pentru landerele care au aterizat, scrie într-un fișier .txt date referitoare la statusul misiunii lor, urmând structura de mai jos:

| Name LandingSite AverageTemperature

Ține cont că fiecare informație aparținând unui lander trebuie să fie scrise pe același rând.

- Opțional: încearcă să adaugi și vectorul de temperaturi înregistrate până acum de fiecare lander, conform structurii:

| Name LandingSite AverageTemperature TemperatureReadings

- Implementează metodele de serializare și deserializare ale obiectelor de tip `Lander`. Este suficient să serializezi/deserializezi un singur obiect.
- Utilizând concepte de JCF, creează o structură de date care să stocheze obiectele de tip `Lander` în ordine crescătoare în funcție de greutate. Scrie elementele acestei colecții într-un fișier binar.
- Folosindu-te de fișierul binar creat la punctul anterior, afișează al doilea cel mai ușor `Lander`.
- Înainte de a preda proiectul, mai rămâne de implementat o singură funcționalitate. S-a stabilit că este nevoie de o modalitate prin care să gestionăm câte misiuni active avem pe fiecare corp ceresc. Pentru a ne ajuta, echipa de sistem ne-a pus la dispoziție fișierul `landers.csv` unde ne sunt oferite alocările site-urilor pe fiecare corp ceresc. Odată citite aceste alocări și finalizată analiza de date, implementează o modalitate prin care acest sumar să fie exportat într-un fișier care va fi folosit în următorul comunicat de presă privind statusul misiunilor.
 - Hint: pentru a facilita implementarea, te poți folosi de una dintre colecțiile din JCF și de formula de mai jos:

| $\text{cod} = \text{cod} + 31 * \text{litera} + \text{avgTemp}$

Spor la treabă! 🚀