**HTTP. Rest APIs. Requests**

**Folosim** [**https://jsonplaceholder.typicode.com/guide/**](https://jsonplaceholder.typicode.com/guide/) **Toate requesturile se vor face prima data in Postman (pentru verificare), iar apoi folosind libraria requests din Python.**

**Structura datelor este următoarea:**

**- fiecare post este deținut de un user, și poate avea mai multe comments**

**- fiecare album este deținut de un user, și poate avea mai multe photos**

**- fiecare todo este detinut de un user.**

**1. Alege un user din lista de useri predefiniti. Pentru acesta, fa requesturile necesare pentru a afișa următoarele informații:**

**-> lista de posts**

**-> lista de albume**

**-> lista de todos**

**Pentru a menține output-ul la un nivel acceptabil, afișează la fiecare dintre aceste liste doar informații despre primele 3 obiecte, iar apoi afiseaza "+x more posts/albums/todos", unde x este numărul de obiecte rămase.**

**2. Alege un post, și afișează lista de comentarii. Alege un album, si afiseaza lista de photos.**

**3. Creeaza un post nou (atentie, acesta NU va fi salvat, dar putem analiza răspunsul pentru a vedea cum ar arata în viitor dacă ar fi fost salvat). Încearcă să-l creezi cu si fără id. Ce observi?**

**4. Alege un post existent și realizează operațiunile de PUT si PATCH (atentie, modificările NU vor fi salvate, analizăm doar răspunsul). Ce observi ca este diferit între cele 2 metode?**

**5. Folosind query parameters, filtrează lista de todos pentru userul ales astfel incat sa afisezi doar todos care nu sunt completed.**

**6. Alege un album, și ia pozele din acesta în 2 moduri diferite (o data cu nested resource, o data folosind query params). Verifica dacă exista vreo diferenta intre cele 2 rezultate.**

**Folosim Simple Books API, descris aici :** [**https://github.com/vdespa/introduction-to-postman-course/blob/main/simple-books-api.md**](https://github.com/vdespa/introduction-to-postman-course/blob/main/simple-books-api.md)

**Toata rezolvarea se va face într-o clasa numita Books API Client. Pentru testare se va crea un obiect din aceasta clasa și se vor apela metodele implementate.**

**1. Folosind endpoint-ul de authentication, genereaza un access token (fa asta in constructor, client name si email ar trebui sa fie atribute).**

**2. Adaugă o metoda prin care poți vedea toate comenzile.**

**3. Adaugă o metoda prin care poți vedea toate cărțile.**

**4. Adaugă o metoda prin care poți posta o comanda.**

**5. Adaugă o metoda prin care poți șterge o comanda.**