

1. Structura in care sunt stocate datele elevilor s-a modificat și arata cam asa:

```
st = {  
  23: {'fname': 'Dana', 'lname': 'Popescu', 'class': '9b'},  
  2: {'fname': 'Ion', 'lname': 'Pop', 'class': '10c'},  
  31: {'fname': 'Gelu', 'lname': 'Ionescu', 'class': '10c'},  
  15: {'fname': 'Geta', 'lname': 'Ionescu', 'class': '9b'},  
}
```

Pagina cu toti elevii (ruta /students) trebuie sa arate asa ->

Numele complete ale elevilor sunt link-uri spre pagina /student/id_student

Fast Track IT

Date elev:

Nume	Ionescu
Prenume	Gelu
Clasa	10c

[Lista elevi](#)

<- Pagina cu datele elevului (ruta /student/id_student) trebuie sa arate asa

In plus, daca id-ul dat ca parametru nu se gaseste printre id-urile din structura elevilor atunci pagina sa contina un mesaj de eroare cu rosu gen "Elev inexistent".

Fast Track IT

Lista elevi

Nume	Clasa
Popescu Dana	9b
Pop Ion	10c
Ionescu Gelu	10c
Ionescu Geta	9b

[Main page](#)

2. Clasa devine și ea link in pagina cu elevii. Daca se da click pe clasa se va afisa o pagina cu toti elevii din clasa respectiva. Generati o noua ruta /class/<class_name> sau asa ceva, deja stiti.. :-) In cazul in care se apeleaza aceasta ruta cu un nume de clasa care nu exista, sa se afiseze in pagina un mesaj cu rosu cum ca acea clasa nu exista.

Fast Track IT

Lista elevi

Nume	Clasa
Popescu Dana	9b
Pop Ion	10c
Ionescu Gelu	10c
Ionescu Geta	9b

[Main page](#)

Fast Track IT

Elevii clasei 10c:

Nume
Pop Ion
Ionescu Gelu

[Lista elevi](#)

3. Cautarea din pagina cu elevii sa functioneze acum in felul urmator: se cauta atat in nume cat și in prenumele elevului și in oricare dintre aceste campuri daca se gaseste valoarea data la cautare, elevul respectiv va fi afisat.

4. Structura cu elevii de la punctul 1 va contine acum și notele fiecarui elev. Adaugati o cheie 'grades' la fiecare student și aceasta va contine o lista de note gen [4, 6, 7] etc..

In pagina de elev vor mai aparea doua informatii in tabela cu datele elevului și anume: lista notelor elevului respectiv media notelor sale.

5. Includem și bazele de date in ecuatie. Practic cam ce ati rezolvat la problemele anterioare sa rezolvam și la asta dar datele sa fie stocate intr-o baza de date. Va las pe voi sa faceti integrarea cu baza de date. Revedeti laboratoarele unde am discutat de baze de date și cum le accesam din Python. Scoala pentru care dezvoltam solutia vrea sa aiba urmatoarele pagini:

- pagina unde sunt listate toate clasele, fiecare clasa este un link. La click pe link se deschide pagina cu clasa respectiva.
- pagina cu clasa, unde sunt listati in ordine alfabetica toti elevii. Numele sunt link-uri. La click pe link se deschide pagina cu elevul
- pagina cu elevul, care sa contina numele, clasa și notele lui
- o functionalitate de search. Poate fi integrata in pagina cu clasele sau sa fie de sine statatoare. Sa se poata cauta dupa nume și prenume și sa se afiseze toti elevii care corespund și ei sa fie link-uri, adica daca dau click pe el sa ma duca la pagina lui.

Incercati sa faceti aplicatia sa arate cat mai bine, sa fie cat mai usor de utilizat. Ganditi-va la cei care o vor utiliza. Trebuie sa le fie de folos și sa se descurce usor cu ea. Și de asemenea pentru voi e util sa fie codul cat mai organizat, fisierele separate și organizate frumos (python, js, css, html, template-uri, fisiere statice, layouts, etc..)