

Dezvoltarea Aplicatiilor Web-Anul 2

Laborator 2

Exercitiul 1

Sa se creeze Controllerul **ExampleController** (Vezi Curs 2) si sa se rezolve urmatoarele cerinte, creand metodele aferente in acest Controller (Actions):

1. Sa se adauge o ruta numita "Concatenare" care sa raspunda prin **/concatenare** urmata de 2 parametri de tip String. Sa se afiseze valorile lor concatenate.

In RouteConfig.cs

```
routes.MapRoute(
    name: "Concatenare",
    url: "concatenare/{param1}/{param2}",
    defaults: new { controller = "Example", action = "Concatenare",
param1 = UrlParameter.Optional, param2 = UrlParameter.Optional }
);
```

In ExampleController.cs

```
public string Concatenare(string param1, string param2)
{
    return param1 + " " + param2;
}
```

2. Sa se adauge o ruta numita "Produs" care primeste 2 parametri de tip intreg. Al doilea parametru este optional. Sa se afiseze produsul celor 2 parametri. Daca al doilea parametru lipseste se va afisa mesajul "Introduceti ambii parametri".
3. Sa se adauge o ruta numita "Operatie" care primeste 3 parametri astfel: 2 de tip intreg si unul de tip string. Parametrul de tip string reprezinta o operatie (plus, minus, ori, div) efectuata intre parametrii de tip intreg.

Sa se afiseze pe ecran parametrii impreuna cu operatia, cat si rezultatul acestora. In cazul in care un parametru lipseste se va afisa pe ecran mesajul "Va rog introduceti parametrul 1/2/3" – in functie de parametrul care lipseste. De asemenea, parametrii sunt optionali.

Exercitiul 2

Presupunem ca avem o aplicatie in care exista resursa Student. Sa se adauge un Controller – **StudentController** si toate rutele asociate operatiilor de tip C.R.U.D. – CREATE, READ, UPDATE, DELETE (de exemplu, in Controller o sa avem urmatoarele metode: Index – pentru afisarea tuturor studentilor, Show – afisarea unui singur student, in functie de id-ul acestuia, Create – crearea unui nou student, Edit – editarea unui singur student, Delete – stergerea unui singur student) asupra acestei resurse. Rutele trebuie sa contina parametrii necesari operatiilor si de asemenea, sa nu existe conflict intre rute.

Exemplu:

In RouteConfig.cs

```
routes.MapRoute(  
    name: "StudentIndex",  
    url: "students/all",  
    defaults: new { controller = "Student", action = "Index" }  
);
```

In StudentController.cs

```
// Lista tuturor studentilor  
public string Index()  
{  
    return "Lista tuturor studentilor";  
}
```

Exercitiul 3

Sa se creeze Controllerul **SearchController**. In cadrul acestuia sa se adauge urmatoarele metode: **NumarTelefon, CNP**. Fiecare metoda reprezinta o pagina de cautare care primeste un parametru dupa care se face cautarea. Adaugati metodele impreuna cu parametrii lor, cat si rutele asociate impreuna cu urmatoarele constrangeri si afisati valorile parametrilor in pagina:

- Ruta pentru cautarea numarului de telefon (**/search/telefon/{telefon}**) va primi un singur parametru reprezentat de un sir de caractere de lungime = 10 caractere; toate caracterele sunt cifre; sirul incepe cu caracterul 0; dupa caracterul 0 trebuie sa fie obligatoriu inca 9 cifre (<https://regexr.com/>). Ce observati in cazul in care numarul de telefon lipseste din request sau in cazul in care acesta are lungimi diferite?
- *** OBS:** Pentru aceeasi metoda din Controller sa se adauge o noua ruta care va primi un singur parametru, reprezentat de un sir de caractere de orice lungime; toate caracterele sunt cifre. Cum putem rescrie expresia regulata, astfel incat validările parametrului (lipsa numarului de telefon; dimensiunea acestuia) sa se intample la nivel de Controller?
- Ruta CNP (**/search/cnp/{cnp}**) va primi un singur parametru reprezentat de un sir de caractere de lungime = 13; toate caracterele sunt cifre; sirul incepe cu 1,2,5 sau 6; (<https://regexr.com/>). Ce observati?