

How to specify mapping rule when names of properties differ

Asked 6 years, 6 months ago Active 1 month ago Viewed 49k times



I am a newbie to the Automapper framework. I have a domain class and a DTO class as follows:

128



19

```
public class Employee
{
    public long Id {get;set;}
    public string Name {get;set;}
    public string Phone {get;set;}
    public string Fax {get;set;}
    public DateTime DateOfBirth {get;set;}
}

public class EmployeeDto
{
    public long Id {get;set;}
    public string FullName {get;set;}
    public DateTime DateOfBirth {get;set;}
}
```

Note: The name of property **"Name"** of `Employee` class is not the same as that of property **"FullName"** of `EmployeeDto` class.

And here's the code to map the `Employee` object to `EmployeeDto` :

```
Mapper.CreateMap<Employee, EmployeeDto>(); // code Line (***)
EmployeeDto dto = Mapper.Map<Employee, EmployeeDto>(employee);
```

My question is: If I want to map `Employee` (source class) to `EmployeeDto` (destination class), how can I specify the mapping rule? In other words, how should I do more with code line (***) above?

c#

automapper

edited Mar 16 '16 at 17:41

Anderson Pimental

asked Feb 8 '13 at 17:00

Thomas Benz

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

2 Answers



Never mind, I myself found a solution:

259



```
Mapper.CreateMap<Employee, EmployeeDto>()
    .ForMember(dest => dest.FullName, opt => opt.MapFrom(src => src.Name));
```

answered Feb 8 '13 at 17:43



[Thomas.Benz](#)

4,740 8 30 52

- 1 Is there a way to specify it globally for any class, something like prefix/suffix thing? For example, I have a bunch of classes from a library that have "Localizations" property each. And I want to map them to classes where corresponding property is called "LocalizedName". Is my only option is to add ForMember to each mapping configuration? – [NickAb](#) May 20 '16 at 14:43
- 2 there is no CreateMap method in Mapper class :(– [Navid_pdp11](#) Jul 20 '16 at 8:01
- 3 @Navid_pdp11 yes there is. You need to have the static class Mapper not an instance – [Jordy van Eijk](#) Jul 28 '16 at 11:42
- @JordyvanEijk What version are you using that has a static "CreateMap" method? In 5.1.1.0 there's no such thing – [Matt Thomas](#) Oct 6 '16 at 19:42
- 6 @MattThomas After taking a look at the github of automapper i see that as of version 5.0 CreateMap is obsolete and they say you need to use the MapperConfiguration or Mapper.Initialize. – [Jordy van Eijk](#) Oct 10 '16 at 12:00



Really old thread, but just to roll the comments above into an updated approach using AutoMapper 8.1+...

0



```
var mapConfig = new MapperConfiguration(
    cfg => cfg.CreateMap<Employee, EmployeeDto>()
        .ForMember(dest => dest.FullName, opt => opt.MapFrom(src => src.Name))
);
```

Then you would build the mapper using the mapConfig:

```
var mapper = mapConfig.CreateMapper();
```

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

