Swagger

Zawartość

[Documentation of your first API 2](#_Toc67240565)

[Add Swagger to project 2](#_Toc67240566)

[Create Documentation in XML 3](#_Toc67240567)

[Turn off a Warnings 7](#_Toc67240568)

[More information to Swagger Doc 8](#_Toc67240569)

[Overriding convention in OpenAPI 9](#_Toc67240570)

[Responses 9](#_Toc67240571)

[API Analyzer 10](#_Toc67240572)

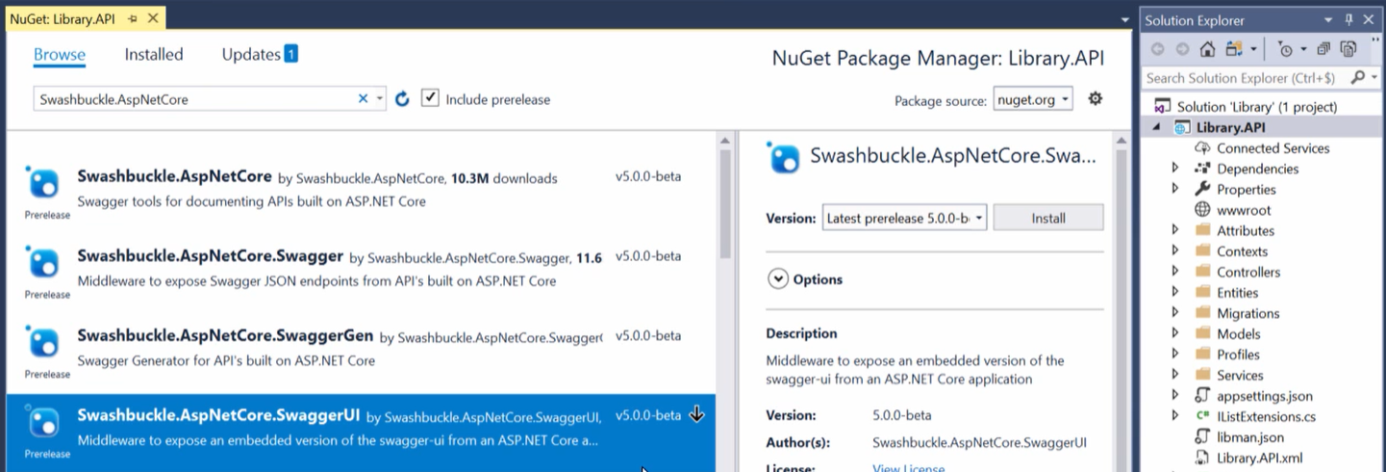
[Working with API Convention 11](#_Toc67240573)

[Create custom convention 12](#_Toc67240574)

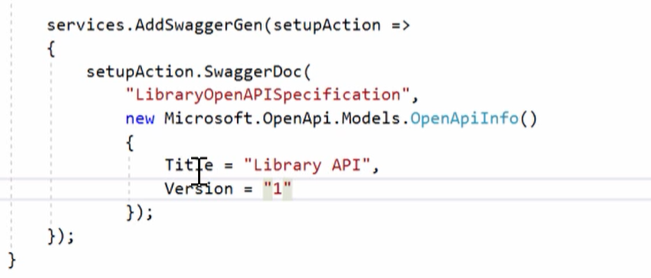
[Content negotiations 14](#_Toc67240575)

# Documentation of your first API

## Add Swagger to project



Add first package or 3 package under. First one consist of three of them



Register service

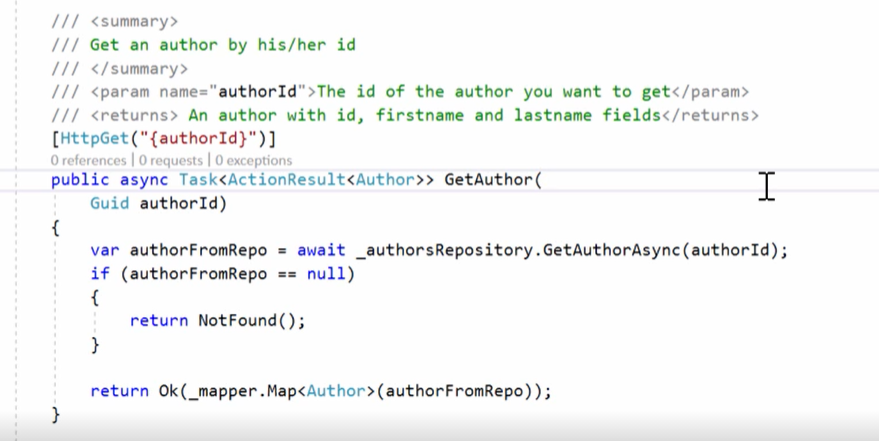


And to pipeline

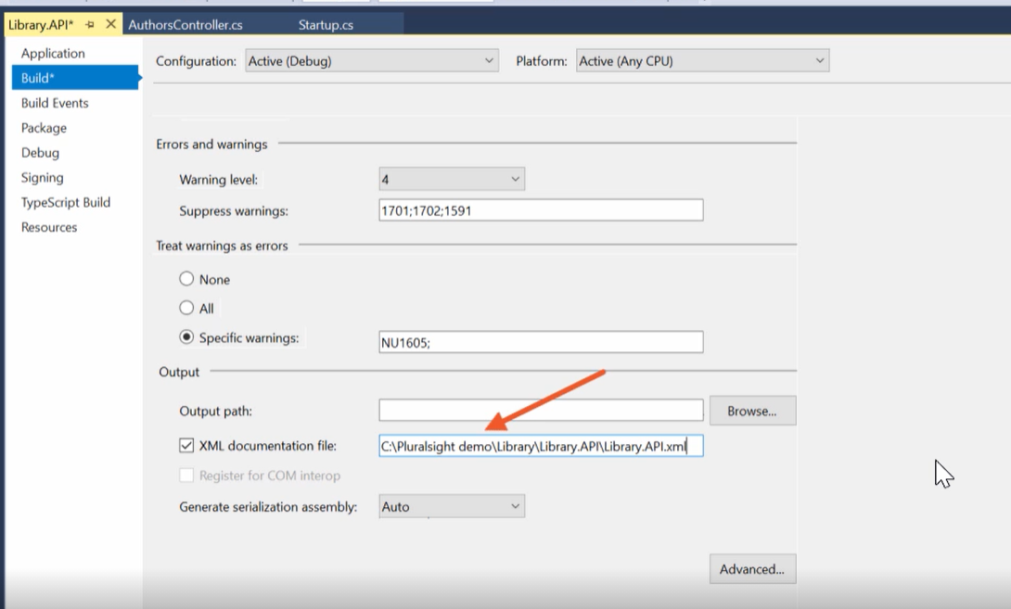


Add swaggerUI. After swagger in pipeline add next item swaggerUI and on swaggerEnpoint set two values endpoint and swagger documentation name. Remember case sensitive!!!

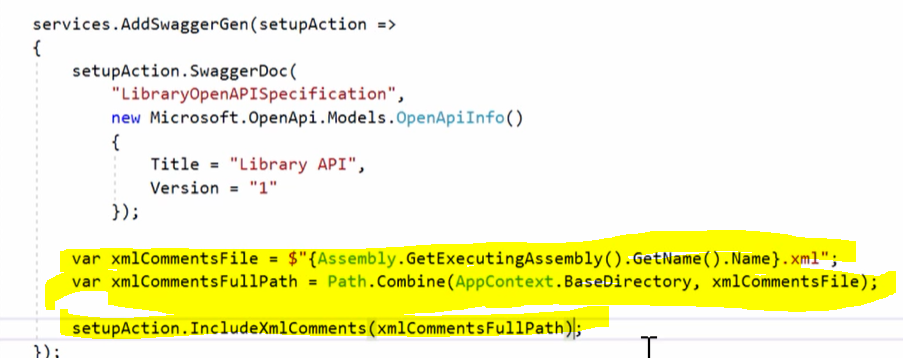
## Create Documentation in XML



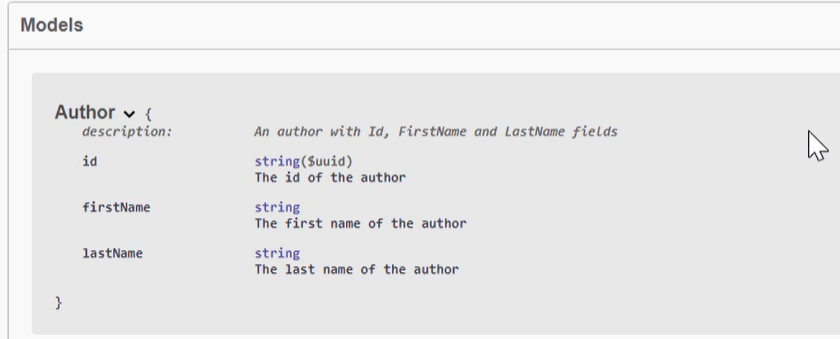
Add summary



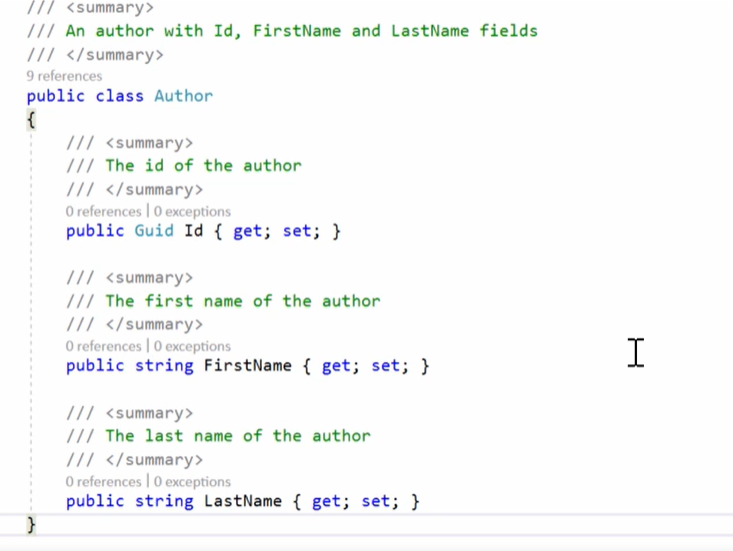
In project properties set xml documentation file.



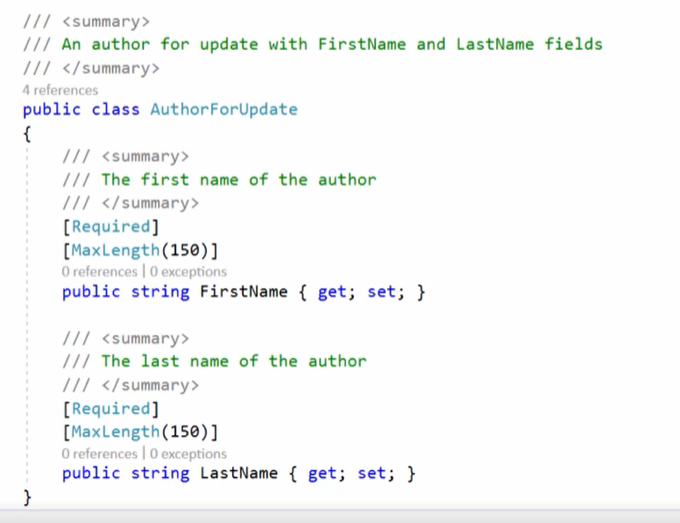
Add setup to swaggerGen.



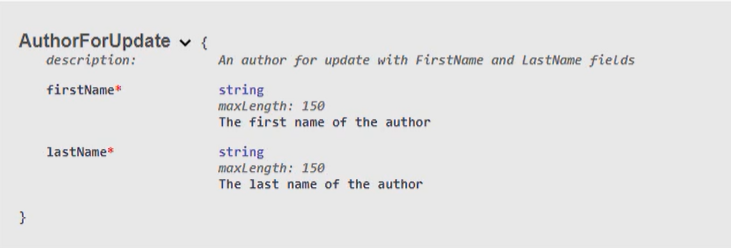
The same thing can be did which each parameters in classes.



Only add some comments.



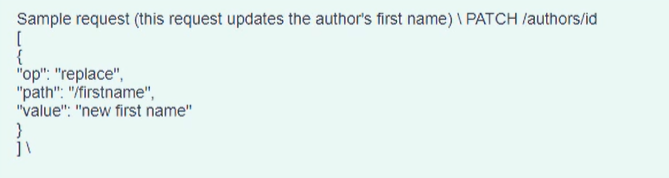
The same action can be commited in DTO classes. Add some comments.



And admire your result in swagger UI.



Remarks is useful to add sample request.

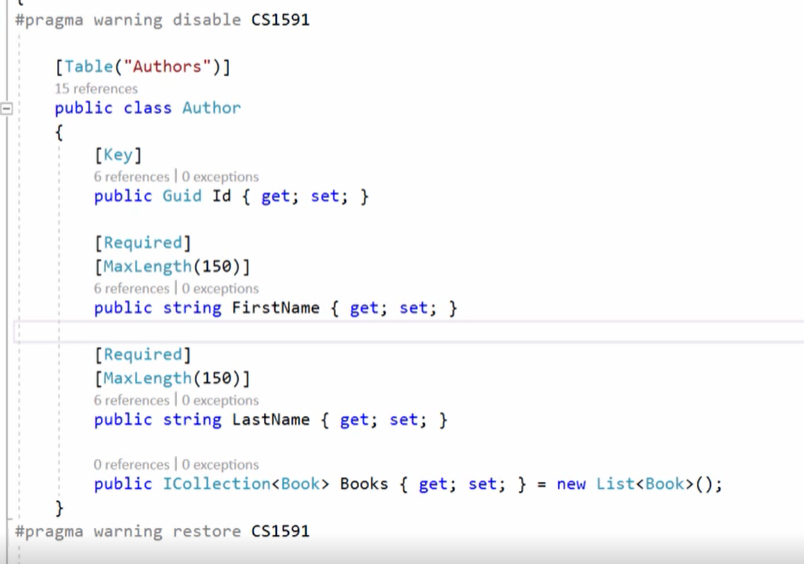


Backslash break a line

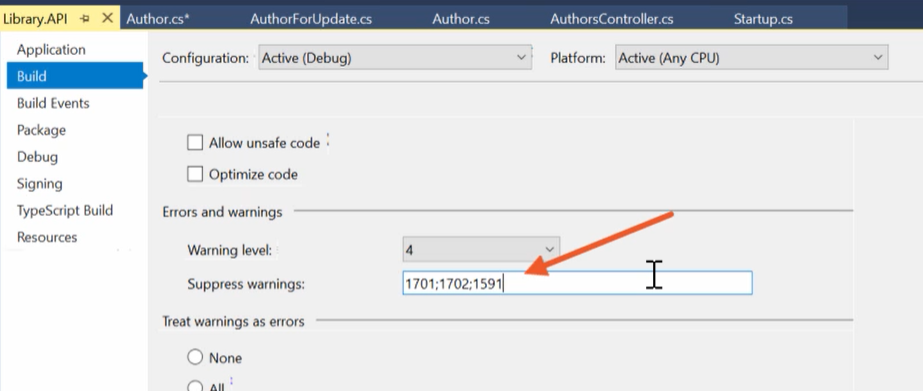


Without look like this

## Turn off a Warnings

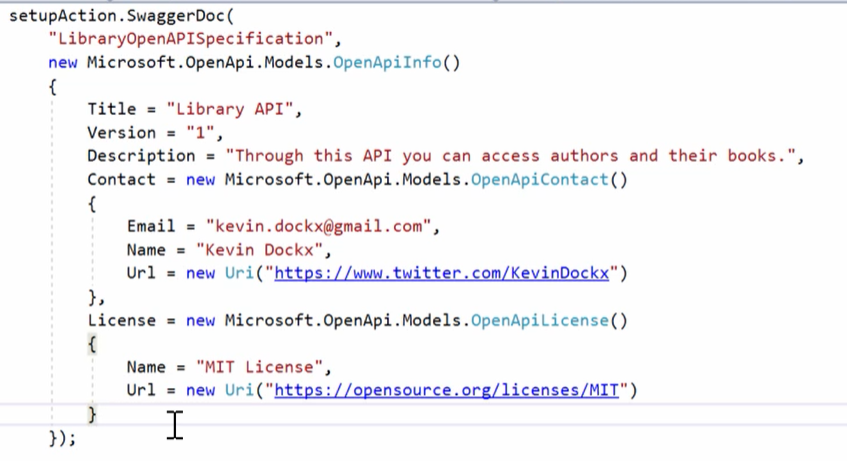


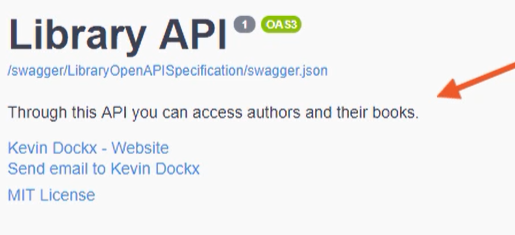
First way make everywhere sectors with turned off warning. Yes the same way is add all properties action and classes comments.



From properties we can set warning 1591 as turned off and everything will be fine.

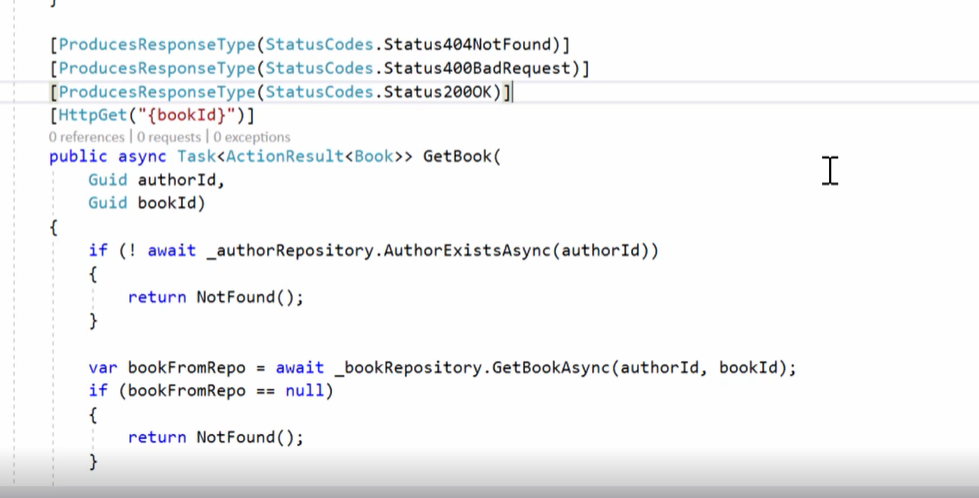
## More information to Swagger Doc

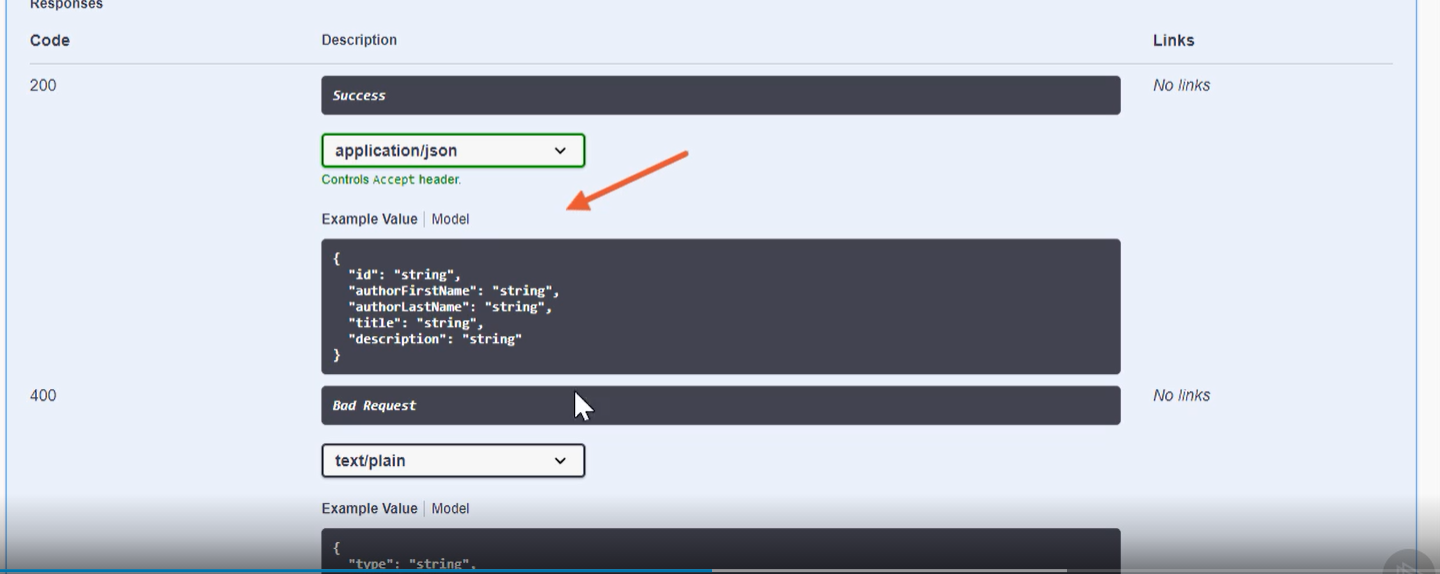




# Overriding convention in OpenAPI

## Responses



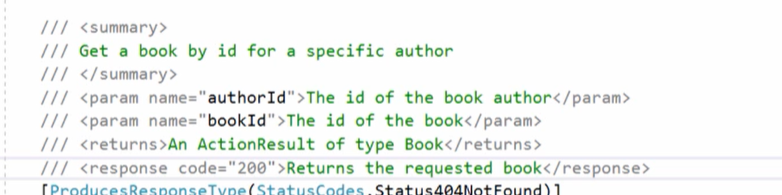




And now our method has documentation of each actions result.



You can send information about an object type with generic ActionResult<T> or in ProduceResponseType. Goot practice is use ActionResult<T> so do it.



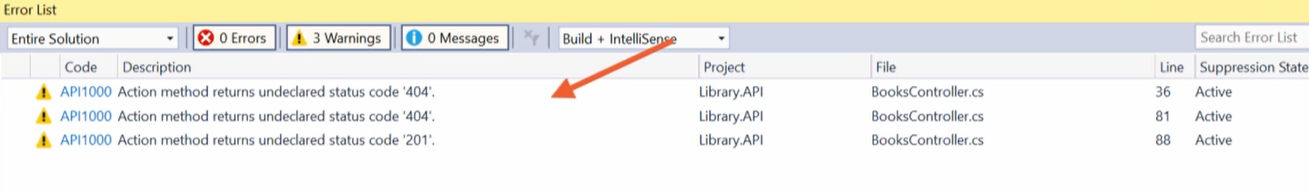
In summary can be more information about an action.



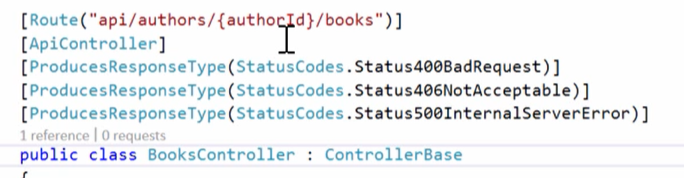
## API Analyzer



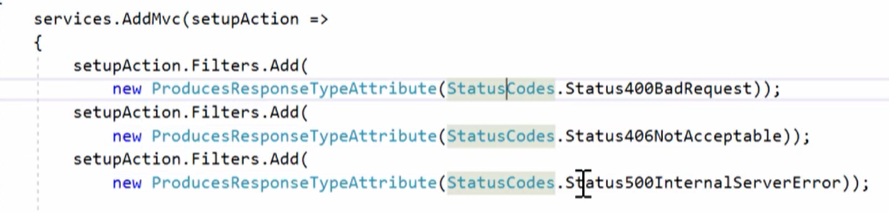
Add package via nugget



And now we have a warning for each action. But remember this tool don’t take care for each status code. 422  Unprocessable Entity or 405 not allowed is not operate

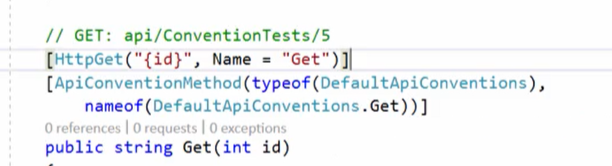


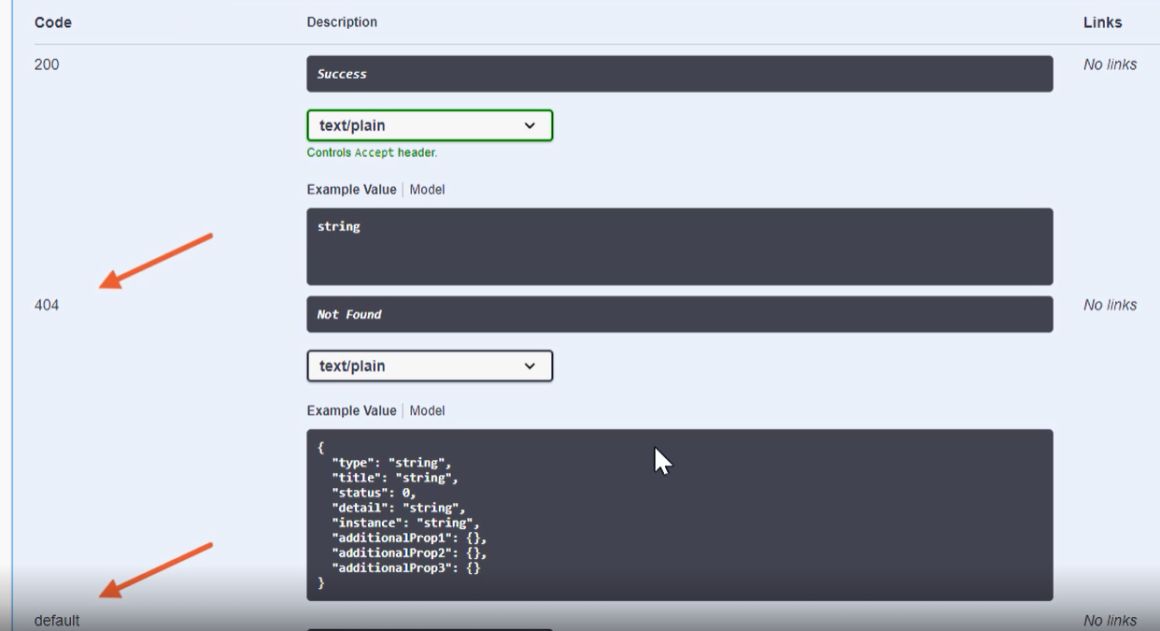
But we can add some response under contro ller to assign its to all action in this class.



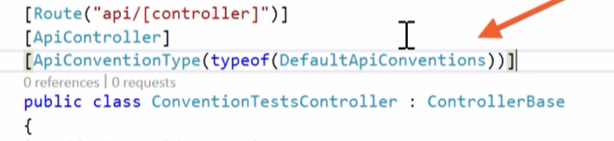
Next way to add some status code is a global approach. In add mvc method add filters with statuscode and now all responses will be available in controllers.

## Working with API Convention

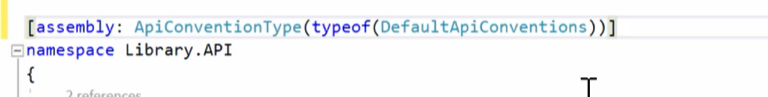




Convention is most uses a status codes

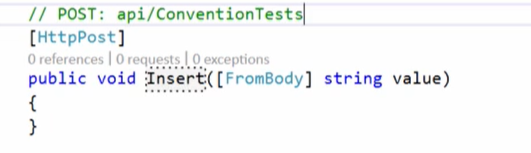


It can be applied to controller level.

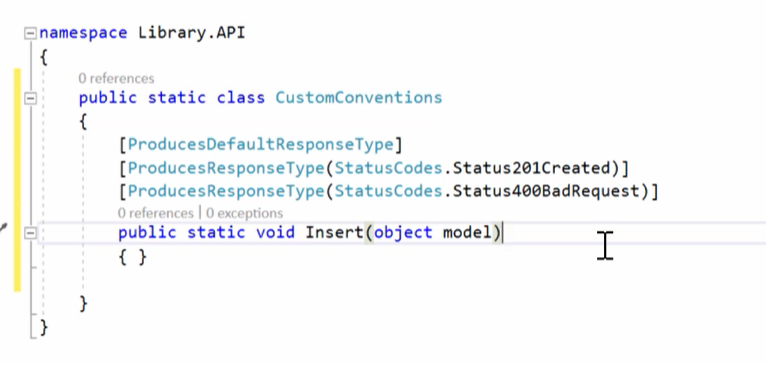


Of course the highest level global can be use too for convention. But remember attributes override conventions

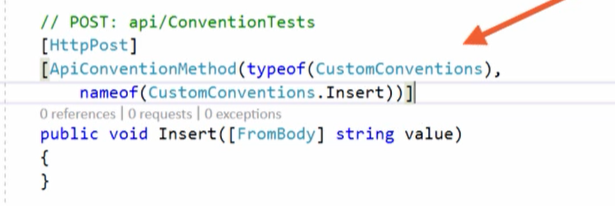
## Create custom convention



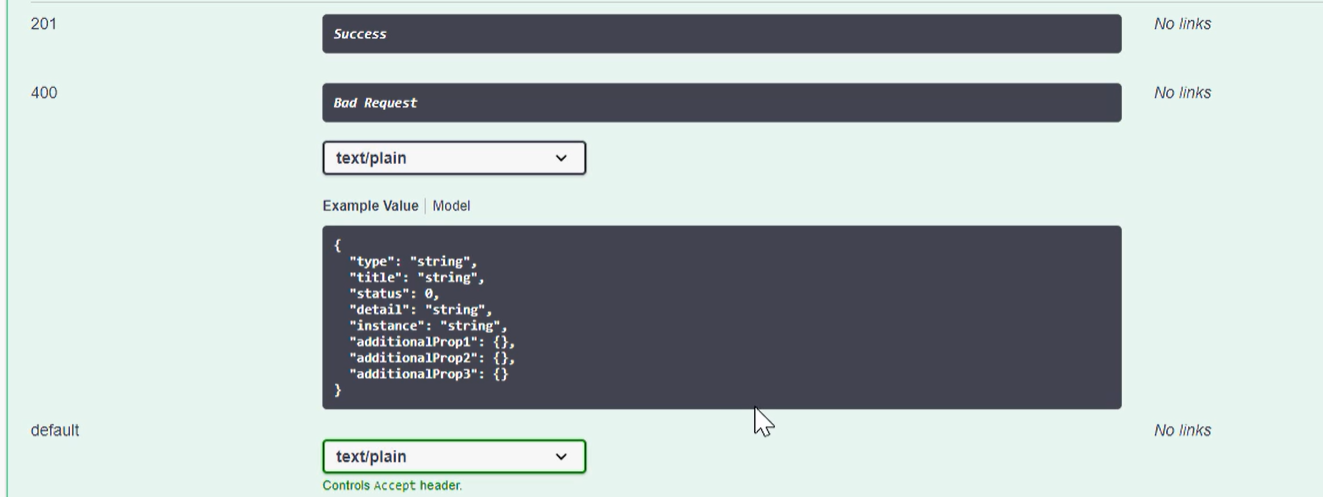
Instead of post is put an insert as name and now convention doesn’t match to this name and we need custom one.



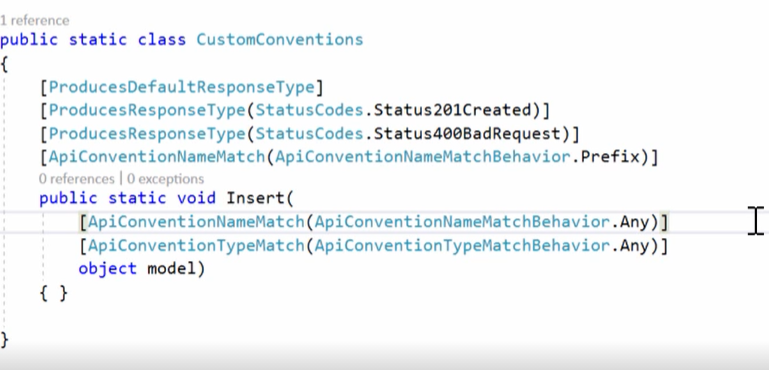
So convention class is simple. It’s static and has annotations about responses. Last thing to remember is to add the same method name type etc to convention method.



Add convention class and method name.

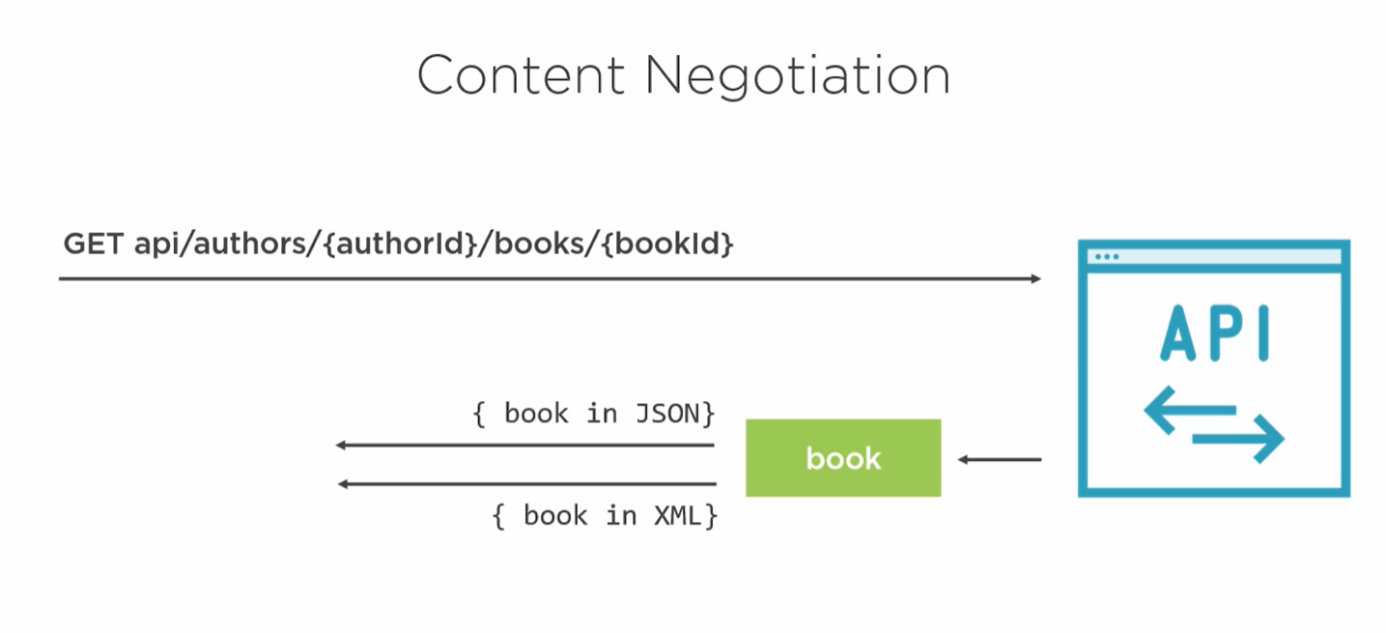


Yep its works.

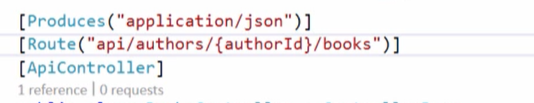


Convention can have more information.

## Content negotiations



It is returning a two or more type in the same api path.



Now our api return only a json.



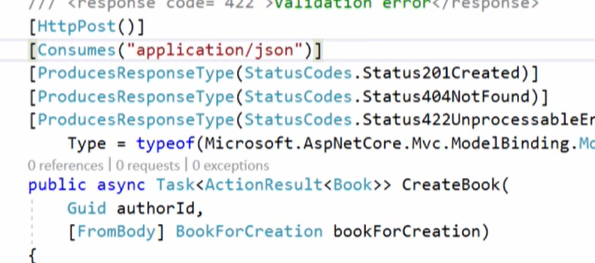
Now our api return a xml but not in swagger, however in postman it works.



So add new produce type and get a xml from swagger.



Of course remember about this attribute above to true. If it has true then doesn’t return a result as default format. Each action with doesn’t supported media type will return 406 not accatable media type.



For post nad put required is consume type not produce like to retrieve operation.