# ASP.NET Middleware map practice

1. Create empty ASP.NET Core project named Middleware\_Map
2. If Project created Startup class, clear all code from Startup class Configure method.
3. Add middleware for each of the following mathematic methods (multiply, divide , add , subtract) each middleware will handle different math method.
4. Receive and split the parameters from the Request using httpContext.Request.RouteValues OR httpContext.Request.Path.Value to determine the math action and the numbers.
5. Return the result using httpContext.Response.WriteAsync method.
6. Url examples
   1. <https://localhost:44335/divide/2/3>
   2. <https://localhost:44335/multiply/2/3>
   3. <https://localhost:44335/subtract/2/3>
   4. <https://localhost:44335/add/2/3>
   5. Use

app.Map("{pattern}",async (context, next) =>

{

//middleware code

});

1. Handle all other request with a Run handler that responds with "invalid" string.
2. Add catch all exceptions in middleware handler, that writes the exception message to the response, using

UseMiddleware method.