ASP .NET Core Life Cycle - Overview

Web Host

Startup.cs

Middleware Handler

alt

[App is started = False]

HTTP Request

(1) Startup Construction
Startup(IHostingEnvironment env
,[DI Interface])
Setup configuration info
(files, variables,..)

(2) Service Configure
ConfigureServices(IServiceCollection services,
[DI Interface])
Add services
(dependency injection, MVC Service,..)

(3) App Configure
Configure(IApplicationBuilder app,
IHostingEnvironment env,
LoggerFactory loggerFactory,
[DI Interface])
Setup middleware
(error handling, security, MVC Middleware,..)

HTTP Request

Invoke Middleware Logic

Handling request logic

[else]

Handling response logic

HTTP Response

<Http Response

The order in which you add middleware components **is** generally **the order** in which they **take effect** on the request, and then in reverse for the response.
*See **(2) App Configure** for adding middlewares*

Available Middlewares(build-in) in .NET Core:
- Error handling
- Diagnostics
- Static file server
- CORS
- Security
- URL Rewriting
- Buffering
- HTTP Overrides
- Response Compression
- Response Caching
- Session
- ROUTING
- MVC
- [User defined middleware]
- [And more build-in middleware in the future]

IMPORTANT:
*[1] Middlewares can share their data by using **HttpContext.Items***
(See: https://docs.microsoft.com/en-us/aspnet/core/fundamentals/app-state)
[2]: The request flow:

Request

Middleware 1
// logic
next();

Middleware 2
// logic
next();

Middleware 3
// logic

// more logic

// more logic

Response

// more logic