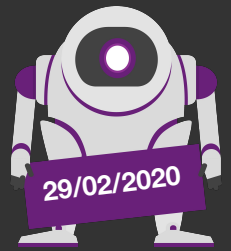
2020

Netcoreconf

Configuración de
aplicaciones en .net core



Fermín Sáez
Programador en AIDA
@mintxelas

Sponsors

Microsoft

avanade

plain concepts

devs dna

everis
an NTT DATA Company

encamina
PIENSA EN COLORES

KABEL

SolidQ

intelequia

東京 TOKIOTA

dinahosting

Application configuration

- Host Configuration vs. App Configuration
- Configuration comes as key/value pairs
- Dictionary-like structure filled by an arbitrary number of providers
- Settings can be overridden: last provider wins.
- Documentation: https://docs.microsoft.com/en-us/aspnet/core/fundamentals/configuration/index?view=aspnetcore-3.1
- More providers: https://github.com/andrewlock/NetEscapades.Configuration
- Examples repo: https://github.com/mintxelas/netcoreconfiguration

Command line arguments, key convention

(DEMO - example 1)

Nuget: Microsoft.Extensions.Configuration.CommandLine

```
var conf = new ConfigurationBuilder()
                    .AddCommandLine(args)
                    .Build();
```

FOO=“Bar” → conf[“foo”] == ”Bar”

Environment Variables, convention

(DEMO - example 2)

(Included in asp.net SDK projects by default)

```
Host.CreateDefaultBuilder(args)
      .ConfigureAppConfiguration((context, builder) =>
            { builder.AddEnvironmentVariables("PREFIX_"); })
      .ConfigureWebHostDefaults(webBuilder =>
            { webBuilder.UseStartup<Startup>(); });
```

PREFIX_FOO = “Bar” ⟶ conf[“foo”] == ”Bar”

Json Files, complex hierarchy

(DEMO - example 3)

(Included in asp.net SDK projects by default)

```
Host.CreateDefaultBuilder(args)
      .ConfigureAppConfiguration((context, builder) =>
            {
                  builder.AddJsonFile("appsettings.json");
                  builder.AddJsonFile($"appsettings.{context.HostingEnvironment.EnvironmentName}.json");
            })
      ...
```

```
{
   “Section”: {
      “Foo”: “Bar”
   }
}
```

→ Conf[“section:foo”] == “Bar”

Azure App Configuration, UserSecrets

(DEMO - example 4)

Azure App Configuration nuget: Microsoft.Extensions.Configuration.AzureAppConfiguration

UserSecrets nuget: Included in asp.net SDK projects by default

```
Host.CreateDefaultBuilder(args)
      .ConfigureAppConfiguration((context, builder) =>
            {
                  builder.AddJsonFile("appsettings.json"); // not strictly necessary
                  builder.AddUserSecrets<Program>();
                  var conf = builder.Build();
                  builder.AddAzureAppConfiguration(conf["ConnectionStrings:AzureAppConfiguration"]);
            })
      ...
```

C:\Users\xxxxxx\AppData\Roaming\Microsoft\UserSecrets

Custom Configuration Provider

(DEMO - example 5)

```
public class CustomConfigurationSource : IConfigurationSource

public class CustomConfigurationProvider : IConfigurationProvider
```

```
public static IConfigurationBuilderAddCustomProvider(this IConfigurationBuilder builder, string connectionString)
{
      return builder.Add(new CustomConfigurationSource(connectionString));
}

builder.AddCustomProvider(conf["ConnectionStrings:sqlserver"]);
```

Strongly-Typed configuration

(DEMO - example 6)

```
services.Configure<ExternalApiConfiguration>(Configuration.GetSection("ExternalApi"));
```

```
public class ExternalApiConfiguration
 {
     public string AuthorizationServer { get; set; }
     public string ClientId { get; set; }
     public string ClientSecret { get; set; }

 }
```

```
{
  ...,
  "ExternalApi": {
    "AuthorizationServer": "",
    "ClientId": "settings tipados",
    "ClientSecret":  ""
  }
}
```

Configuration injection, hot reload

(DEMO - example 7)

```
builder.AddJsonFile("appsettings.json", optional: true, reloadOnChange: true);

services.Configure<ExternalApiConfiguration>(Configuration.GetSection("ExternalApi"));

private readonly ExternalApiConfiguration configuration;
private readonly ExternalApiConfiguration configurationSnapshot;

public HomeController(IOptions<ExternalApiConfiguration> options,
          IOptionsSnapshot<ExternalApiConfiguration> optionsSnapshot)
{
    configuration = options.Value;
    configurationSnapshot = optionsSnapshot.Value;
}
```

Sponsors

Microsoft

avanade

plain concepts

devs dna

everis
an NTT DATA Company

encamina
PIENSA EN COLORES

KABEL

SolidQ

intelequia

東京 TOKIOTA

dinahosting

Más información:

info@netcoreconf.com
@Netcoreconf

Visítanos en:
netcoreconf.com