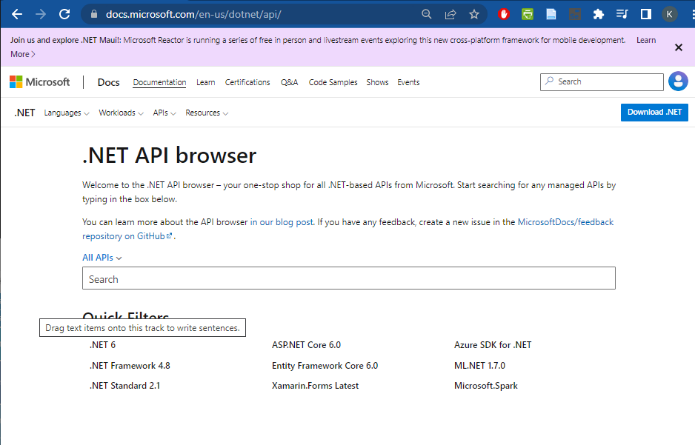
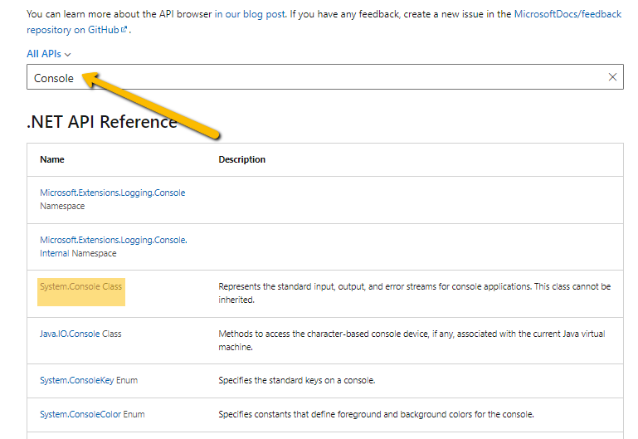
**Læringsmål:**

Brug API beskrivelser til at undersøge hvad vi kan gøre med .NET’s Console klasse.

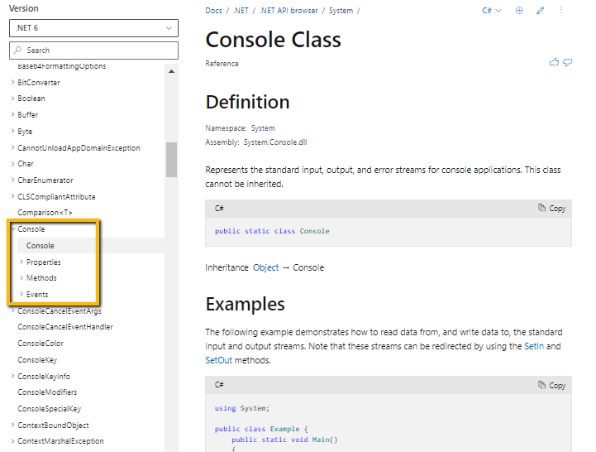


Når vi arbejder med kode bruger vi et API og vi har tit brug for at kunne slå op hvad der gælder om dele vores kode består af. Når vi arbejder med Microsoft .NET vil det følgende link til den officielle dokumentation være et godt sted at lede:

<https://docs.microsoft.com/en-us/dotnet/api/>



Søger vi efter Console får vi en lang række forslag, det vi leder efter er den basale System.Console klasse



Vi ender på et link som dette og kan nu se hvordan dokumentationen er bygget op

<https://docs.microsoft.com/en-us/dotnet/api/system.console?view=net-6.0>

Vi ser der er et stort afsnit med eksempler, derudover findes der

* Properties
* Methods
* Events

**Console.BackgroundColor og Console.ForegroundColor**

Lad os sige at vi er interesseret i at arbejde med baggrundsfarven i vores consol program

<https://docs.microsoft.com/en-us/dotnet/api/system.console.backgroundcolor?view=net-6.0>

Af linket lærer vi at der findes en ConsoleColor der kan bruges til at angive farve.

<https://docs.microsoft.com/en-us/dotnet/api/system.consolecolor?view=net-6.0>

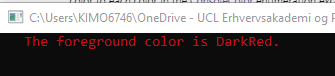
og at vi kan skrive feks

ConsoleColor myColor = ConsoleColor.DarkRed;

Console.ForegroundColor = myColor;

Console.WriteLine(" The foreground color is {0}.", myColor);

For at få dette output



|  |  |
| --- | --- |
| **Opgave 1: Baggrundsfarve og forgrundsfarve**  Søg rundt I API beskrivelserne   * Console Class * Console.BackgroundColor Property * Console.ForegroundColor Property * ConsoleColor Enum   Og skriv et consol program der giver det viste output |  |
| **Opgave 2: Console Title**  I samme program tag magten over defaults og give dit konsol vindue en relevant titel  **Console.Title Property** |  |
| **Opgave 3: Sound a beep**  Console.Beep(); | Når programmet afslutter  Lad konsolen afgive et beep |

**Demo og opgave: Pimp your Console med Spectre!**

Se på denne side

<https://www.hanselman.com/blog/spectreconsole-lets-you-make-beautiful-console-apps-with-net-core>

Skribenten hedder Scott Hanselman, skvatter du over det navn når du søger efter artikler, er den temmelig sikkert værd at læse. Her skriver han om en udvidelse der giver os mange flere muligheder med et konsol program. Ved at bruge en udvidelse kan vi flere ting med vores Console.

Kernen i vores muligheder ligger i at inkludere Spectre.Console i vores konsol program. Når vi har fået det til at virke, vil vi sammen med dokumetationen du finder her: <https://spectreconsole.net/> være i stand til at gøre meget mere end hvad vi så i de første opgaver.

Men først skal vi have sat projektet op, hvilket vil omfatte koncepter vi først senere på kurset vil komme tættere på. Men følger du denne screencast som code-along, vil du kunne eksperimentere med Spectre.Console. Den varer 25 minutter.

<https://www.studietube.dk/video/7728328/840863ffa9fd3b9451de8b3f4c06a33f>

**Opgave**

Når du har fået det til at fungere kan du prøve om i dit konsol program kan få dette til at virke. Juster i koden så progressbaren ikke færdiggører lige med det samme-:

AnsiConsole.Progress()

.Start(ctx =>

{

// Define tasks

var task1 = ctx.AddTask("[green]Reticulating splines[/]");

var task2 = ctx.AddTask("[green]Folding space[/]");

while (!ctx.IsFinished)

{

task1.Increment(1.5);

task2.Increment(0.5);

System.Threading.Thread.Sleep(50);

}

});