

F# in the real world

Part 1

F# basics

C#:

```
var color = "blue";
```

```
public int Add(int a, int b){
```

```
    return a + b;
```

```
}
```

F#:

```
let color = "blue"
```

```
let add a b = a + b
```

Pipelineing

C#:

```
public int AddOne(int a){  
  
    return a + 1;  
  
}
```

```
var four = addOne(addOne(addOne(1)));
```

F#:

```
let addOne a = a + 1
```

```
let four = 1 |> addOne |> addOne |> addOne
```

Pipelineing

F#: Legg til 3 til alle tall i lista, filtrer vekk oddetall, skriv ut

```
[| 1..10 |]  
|> Array.map (fun i -> i+3)  
|> Array.filter (fun i -> i%2 = 0)  
|> Array.iter (printfn "value is %i. ")
```

Typer

```
public class Car
{
    public int Id { get; }
    public string Make { get; }
    public string Model { get; }

    public Car(int id, string make, string model)
    {
        Id = id;
        Make = make;
        Model = model;
    }
}
```

Typer

```
type Car = {
```

```
  Id : int
```

```
  Make : string
```

```
  Model : string
```

```
}
```

Instansiere:

```
let car = {Id = 1; Make = "Volvo"; Model = "V70"}
```

Part 2

SQL TypeProvider

- Gir deg typer, properties og metoder for å jobbe direkte med tabeller i SQL-databaser
- Dokumentert fint her: <https://msdn.microsoft.com/en-us/library/hh361033.aspx>
- Bruker connection string til å generere typer rett fra databasen
- Gir deg typesikker tilgang til databasen. Eliminerer muligheten for en del teite runtime-feil helt #winning
- For de som liker å skrive SQL selv: <http://fsprojects.github.io/FSharp.Data.SqlClient/>

Part 3

JSON-TypeProvider

- Statisk typet JSON-leser
- Fra biblioteket F# Data
- Kan types fra en mal(json-fil) eller rett fra et endepunkt
- Hvis man bruker et endepunkt som mal, vil koden ikke kompliere dersom endepunktet endrer seg

JSON-TypeProvider

```
type Provider = JsonProvider<sampleJson>
```

```
type Simple = JsonProvider<"" { "name":"John", "age":94 } "">
```

```
type WorldBank = JsonProvider< "../data/WorldBank.json" >
```

```
let doc = WorldBank.Load("http://api.worldbank.org/country/cz/indicator/GC.DOD.TOTL.GD.ZS?  
format=json")
```

JSON-TypeProvider

F# Data:

```
type GitHub = JsonProvider<"https://api.github.com/repos/fsharp/FSharp.Data/issues">
```

```
let sample = GitHub.LoadSamples()
```

JSON-TypeProvider

Oppgaver: [fsharp-in-the-real-world\Part3\readme.md](#)

Utgangspunkt: [fsharp-in-the-real-world\Part3\Exercise](#)

Løsning: [fsharp-in-the-real-world\Part3\Solution](#)