

# Using Components in Blazor

Cristian Civera

Senior Software Architect

cristian@cristiancivera.com | @cristiancivera

Blazor  
Conference  
2021

# Component

- File component.razor
  - Component.razor.cs
  - Classe con namespace
- Utilizzo come tag
  - Attributi-> proprietà
- Parametri
  - Attributo Parameter
  - Auto property
- OnParametersSet e OnParametersSetAsync

# Child content

- Proprietà ChildContent
  - Predefinita per figli componenti
- RenderFragment
  - Delegato rendering parziale
- Proprietà con altri nome
  - Sotto tag del componente
- Supporto a generic
  - @typeparam

# Attribute splatting and arbitrary parameters

- Cattura degli attributi addizionali
- Parameter con CaptureUnmatchedValues
  - `IEnumerable<KeyValuePair<string, object>>` o `IReadOnlyDictionary<string, object>`
- Attributo speciale `@attributes`
  - Applica gli attributi

```
[Parameter(CaptureUnmatchedValues = true)]
public Dictionary<string, object> InputAttributes { get; set; }

<input id="useAttributesDict" @attributes="InputAttributes" />
```

# Cascading value

- Trasporta un valore sull'albero dei componenti
- Attributo CascadingParameter
  - Per leggerlo
- EditContext
  - Fornito da EditForm
  - Per interagire con validazione
- Creazione custom con CascadingValue
  - Idee: Spinner, gestione errori

# Eventi

- Parametro di tipo EventCallback
  - Anche tipizzato
- Supporto sincrono e asincrono
  - InvokeAsync per lanciare l'evento
- Invalida automaticamente chi gestisce l'evento
  - StateHasChanged non richiesto

# Data binding

- Supporto a @bind-Proprietà
- Proprietà/parameter per lettura/scrittura
- Evento ProprietàChanged
- Proprietà ProprietàExpression
  - Expression<Func<T, object>>
  - Opzionale
  - Da usare eventualmente insieme a EditContext

# Demo

SubmitButton

Blazor  
Conference  
2021

@cristiancivera  
cristian@cristiancivera.com

Slide e materiale su  
<https://aspit.co/BlazorConf-21>

Blazor  
Conference  
2021