

1. RIFERIMENTI DOCUMENTAZIONE

https://instruments.jamio.com/knowledge_base/index.html?costruiredocumentling.htm

https://instruments.jamio.com/knowledge_base/index.html?plugdocumentautomation.htm

2. DEFINIZIONE INPUT DEL METODO

Negli esempi che seguono si ipotizza di aver fornito in input al metodo di autocomposizione un dato struttura contenente

- un campo denominato **Risorsa** di tipo collegamento. Questo campo è valorizzato con il collegamento di una istanza di form di tipo Dati.
- Un campo denominato **InfoAggiuntive** di tipo dato strutturato. Il dato strutturato è così composto
 - **DataStampa**, di tipo data e orario
 - **RichiedenteStampa**, di tipo testo
 - **RichiestaStart**, di tipo collegamento

3. FORMALISMO DI SCRITTURA

Il formalismo LINQ prevede la definizione del campo da stampare racchiuso tra due parentesi angolari di apertura "<<" e due di chiusura ">>".

All'interno di tali parentesi il campo da stampare deve essere racchiuso a sua volta da due parentesi quadre, una di apertura e una di chiusura.

<<[CAMPO DA STAMPARE]>>

Per la definizione del campo da stampare possiamo seguire il formalismo simile a quanto utilizzato nella modalità espressione del composer.

Ad esempio, se volessi stampare il Name (proprietà di piattaforma di un modello) allora si dovrà scrivere come segue

<<[Risorsa.Name]>>

Dove **Risorsa** è il campo passato in input al metodo di autocomposizione (come riportato al paragrafo precedente)

Qualora si desidera stampare un campo personalizzato della Risorsa, i cosiddetti Fields che il progettista può definire, allora si aggiunge il costrutto *Fields* e all'interno di due parentesi quadre il nome del campo

Fields[NOME CAMPO DA STAMPARE]

Ad esempio, se volessi stampare il campo personalizzato *RagioneSociale* (un Fields custom del modello) allora si dovrà scrivere come segue

```
<<[Risorsa.Fields["RagioneSociale"]]>>
```

Dove **Risorsa** è il campo passato in input al metodo di autocomposizione (come riportato al paragrafo precedente) e **RagioneSociale** il nome del campo fields

Qualora, invece, sia necessario impostare una formattazione particolare per il campo è possibile aggiungerla antecedente alle parentesi angolari di chiusura.

Ad esempio, se volessi stampare il campo CostoPreventivo, un campo di tipo numerico decimale e vorrei visualizzarlo con due cifre decimali

```
<<[Risorsa.Fields["CostoPreventivo"].Number]:"0.00">>
```

Dove

- **Risorsa** è il campo passato in input al metodo di autocomposizione (come riportato al paragrafo precedente)
- **CostoPreventivo**, un campo numerico decimale definito come Fields dell'istanza **Risorsa**

4. ESEMPI

1.1 Campo testo

```
<<[Risorsa.Fields["RagioneSociale"]]>>
```

Stampare una stringa solo se valorizzata

```
<<if [Risorsa.Fields["Responsabile"].Name.ToString().Length>0]>>  
<<[Risorsa.Fields["Responsabile"].Name]>><<else>>Non presente<</if>>
```

1.2 Campo RichTextArea

```
<<[Risorsa.Fields["CorpoDelMessaggio"]] -html>>
```

1.3 Campo data

Stampato il <<[InfoAggiuntive.Fields["DataStampa"].DateTime]:"dd MMMM yyyy HH:mm">>
Stampato da <<[InfoAggiuntive.Fields["RichiedenteStampa"]]>>

Stampare il campo data solo se valorizzato

```
<<if [Risorsa.Fields["Data"].DateTime.Year==0001]>> <<else>>  
<<[Risorsa.Fields["Data"].DateTime]:"dd/MM/yyyy HH:mm:ss">> <</if>>
```

1.4 Campo numerico

in questo caso abbiamo definito la stampa con due cifre decimali

```
<<[Risorsa.Fields["CostoPreventivo"].Number]:"0.00">>
```

In questo caso con formattazione ad esempio 2.000.500,50
€ <<[Risorsa.Fields["CostoPreventivo"].Number]:"###,##0.00">>

1.5 Somma

<<[Risorsa.Fields["Articoli"].Sum(item => item.Fields["Costo"].Number)]:"0.00">> €

1.6 Campi booleani con if

<<if [Risorsa.Fields["PersonaFisica"].IsTrue]>> <<[Risorsa.Fields["RagioneSociale"]]>> <<else>>
<<[Risorsa.Fields["Nome"]]>> <<[Risorsa.Fields["Cognome"]]>> <</if>>

Documentazione presente <<if [Risorsa.Fields["ck_DocumentazionePresente"].IsTrue]>> ☒ <<else>> ☐
<</if>>

<<if [Risorsa.Fields["Sesso"].ToString() == "M"]>> **MASCHIO**<<else>> **FEMMINA**<</if>>

1.7 Campo lista collegamenti di cui stampiamo il Name

<<if [Risorsa.Fields["Approvatori"].Any()==true]>>**Approvatori:**
<<foreach [item in Risorsa.Fields["Approvatori"]]>><<[item.Name]>>
<</foreach>><</if>>

1.8 Campo lista dati strutturati o collegamenti estesi

<<if [Risorsa.Fields["Note"].Any()==true]>>**ANNOTAZIONI**

Autore	Data	Nota
<<foreach [item in Risorsa.Fields["Note"]]>><<[item.Fields["Autore"]]>>	<<[item.Fields["Data"].DateTime]:"dd/MM/yyyy HH:mm">>	<<[item.Fields["Annotazioni"]]>> <</foreach>>

<</if -greedy>>

In questo caso si utilizza -greedy per concludere un IF così se non sono presenti note lo spazio occupato dal testo eliminato senza lasciare spazi bianchi.

1.9 Link risorsa Jamio

<<link ["https://paas.jamio.com/jamio/?url=" + InfoAggiuntive.Fields["RichiestaStart"].Url] [[Clicca per visualizzare i dettagli](#)]">>