

[Global Code] Como criar uma Classe Global em C#?

#GlobalCode

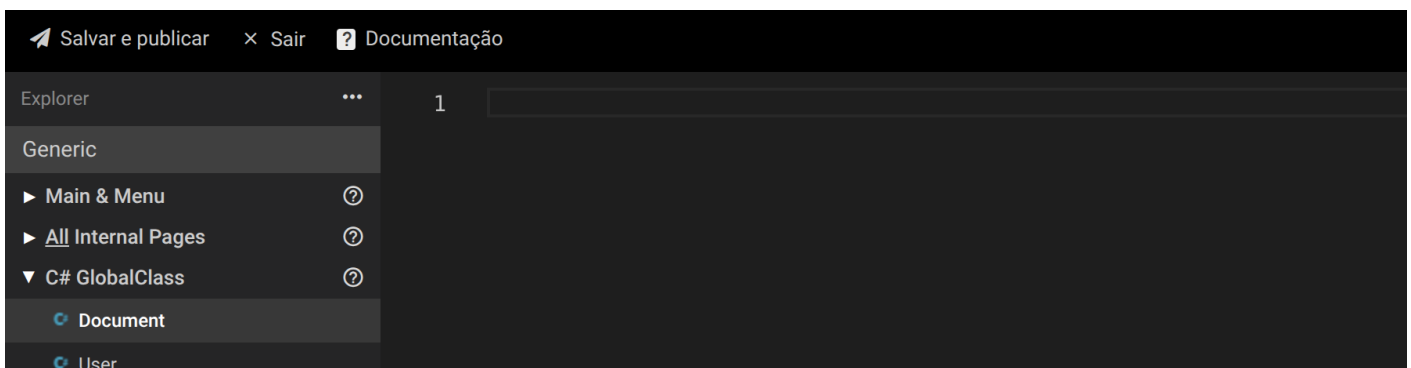
IMPORTANTE: a class não pode ser instanciada diretamente, ex: `new MyClass()` utilize `CreateInstanceMyClass()` que é gerado automaticamente

IMPORTANTE: a class não pode ter construtor definido

Como definir e instanciar uma classe C# que está definida no Global Code?

Entre em **Global Code** na área de configuração, na aba C# GlobalClass e clique em Novo C#.

Crie um C# chamado "Document"



No código C#, copie e cole o código abaixo, repare que foi criado manualmente o metodo fora da classe chamado `CreateDocument` com 2 parametros que instancia a classe `Document`, através dele criaremos a instancia quando formos usar a classe `Document`, mas o Agilityflow também gera um metodo padrão chamado **CreateInstanceDocument** sem parametro para gerar a instancia, caso você precise.

Importante: para toda classe criada no agilityflow, o sistema gera automaticamente um metodo de criação da instancia dessa classe com o nome CreateInstanceXXXXX onde o XXXX é o nome da classe, se for uma classe chamada Document, ele cria **Document CreateInstanceDocument()**, sempre sem parametros de entrada

```
//crie uma classe
public class Document{
    public string Title {get;set;}
    public string Description {get;set;}

    public async Task<List<UserClass>> ListUserExamplesAsync(){
        var userNames = new List<UserClass>();

        var dt = await GlobalContext.GetDataTableAsync("select coalesce(usu_nome,") as name from tbl_usuario
where deletado = false limit 5");

        for(var i = 0; i < dt.Rows.Count; i++){

            //repare que pra criar a instancia da classe UserClass, eu utilizo o metodo CreateInstance
            var userClass = CreateInstance<UserClass>();
            userClass.Name = dt.Rows[i]["name"].ToString();
            userNames.Add(userClass);
        }

        return userNames;
    }
}

public class UserClass{
    public string Name {get;set;}
}

//caso queira, crie um metodo para fazer a instancia dessa classe
public Document CreateDocument(string title, string description ){

    var doc = CreateInstance<Document>();
    doc.Title = title;
    doc.Description = description;
```

```
return doc;
}
```

Como instanciar uma classe do Global Code entre outras classes do Global Code?

Suponhamos que você tenha uma classe chamada OutraClasseGlobal no Global Code

```
public class OutraClasseGlobal{

    public string Nome {get;set;}

}
```

Para instanciar essa classe OutraClasseGlobal dentro do próprio Global Code você tem as opções abaixo:

CreateInstance<T>()

```
public class MinhaClasseGlobal{

    public void Metodo(){

        var classe2 = CreateInstance<OutraClasseGlobal>();

    }

}
```

Como usar em outras partes do sistema essa classe C# Global (em outros contextos) ?

Por ser uma classe Global, você pode utilizar em qualquer lugar que possa executar C# no sistema, exemplos: Apis, Forms, Reports, Kanban, etc..

Para Utilizar, por exemplo dentro de um Cshtml de um form, você tem essas opções:

Opção 01: Chamando o metodo que o agilityflow gera automaticamente **(recomendado)**:

Importante: para toda classe criada no agilityflow, o sistema gera automaticamente um metodo de criação da instancia dessa classe com o nome CreateInstanceXXXXX onde o XXXX é o nome da classe, se for uma classe chamada Document, ele cria **Document CreateInstanceDocument()**

```
@{
    //instanciando uma class q foi declarada dentro do Global dinamicamente
    var documentClass01 = GlobalClass.CreateInstanceDocument();

    var json = JsonConvert.SerializeObject( new { documentClass01 } );

}
@json
```

Opção 02: Chamando o metodo nativo do agiltyflow para instanciar classes:

```
@{
    //instanciando uma class q foi declarada dentro do Global dinamicamente
    var documentClass02 = GlobalClass.CreateInstance("Document");

    var json = JsonConvert.SerializeObject( new { documentClass02 } );

}
@json
```

Opção 03: Caso você crie um método manualmente que instancie a classe, exemplo o método criado no exemplo anterior

CreateDocument (que você mesmo criou) :

```
@{
    //instanciando uma class q foi declarada dentro do Global dinamicamente
    var documentClass01 = GlobalClass.CreateDocument("string title", "string description");

    var json = JsonConvert.SerializeObject( new { documentClass01 });
}
@json
```

Regras e Limitações:

- Para toda classe criada no agilityflow, o sistema gera automaticamente um método de criação da instância dessa classe como nome `CreateInstanceXXXXX` onde o `XXXXX` é o nome da classe, se for uma classe chamada `Document`, ele cria **Document CreateInstanceDocument()**, porém sempre sem parâmetros de entrada, você pode gerar manualmente um **método com o mesmo nome mas com outros parâmetros de entrada, exemplo Document CreateDocument(string title, string description)**

A classe não deve ter construtor, para instanciar, utilize o método

CreateInstance<MyClass>()

- Fora da área do C# do Global Code, você não tem acesso ao "Tipo" da classe que você criar. Então fora do seu `class c#` de Global Code você não vai conseguir usar esse "Tipo", ex: como **ToList<MyClass>()** ou por exemplo em uma referência implícita no `foreach`,:

```
//forma ERRADA se estiver FORA e consumindo o GlobalCode, (se estiver dentro do GlobalCode, você pode referenciar normalmente)
```

```
//referenciar explicitamente o UserClass no foreach
```

```
var userExamples = await GlobalClass.ListUserExamplesAsync();
```

```
foreach(UserClass user in userExamples){  
  <text>  
    @user.Name  
  </text>  
}
```

//forma CORRETA se estiver FORA e consumindo o GlobalCode, (se estiver dentro do GlobalCode, você pode referenciar normalmente)

//referenciar implicitamente o UClass no foreach, usando var ou dynamic

```
var userExamples = await GlobalClass.ListUserExamplesAsync();
```

```
foreach(var user in userExamples){  
  <text>  
    @user.Name  
  </text>  
}
```

Revision #46

Created 24 February 2025 21:33:17 by agilityflow

Updated 14 July 2025 12:44:34 by agilityflow