文章
姚 鑫 · 七月 24, 2021 阅读大约需 2 分钟

类关键字DependsOn，Deprecated，Final，GeneratedBy

第十七章 类关键字 - DependsOn

指定此类应在编译器使其他(指定的)类可运行后编译。

用法

要指示类编译器应该在其他类可运行后编译此类，请使用以下语法:

```
Class MyApp.MyClass [ DependsOn = classlist ]
{ //class members }
```

其中classlist是下列之一:

- 类名

```
[ DependsOn = MyApp.Class1 ]
```

- 用逗号分隔的类名列表，用括号括起来。

```
[ DependsOn = (MyApp.Class1,MyApp.Class2,...) ]
```

详情

此关键字指定类编译器应该在使指定的类可运行后编译此类。

如果此类的编译使用方法生成器逻辑中的这些其他类，则此关键字非常有用。如果类包含调用其他类的初始表达式，这也很有用。

此关键字仅影响编译顺序，不影响运行时行为。

注意:DependsOn关键字只影响具有System关键字的公共值的类。

此外，如果一个类有DependsOn=ClassA，那么它也有CompileAfter = ClassA也是多余的。

对子类的影响

这个关键字继承自所有超类。如果子类为关键字指定了一个值，该值指定了在子类被编译之前必须运行的附加类。

默认

默认情况下，不指定该关键字。

第十八章 类关键字 - Deprecated

指定不推荐使用此类。该关键字被类编译器忽略，只是提供了一个人类可读的指示，表明该类已被弃用。

用法

要将类标记为已弃用，请使用以下语法:

```
Class MyApp.MyClass [ Deprecated ]
{ //class members }
```

否则，省略此关键字或将单词Not放在关键字的前面。

第十九章 类关键字 - Final

指定此类是否是最终类(不能有子类)。

用法

要指定一个类是最终类，请使用以下语法:

```
Class MyApp.Exam As %Persistent [ Final ]  { //class members }
```

否则，省略此关键字或将单词Not放在关键字的前面。

详情

如果一个类是最终类，它就不能有子类。

此外，如果一个类是最终类，类编译器会利用某些代码生成优化(与最终类的实例不能多形态使用的事实有关)。

默认

如果省略此关键字，类定义不是最终的。

第二十章 类关键字 - GeneratedBy

指示此类是由另一个类中的代码生成的，因此不应编辑。

用法

以下语法指示此类是由另一个类中的代码生成的:

```
Class MyApp.MyClass [ GeneratedBy = MyApp.Generator.cls ] { //class members }
```

MyApp.Generator是一个完全限定的类名。

详解

如果指定了该关键字，Studio将以灰色背景显示该类，以指示不应编辑该类。

对子类的影响

此关键字不是继承的。

默认

如果省略此关键字，Studio将正常显示该类。

#Caché #Ensemble

源URL: https://cn.community.intersystems.com/post/%E7%B1%BB%E5%85%B3%E9%94%AE%E5%AD%97dependson%EF%BC%8Cdeprecated%EF%BC%8Cfinal%EF%BC%8Cgeneratedby