

[文件](#) » 蜘蛛合同

# 蜘蛛合同

0.15版本的新功能。

## ⓘ 注意

这是一项新功能（在Scrapy 0.15中引入），可能会受到次要功能/ API更新的影响。检查[发行说明](#)以获得更新通知。

测试蜘蛛会变得特别烦人，虽然没有什么能阻止你编写单元测试，但任务很快就会变得很麻烦。Scrapy提供了一种通过合同测试蜘蛛的综合方法。

这允许您通过对示例URL进行硬编码来测试蜘蛛的每个回调，并检查回调如何处理响应的各种约束。每个合同都带有一个前缀@并包含在docstring中。请参阅以下示例：

```
def parse(self, response):  
    """ This function parses a sample response. Some contracts are mingled  
    with this docstring.  
  
    @url http://www.amazon.com/s?field-keywords=selfish+gene  
    @returns items 1 16  
    @returns requests 0 0  
    @scrapes Title Author Year Price  
    """
```

使用三个内置契约测试此回调：

## 类 scrapy.contracts.default.UrlContract

此contract ( @url ) 设置在检查此蜘蛛的其他合同条件时使用的样本URL。该合同是强制性的。运行检查时，将忽略缺少此合同的所有回调：

```
@url url
```

## 类 scrapy.contracts.default.ReturnsContract

此contract ( @returns ) 设置蜘蛛返回的项目和请求的下限和上限。上限是可选的：

```
@returns item(s)|request(s) [min [max]]
```

此contract ( @scrapes ) 检查回调返回的所有项是否具有指定的字段：

```
@scrapes field_1 field_2 ...
```

使用该 `check` 命令运行合同检查。

## 自定义合同

如果您发现需要比内置scrapy合同更多的功能，则可以使用以下 `SPIDER_CONTRACTS` 设置在项目中创建和加载自己的合同：

```
SPIDER_CONTRACTS = {  
    'myproject.contracts.ResponseCheck': 10,  
    'myproject.contracts.ItemValidate': 10,  
}
```

每个合同都必须继承 `scrapy.contracts.Contract` 并且可以覆盖三种方法：

---

**class scrapy.contracts.Contract ( 方法 , \* args )**

- 参数：**
- `method ( function )` - 与合同关联的回调函数
  - `args ( list )` - 传递给docstring的参数列表（以空格分隔）

**adjust\_request\_args ( args )**

它接收一个 `dict` 包含 `Request` object的默认参数的参数。必须返回相同或修改后的版本。

**pre\_process ( 回应 )**

这允许在将样本请求传递给回调之前对从样本请求接收的响应进行各种检查。

**post\_process ( 输出 )**

这允许处理回调的输出。在传递给这个钩子之前，迭代器被转换为listified。

这是一个演示合同，它检查收到的响应中是否存在自定义标头。 `scrapy.exceptions.ContractFail` 为了让失败得到印刷而提升：

```
from scrapy.contracts import Contract
from scrapy.exceptions import ContractFail

class HasHeaderContract(Contract):
    """ Demo contract which checks the presence of a custom header
        @has_header X-CustomHeader
    """

    name = 'has_header'

    def pre_process(self, response):
        for header in self.args:
            if header not in response.headers:
                raise ContractFail('X-CustomHeader not present')
```