الكود يستخدم Scrapy، وهو إطار عمل لتنقيح الويب في Python، لتنقيح مقالات الأخبار من الموقع "<https://www.independent.co.uk/>" و ها في ملف CSV يدعى "results.csv".

تصدير الوحدات

* وحدات Scrapy: يتم تصدير مكونات أساسية من Scrapy، بما في ذلك CrawlerProcess, Request, و ItemLoader. هذه تستخدم لتعريف الروبوت، وإدارة عملية التنقيح، وإدارة الطلبات HTTP، وتحميل العناصر المتنقية.
* تغيير المسار النظامي: sys.path.insert(0, '..') يضيف الدليل الأب إلى المسار النظامي، مما يسمح للكود بإعادة تشغيل تعريف العناصر المخصصة من وحدة محلية تسمى items.
* التعامل مع CSV: يتم تصدير وحدة csv لكتابة بيانات المتنقة في ملف CSV.

تعريف الروبوت

* الفئة newzspider3 تورث من scrapy.Spider: هذا يعرف روبوت Scrapy جديد باسم 'newzspider3'.
* تكوين الروبوت:
  + start\_urls: قائمة بأرقام بداية للروبوت. هنا، يبدأ بموقع الصفحة الرئيسية لـ "<https://www.independent.co.uk/>".
  + allowed\_domains: قائمة بالأطر الزمنية التي يسمح للروبوت بتنقيحها. هنا، محدد على "independent.co.uk".
  + headers: رأس HTTP مخصص يتم إرساله مع كل طلب يقوم به الروبوت. هذه الرؤوس تتقليد طلب ويب عادي لتجنب الحظر من قبل الخادم.

التعامل مع الطلبات

* الدالة start\_requests: تولد الطلب الأولي للموقع الرئيسي للموقع باستخدام الرؤوس المحددة وتعيين parse كدالة استدعاء لمعالجة الاستجابة.

منطق التنقيح

* الدالة parse: استخراج الروابط من الصفحة الرئيسية التي تؤدي إلى مقالات أخبار فردية، مع تجنب تلك المرتبطة بـ الرموز. لكل رابط، يرسل طلب متابعة إلى صفحة المقال بنفس الرؤوس وتعيين parse\_details كدالة استدعاء.
* الدالة parse\_details: معالجة محتوى الصفحة لكل مقال.
  + استخراج النص، العنوان، اسم الكاتب، وموقع URL.
  + التحقق مما إذا كانت الرسالة موجودة بالفعل في ملف CSV لتجنب النسخ المكررة.

وحدة التنفيذ الرئيسية

* إعداد وتنفيذ CrawlerProcess: إنشاء مثيل من CrawlerProcess, أمره بالتنقيح الروبوت newzspider3, وبدء عملية التنقيح.

ملخص

يعرض هذا الكود كيفية إعداد روبوت Scrapy الأساسي لتنقيح مقالات أخبار من موقع ويب، واستخراج تفاصيل ذات صلة، وتخزينها في ملف CSV بينما يتجنب النسخ المكررة بناءً على URL. يوضح استخدام ميزات Scrapy الأساسية، بما في ذلك تعريف الروبوتات، التعامل مع الطلبات HTTP، فحص المحتوى HTML، وإدارة سلاسل العناصر.

في الكود الذي قدمته، هناك مراحل رئيسية تؤدي إلى الحصول على ملف CSV يحتوي على بيانات متنقحة من موقع الويب. هذه المراحل هي:

1. تعريف الروبوت (Spider): يتم تعريف الروبوت newzspider3 الذي يحدد بداية التنقيح، الأطر الزمنية المسموح بها، وأرقام بداية التنقيح. الروبوت يبدأ بتنقيح الصفحة الرئيسية للموقع "<https://www.independent.co.uk/>".
2. استخراج الروابط: في الدالة parse، يتم استخراج الروابط من الصفحة الرئيسية التي تؤدي إلى مقالات أخبار فردية. يتم تجنب الروابط المرتبطة بالرموز. لكل رابط، يتم إرسال طلب متابعة إلى صفحة المقال.
3. معالجة صفحات المقالات: في الدالة parse\_details، يتم معالجة محتوى صفحة كل مقال. يتم استخراج النص، العنوان، اسم الكاتب، وموقع URL. يتم التحقق مما إذا كانت الرسالة موجودة بالفعل في ملف CSV لتجنب النسخ المكررة.
4. كتابة البيانات إلى ملف CSV: بعد استخراج البيانات وتحقق من عدم وجود نسخ مكررة، يتم كتابة البيانات إلى ملف CSV يدعى "results.csv". يتم فتح الملف في وضع القراءة لتحديد ما إذا كانت الرسالة موجودة بالفعل، ومن ثم فتح الملف مرة أخرى في وضع الكتابة لإنشاء السطر الجديد.
5. تنفيذ الروبوت: في الجزء الأخير من الكود، يتم إنشاء مثيل من CrawlerProcess، وتشغيل الروبوت newzspider3، والبدء في عملية التنقيح.

هذه المراحل تشكل العملية الأساسية لتنقيح البيانات من موقع الويب واستخدامها لإنشاء ملف CSV.