

스마트 스토어 크롤링 안내 - v.3.0 - 최종

[파일 돌리기 전 준비사항]

```
1  '''시작전 기본 세팅
2  모듈 설치
3  pip install openxyl
4  pip install requests
5  pip install beautifulsoup4
6
7  크롤링 하는 동안 엑셀 파일은 열지 말아주세요. 열려있는 경우 결과값이 저장되지 않습니다.
8  엑셀 파일과 파이썬은 같은 폴더안에 넣어주세요.
9  '''
```

[작업할 범위 지정]

파일 16 ~ 20 번째 줄에서 지정 해주시면 됩니다

(20 번째 줄은 몇 개의 링크를 parsing 후에 저장할건지 조정하는 숫자. 숫자가 낮을수록 크롤링 시간이 느려집니다.)

```
16  excel_file_name = "speed copy 5"      #작업할 엑셀 파일 이름
17  excel_sheet_name = "Sheet1"          #작업할 엑셀 파일 시트 이름
18  start_row_value = "F2"                #작업할 셀 시작 번호
19  end_row_value = "F50"                 #작업할 셀 마지막 번호
20  save_file_per = 30 #저장할 단위 지정
```

[추가 예외가 발생했을 경우]

파일 45~48 또는 50~53 과 같은 양식으로 아래에 붙여주시면 되며, if 문 안의 “상품이 존재하지 않습니다.” 부분을 변경해주시면 됩니다.

```
45  if(soup.title.text=="판매자의 사정에 따라 일시적으로 운영이 중지되었습니다."):
46      print("[F A I L]",cell,"판매자의 사정에 따라 일시적으로 운영이 중지되었습니다.")
47      totalFail+=1
48      continue
49
50  if(soup.title.text=="상품이 존재하지 않습니다."):
51      print("[F A I L]",cell,"상품이 존재하지 않습니다.")
52      totalFail+=1
53      continue
```

[기본 출력 세팅]

실패한 셀 값과 이유를 출력합니다.

마지막줄에는 성공한 개수와 작업에 걸린 시간을 보여줍니다.

```
[SUCCESS] <Cell 'Sheet1'.F34>
[SUCCESS] <Cell 'Sheet1'.F35>
[SUCCESS] <Cell 'Sheet1'.F36>
[SUCCESS] <Cell 'Sheet1'.F37>
[SUCCESS] <Cell 'Sheet1'.F38>
[SUCCESS] <Cell 'Sheet1'.F39>
[SUCCESS] <Cell 'Sheet1'.F40>
[SUCCESS] <Cell 'Sheet1'.F41>
[SUCCESS] <Cell 'Sheet1'.F42>
[SUCCESS] <Cell 'Sheet1'.F43>
[SUCCESS] <Cell 'Sheet1'.F44>
[SUCCESS] <Cell 'Sheet1'.F45>
[SUCCESS] <Cell 'Sheet1'.F46>
[SUCCESS] <Cell 'Sheet1'.F47>
[SUCCESS] <Cell 'Sheet1'.F48>
[SUCCESS] <Cell 'Sheet1'.F49>
[SUCCESS] <Cell 'Sheet1'.F50>
-----
Success / Total : 49 / 49
Total Time : 67.39122414588928
```

출력 세팅 SUCCESS 도 보고 싶으시면 150 번째 줄 주석 해제 하시면 됩니다.

```
188 #셀 하나 처리 완료될때마다 status확인 하고 싶으면 주석 삭제하시면 됩니다.
189 # print("[SUCCESS]",cell)
```

[엑셀파일에 저장되는 값]

1. 상품 이미지 주소
2. 상품 타이틀
3. 제품 등록일
4. 제품 세일 후 최종가격
5. 구매 건수, 최근 구매 건수 (구매건수 – Line 123, 최근 구매건수 – Line 130)

0 일때 저장하시고 싶으시면 아래 ""안을 0 으로 바꿔주시면 됩니다.

예시) resCumulationSaleCount = 0

```
#구매 건수
if(cumulationSaleCount[1] == '0'):
    resCumulationSaleCount = ""
```

6. 전체 리뷰 수, 전체 리뷰 평점

5 번과 동일

전체 리뷰 수 – Line 142, 전체 리뷰 평점 – Line 152

7. 제품 원가 (현재는 저장 안함)

제품 원가도 엑셀 파일에 저장하고 싶으시면 주석 해제 하시고 column 변경 해주시면 됩니다. (Line : 89~98, Line 160, Line 171)

* 엑셀파일에 열 추가할 경우 다른 저장되는 값들 column 값도 변경해줘야 합니다.

```
89      # 제품 원가 (할인 전 가격)
90      # if(sNP.count('salePrice')):
91      #     sNPParsed = sNP.split(',')
92      #     for foundData in sNPParsed:
93      #         if(foundData.count('salePrice') and oPFlag==0):
94      #             foundData = foundData.split(':')
95      #             # 엑셀에 저장하시려면 column 숫자 행번호에 맞게 수정하시면 됩니다. (행번호는 1부터 시작)
96      #             resOriginalPrice = foundData[1]
97      #             oPFlag=1
98
```

**엑셀파일에 저장안하고 싶을때. 순서에 맞게 해당 줄 주석 처리하면 됩니다.

(동일한 방식으로 161~169 도 같이 주석처리)

Line 149 : 사진 이미지 주소

Line 150 : 제품명

Line 151 : 등록일

Line 152 : 제품 원가격

Line 153 : 제품 현재 판매가격

Line 154 : 전체 구매 건수

Line 155 : 최근 구매 건수

Line 156 : 전체 리뷰 수

Line 157 : 전체 리뷰 평점

```
149      ws.cell(row = cell.row, column = 7, value=resPhotoSrc)
150      ws.cell(row = cell.row, column = 5, value=resTitleText)
151      ws.cell(row = cell.row, column = 3, value=resRegisterDate)
152      #제품 원가격(할인전) ws.cell(row = cell.row, column = 9999, value=resOriginalPrice)
153      ws.cell(row = cell.row, column = 13, value=resFinalSalePrice)
154      ws.cell(row = cell.row, column = 15, value=resCumulationSaleCount)
155      ws.cell(row = cell.row, column = 16, value=resRecentSaleCount)
156      ws.cell(row = cell.row, column = 17, value=resTotalReviewCount)
157      ws.cell(row = cell.row, column = 18, value=resTotalReviewScore)
160      #제품 원가격(할인전) ws.cell(row = cell.row, column = 9999, value=resOriginalPrice)
```