**1. 네이버 카페별 자사와 경쟁사의 데이터 수집**

**네이버카페\_데이터확보.xaml 전체 구조**

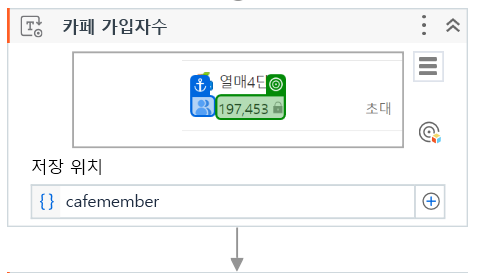
텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

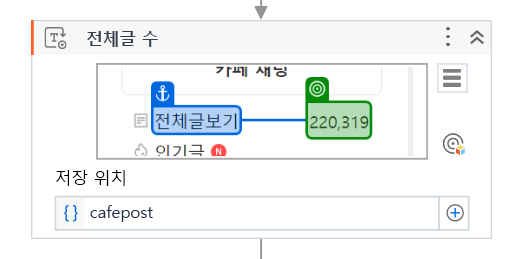
**1-1 가입자 수, 전체글 수, 방문객 수를 가져오고자 하는 카페를 Use Application/Browser를 통해 설정**



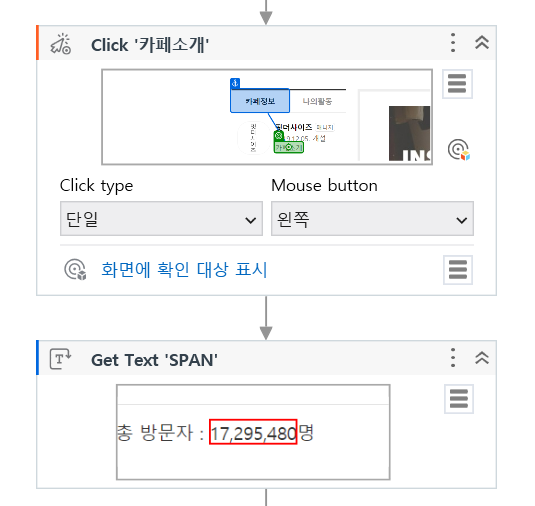
**1-2 카페의 메인 페이지에서 get text로 가입자 수 추출**



**1-3 카페의 메인 페이지에서 get text로 전체글 수 추출**



**1-4 방문객 정보를 확인하기 위해 click으로 카페소개 클릭 후 get text로 방문자 수 추출**

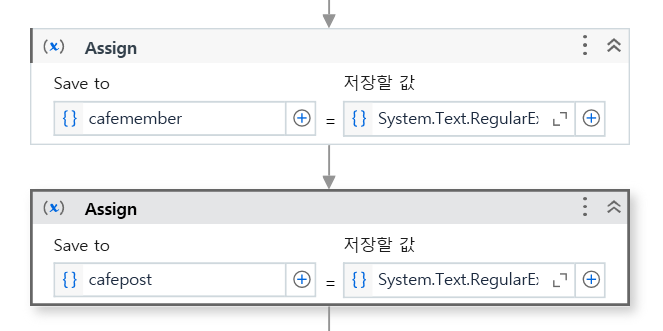


[예외사항] : 카페소개에서 가입자와 전체 게시글, 총 방문자를 동시에 확인하기는 selector를 수정해도 오류 발생율이 높아 전체글과 가입자는 메인 페이지에서 확인한다.

**1-5 추출한 가입자(cafemember)와 전체글의 수(cafepost)에서 숫자를 제외한 문자 제거**

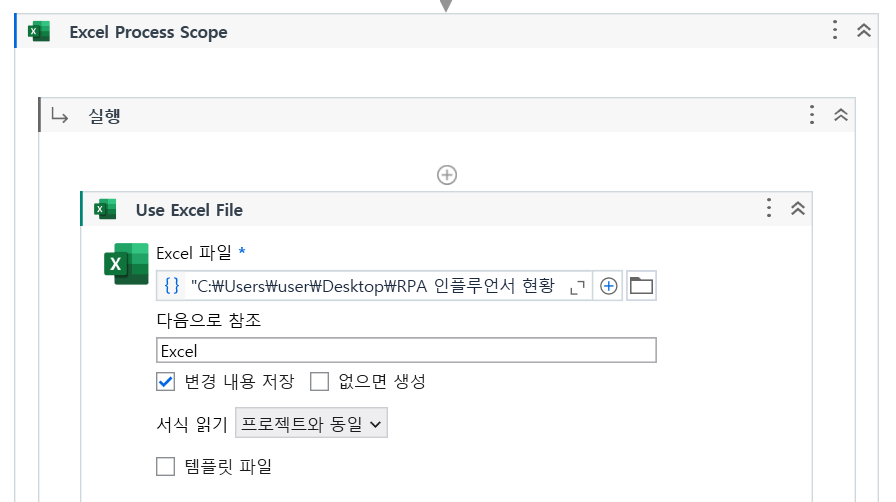
텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

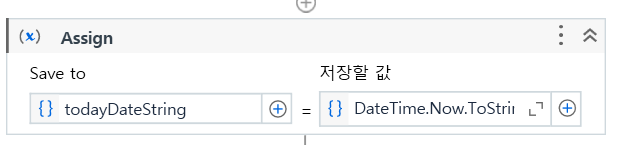


저장할 값에 각각 System.Text.RegularExpressions.Regex.Replace(cafemember, "[^0-9]+", String.Empty), System.Text.RegularExpressions.Regex.Replace(cafepost, "[^0-9]+", String.Empty)를 두어 정규 표현식을 사용하여 숫자가 아닌 모든 숫자([^0-9])가 하나 이상(+) 나오는 부분을 찾아 빈 문자열(String.Empty)로 대체한다.

**1-6 보고서를 작성할 엑셀파일 위치 설정**

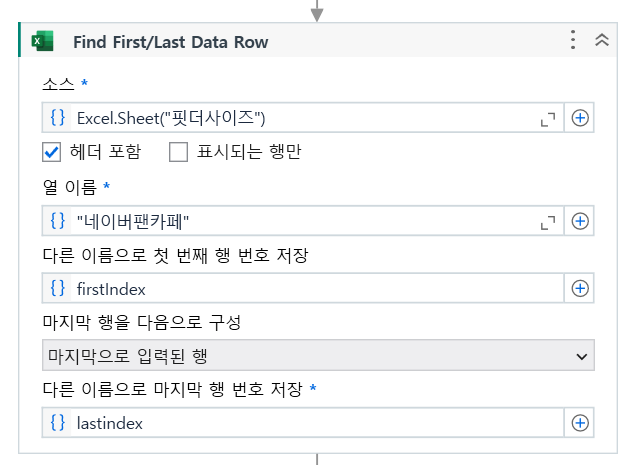


**1-7 오늘 날짜를 todayDateString에 설정**

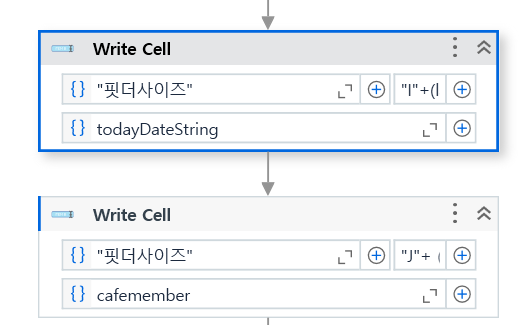


저장할 값에 DateTime.Now.ToString("yyyy-MM-dd")를 입력하여 오늘 날짜를 문자열로 설정한다.

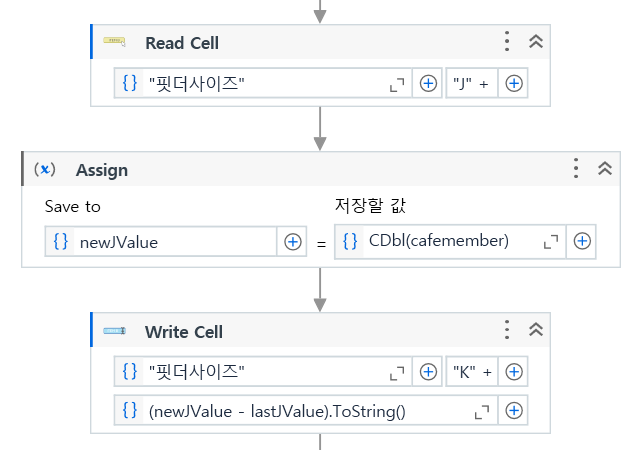
**1-8 Find First/Last Data Row를 통해 원하는 열의 마지막 셀 위치 획득**



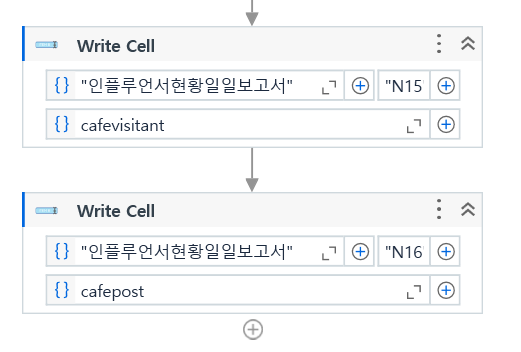
**1-9 [1-8]에서 얻은 셀의 위치를 기반으로 오늘 날짜 작성, 오른쪽 셀에 가입자 수 작성**



**1-10 가입자수를 작성한 셀의 위의 셀과의 비교를 통해 가입자 수의 변화량 계산**



**1-11 방문자 수와 전체글 수 보고서의 정해진 위치에 작성**



**1-12 같은 방식으로 경쟁 카페 정보 추출**



**2. 유튜브별 자사와 경쟁사의 데이터 수집**

**유튜브\_데이터확보.xaml 전체 구조**

**스크린샷, 텍스트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

**텍스트, 도표, 스크린샷이(가) 표시된 사진

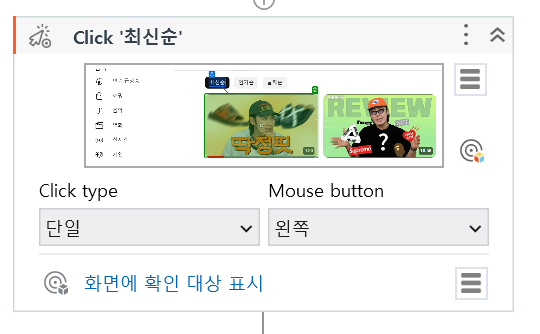
자동 생성된 설명**

**2-1 Use Application/Browser를 통해 유튜브 채널 설정**

스크린샷, 텍스트, 소프트웨어, 웹 페이지이(가) 표시된 사진

자동 생성된 설명

**2-2 최신 동영상 click을 통해 이동**

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

앵커로 최신순을 잡아 새로운 동영상이 생겨도 인식할 수 있도록 한다. 이후 지연을 3초 주어 에러가 발생하지 않도록 한다.

**2-3 get text를 통해 조회수, 구독자 수, 좋아요 수 추출**

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명 스크린샷, 텍스트, 라인이(가) 표시된 사진

자동 생성된 설명

조회수를 뽑는 Get Text 액티비티의 출력값은 views로 하고 구독자수 뽑는 get text는 subscriber로 출력값을 설정한다. 이때 구독자수 뽑는 Get text는 핏더사이즈 채널글자를 앵커로 설정한다. 좋아요수를 뽑는 Get Text 액티비티의 출력값은 likenumber로 한다.

[예외사항] : 유튜브의 조회수가 2.3만의 형태가 아닌 23,405의 형태인 경우 각각의 숫자를 따로 인식하여 selcetor를 수정하지 않으면 오류가 발생하는 경우가 생길 수 있다.

[예외사항] : 좋아요 수가 특정 값 일 때 selector로 추출하지 못하는 경우가 발생할 수 있다.

**2-4 스크롤 클릭하여 댓글이 보이는 화면으로 이동하여 댓글 수 추출**

텍스트, 스크린샷, 라인, 직사각형이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

댓글 글자를 앵커로 잡아 댓글 수를 Get Text 액티비티로 뽑아와 comment 문자열에 저장한다. 이후 구독자수, 좋아요수, 조회수, 댓글수 모두 추출이 되었는지 Message Box 액티비티로 확인한다.

[예외사항] : 네트워크 속도로 인하여 영상이 다 불러와지지 않음에도 스크롤 바를 클릭하면 잘못된 위치를 선택할 가능성이 있다.

**2-5 구독자 수에서 정확한 구독자 수 계산**

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

텍스트, 라인, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

구독자 수에서 .(마침표)가 찍혀 있을 경우 정확한 구독자 계산을 위해 IF 액티비티를 이용한다. 만약 구독자 수에 마침표를 포함하고 있다면 쉼표, 공백, 구독자 글자, 명 글자, 마침표를 제거하고 만 글자는 0000으로 대체하여 subscriber에 저장한다.

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

텍스트, 폰트, 라인, 스크린샷이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

만약 마침표를 포함하지 않는다면 쉼표, 공백, 구독자 글자, 명 글자를 제거하고 만 글자는 0000으로 대체하여 subscriber에 저장한다. 그 후 정규 표현식을 이용하여subscriber의 마침표와 그 뒤 숫자가 있다면 이를 제거하도록 하고 다시 subscriber에 저장한다.

**2-6 조회수에서 정확한 조회수 계산**

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

텍스트, 폰트, 라인, 스크린샷이(가) 표시된 사진

자동 생성된 설명

만약 조회수에서 마침표를 포함하고 있다면 쉼표, 공백, 조회수 글자, 회 글자, 마침표를 제거하고 만 글자는 000으로 대체하여 view에 저장한다.

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

텍스트, 폰트, 라인, 스크린샷이(가) 표시된 사진

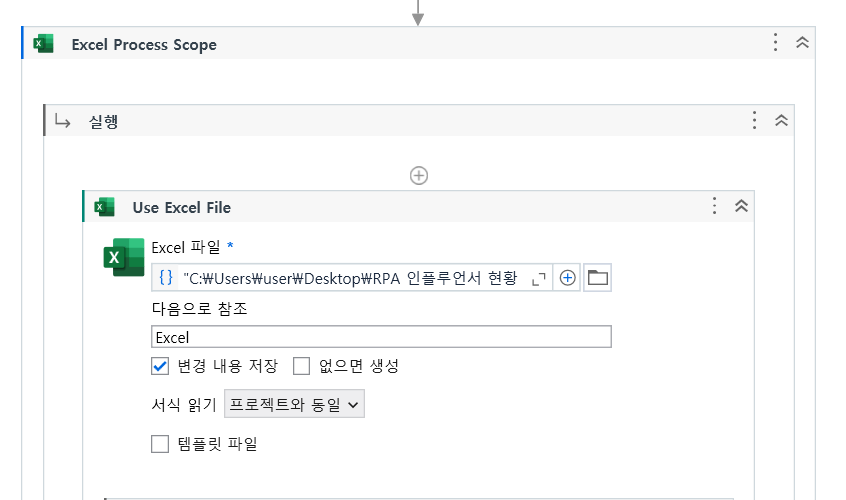
자동 생성된 설명

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

마침표를 포함하고 있지 않다면 쉼표, 공백, 조회수 글자, 회 글자를 제거하고 만 글자를 0000으로 대체하여 views에 저장한다. 그 후 정규 표현식을 이용하여 마침표과 그 뒤의 숫자를 제거하도록 한다.

**2-7 보고서를 작성할 엑셀파일 위치 설정**

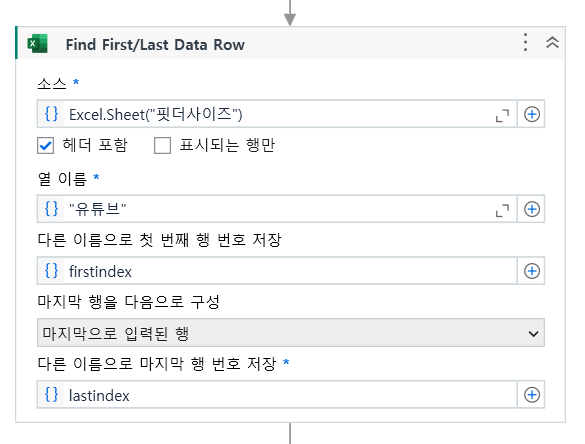


**2-8 오늘 날짜를 todatDateString에 설정**

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

**2-9 Find First/Last Data Row를 통해 원하는 열의 마지막 셀 위치 획득**



**2-10 [2-9]에서 얻은 셀의 위치를 기반으로 오늘 날짜 작성, 오른쪽 셀에 구독자 수 작성**

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

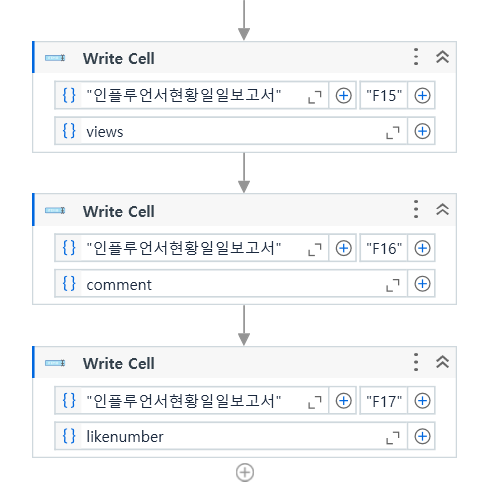
자동 생성된 설명

**2-11 구독자 수를 작성한 셀의 위의 셀과의 비교를 통해 구독자 수의 변화량 계산**

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

**2-12 조회수, 댓글 수, 좋아요 수 보고서의 정해진 위치에 작성**



**2-13 같은 방법으로 경쟁 유튜브 정보 추출**

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

**3. 통계 사이트 데이터 랩에서의 남성 의류 검색어 순위 데이터 추출 및 가공**

**검색어 추출.Xaml 전체 구조**

텍스트, 번호, 스크린샷이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 영수증, 디자인이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

**3-1. 데이터 랩 웹 페이지로 접근하기**

텍스트, 스크린샷, 소프트웨어, 컴퓨터 아이콘이(가) 표시된 사진

자동 생성된 설명

* 데이터 랩에서의 특정 조건에 따른 카테고리 정보를 가져오기 위해서 이와 **같이 Open Browser 및 User Application/Browser Activity를 사용**해 "https://datalab.naver.com/"의 **데이터 랩 주소로 이동**한다.

**3-2. 10대, 20대 남성 의류 카테고리 정보를 확인하기**

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 디스플레이, 번호이(가) 표시된 사진

자동 생성된 설명

* 데이터 랩에 접근한 뒤, 남성 의류 및 연령 층 조건에 맞게 필터링하기 위해서 **Click Activity사용**으로 **[쇼핑 인사이트 🡺 2 분류 🡺 남성 의류 🡺 10대, 20대 🡺 조회하기] 순서**에 맞게 선택해 원하는 정보에 접근한다.

**3-3. 조건에 맞는 순위 데이터 첫 번째 Page 추출하기**

텍스트, 스크린샷, 소프트웨어, 번호이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

* 2-2에서 설정한 조건에 따라 **Extract Table Data Activity를 사용**해 검색어 TOP500 데이터를 가져오도록 한다. 테이블 데이터를 가져올 때, 검색어만 Element로 지정해주어 해당 테이블 검색어 정보들만 가져올 수 있도록 한다. 가져온 **데이터를 ‘KeyworldTable’이라는 DataTable 형식의 변수로 저장**하게 된다.

**[예외사항]** – ‘스투시반팔’, ‘자라원피스’와 같이 브랜드 명이 포함되어 있는 제품을 선정하게 될 경우, 나중에 무신사에서 데이터를 가져오게 될 때 해당 브랜드의 의류만 추천하게 될 우려가 있어 제외해 추출해오게 된다. 그러나 그 양이 많아 2 Page까지 총 40개의 검색어 데이터를 추출해 저장한다.

**3-4. 조건에 맞는 순위 데이터 두 번째 Page 추출하기**

텍스트, 스크린샷, 소프트웨어, 번호이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 디스플레이, 번호이(가) 표시된 사진

자동 생성된 설명

* 20개의 데이터만 제공하는 TOP 500 테이블 데이터에서 두 번째 페이지로 넘어가 나머지 20개의 데이터를 추출한다. **Click Activity를 사용해 두 번째 Page**로 넘어가고, 2-3과 동일하게 해당 **검색어 데이터를 추출해 ‘KeywordTable\_’ 이름의 DataTable 형식의 변수에 저장**한다.

**3-5. 추출한 Data Table 병합하기**

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

* 첫 번째 Page의 20개 데이터와 두 번째 Page의 20개 데이터 테이블을 같이 사용하기 위해 **Merge Data Table Activity를 사용하여 병합**해준다. 두 번째 Page의 데이터들을 첫 번째 Page의 Data Table로 병합시킨다.

**3-6. 추출한 40개의 데이터 가공하기**

텍스트, 스크린샷, 번호, 디스플레이이(가) 표시된 사진

자동 생성된 설명

* 40개의 데이터가 저장된 KeywordTable 데이터 **테이블에 For Each Row in Data Table Activity를 사용해 각각의 Data를 1차적으로 가공**한다. Assign Activity를 사용하여 **currentRow(0).toString 수식을 걸어 해당 Data를 우선 문자열로 변환**한다. 이후 **‘System.Text.RegularExpressions.Regex.Replace(searchKeyword, "[0-9]", "").Trim’** 수식 사용으로 해당 Data의 **숫자를 제거하고 공백을 없애는 작업을 수행**하게 된다.

**3-7. 가공한 데이터를 지피티에 학습시키기 위해 메모장에 정리하기**

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

* 앞에서 **Assign Activity를 사용**하여 가공한 40개의 데이터들이 **Send Hotkey Activity를 통해 빈 메모장에 반복 입력**되도록 한다. 추출한 40개의 데이터들이 브랜드 명을 포함하기도 하고, 중복된 데이터들도 많아 이를 UI Path로 수정하기보다는 생성형 AI 모델을 활용하여 가공 후, 최종 순위 키워드를 추출한다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

* 이와 같이 **각각의 예외사항**들을 **Chat GPT에게 학습**시키기 위해 메모장에 입력해준 40개의 데이터 뒤에 추가로 학습 내용을 같이 입력하게 만든 뒤, Ctrl + A와 Ctrl + C를 통해 전체를 복사한다.

**3-8. Chat GPT를 웹 사이트에 접근하기**

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

* 40개의 데이터 가공을 위한 학습 진행을 위해서 **Open Browser 및 Use Application /Browser Activity를 활용**하여 "https://chatgpt.com/"의 주소로 접근한다.

**3-9. Chat GPT의 가공 후, 최종 5위까지의 순위 Data 확인하기**

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

* Click Activity를 활용하여 GPT의 Input Field를 선택한 뒤, 그 Field에 Send Hotkey Activity를 사용하여 위 **3-7**에서 복사해준 Data 및 학습 자료를 Ctrl + V로 GPT에 붙여 넣고, 학습을 진행한다. 이 과정에서 학습 시간이 걸릴 것을 대비해 **8초간의 Delay**를 주게 된다.

**3-10 GPT를 통한 5위 순위 데이터 복사 후 가공하기**

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명텍스트, 폰트, 스크린샷, 화이트이(가) 표시된 사진

자동 생성된 설명

* GPT가 학습을 하여 결과를 도출하면 오른쪽 예시 사진과 같이 브랜드 명이나 중복 데이터를 포함하지 않는 **결과물을 도출**하는 것을 알 수 있다. **Buil Data Table Activity를 활용해 나중에 분할한 데이터를 저장해줄 DataTable finalTable을 생성**해준다.
* 결과로 등장하게 된 데이터들을 Copy 버튼을 Click Activity를 활용하여 선택하고, 그를 **Get From Clipboard Activity를 사용해 String 형식 변수 ‘Finalpod’에 저장**해준다. 이후, 이 다섯 개의 Data가 저장된 문자열 **Data Finalpod를 Assign Activity를 활용하여 각각의 줄을 기준으로 하나하나 분할해주는 과정을 수행**한다.
* **Finalprod.Split({Environment.NewLine}, StringSplitOptions.RemoveEmptyEntries)** 이와 같이 공백은 제거하고 줄 바꿈을 기준으로 분할해주는 수식을 적용해 **각각을 따로 저장해서 Lines라는 Array[String] 형식의 변수에 저장**해준다.



**3-11. 분할된 Data를 최종 엑셀 파일 ‘데이터 랩’ 시트에 입력하기**

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

* 앞에서 다섯 개의 Data가 저장된 Lines라는 Array 변수 속에서 **For Each를 활용해 Lines 속Data 하나하나 Add Data Row를 활용하여 앞에서 생성해준 finaltable 데이터 테이블 변수에 추가**한다.
* 다섯개의 Data를 모두 추가하게 되면 For Each Activity를 빠져나오게 되고, 해당 Data Table Data들을 “데이터랩” 시트에 Write Range Activity를 활용해서 적어준다.

**4. 패션플랫폼인 무신사에서 추출된 검색어별 상위 4개 제품 정보 수집**

**무신사\_검색.xaml 전체 구조**

**텍스트, 스크린샷, 번호, 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

**4-1. 엑셀 파일에서 추출된 검색어 정보 가져오기**

텍스트, 스크린샷, 디스플레이, 번호이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

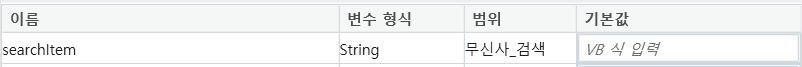
* **Excel Application Scope 액티비티**로 RPA 팀플 마케팅\_최종 엑셀 파일을 가져온다.
* **Read Range 액티비티**로 데이터랩 시트를 가져오고 출력되는 데이터 테이블로는 **datatable**이라는 새로운 변수를 생성한다.

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명 텍스트, 폰트, 스크린샷, 번호이(가) 표시된 사진

자동 생성된 설명

<엑셀 中 데이터랩 시트>



* **For Each Row in Data Table** **액티비티**로 datatable의 각 행들(CurrentRow)을 차례로 읽는다.
* **Get Row Item** **액티비티**로 엑셀의 2번째 열(B열, 열은 0부터 셀 수 있어 열에는 1을 입력해줌)의 행들을 문자열 변수 형식의 **searchItem**에 저장한다. 즉 검색어의 행들을 문자열 변수에 저장하는 것이다.

**4-2. 무신사 웹사이트에서 검색어별 상위 4개 제품 검색해서 정보 수집하기**

텍스트, 스크린샷, 번호, 도표이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 도표, 번호이(가) 표시된 사진

자동 생성된 설명

* **Use Browser Edge 액티비티**로 무신사 웹사이트에 접속한다.
* **Click 액티비티**로 검색 아이콘을 클릭한다.
* **Type Into 액티비티**로 텍스트에 searchItem + “[k(enter)]” 를 입력하여 추출된 검색어를 차례차례 입력하고 검색되도록 한다.
* **Click 액티비티**로 상품쪽 카테고리에서 상품의 필터를 판매순-1주일로 걸도록 한다.

검색어별 상위 4개의 제품 정보를 수집하기 위해 상단 메뉴바의 텍스트, 폰트, 스크린샷, 디자인이(가) 표시된 사진

자동 생성된 설명기능을 이용한다.

텍스트, 스크린샷, 소프트웨어, 컴퓨터 아이콘이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

검색어 검색 페이지에서 제품명, Url, 가격, 브랜드명을 추출한다. 이때 각 검색어 별 제품 수는 최대 4개 행으로 해야 검색어별로 4개의 제품만을 수집할 수 있다.

**[예외사항]**  일부 제품들은 좋아요수와 후기수는 9,999개를 초과하면 9,999+로 표시되어 각 제품 상세페이지를 들어가야 개수가 정확히 얼마인지 판단할 수 있다. 따라서 이후 과정에서 상세페이지를 들어가 제품의 사진과 좋아요수, 후기수를 파악할 예정이다.

**텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

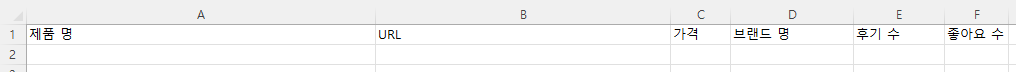
자동 생성된 설명**

* 위 테이블 추출이 완료되면 **Extract Table Data 액티비티**가 생성되게 된다. 추출된 정보는 **ExtractDataTable** 라는 DataTable에 저장한다.

텍스트, 폰트, 라인, 번호이(가) 표시된 사진

자동 생성된 설명

* **Append Range Workbook 액티비티**로 ‘RPA 팀플 마케팅\_최종’ 엑셀파일의 ‘무신사’시트에 ExtractDataTable 내용이 추가되도록 한다. 작성된 내용이 한눈에 들어올 수 있도록 아래와 같이 미리 무신사 시트에 머리글을 추가하고 진행하였다.



* 여기까지 과정이 완료되면 아래와 같이 엑셀파일의 무신사 시트에 제품 명, URL, 가격, 브랜드 명이 들어온 것을 알 수 있다.

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

**4-3. 검색어별 상위 4개의 제품 상세정보 수집하기**

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

* **Build Data Table 액티비티**로 데이터 테이블 칼럼으로 빈 문자열 타입의 Stars, Likes을 둔 starlike 데이터테이블을 생성한다.

텍스트, 스크린샷, 디스플레이, 번호이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

* **Excel Application Scope 액티비티**로 ‘RPA 팀플 마케팅\_최종’ 엑셀 파일을 불러와 **Read Range 액티비티**로 무신사 시트 B2:B21 범위를 읽어와 Itemurl 데이터 테이블에 저장한다.

**텍스트, 스크린샷, 소프트웨어, 디스플레이이(가) 표시된 사진

자동 생성된 설명**

* **For Each Row in Data Table 액티비티**로 Itemurl 데이터테이블의 각 행들(CurrentRow)을 차례로 읽는다.
* **Use Browser Edge 액티비티**로 엣지 홈화면을 연결한다.
* **Type Into 액티비티**로 엣지 홈화면의 웹주소 검색창에 **CurrentRow(0).ToString() + “[k(enter)]”** 를 입력한다. 이는 무신사 시트의 B열인 URL을 이용하여 검색어별 상세페이지로 가는 것이다.

**[예외사항]** 본래는 바로 무신사 시트의 url을 이용하려고 했지만 에러가 나서 엣지 홈화면의 웹주소 검색창을 사용해서 무신사의 제품들의 상세 페이지를 이동하도록 했다.

텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

* **Take Screenshot 액티비티**로 제품 상세페이지에서 제품 사진을 찍어서 저장한다. 화면에 타겟 표시할 때 앵커로 스케치, 도표, 디자인이(가) 표시된 사진

  자동 생성된 설명를 잡아 다른 제품이어도 제품 사진을 찍을 수 있도록 한다. File name은 **"screenshot\image " + imageCounter.ToString() + ".jpg"**로 하여 screenshot이라는 폴더 안에 ‘image+순번+.jpg’ 가 될 수 있도록 사진 이름을 설정한다. 모든 제품 사진의 이름이 다를 것이기 때문에 Auto increment는 없음, 인덱스, 날짜로 설정할 수 있는 기능이 있는데 이는 무엇으로 해도 상관없다.

이때 주의할 것은 imageCounter 변수를 생성할 때 정수형(Int32) 타입으로 하고 기본값에 1을 설정한다.



* **Assign 액티비티**로 한번 반복될 때마다 Imagecounter가 1씩 증가하도록 하여 제품 사진의 이름이 순서대로 저장될 수 있도록 한다.

텍스트, 스크린샷, 디스플레이, 번호이(가) 표시된 사진

자동 생성된 설명

* **Get Text 액티비티**로 제품 상세페이지에서 ‘후기+숫자’ 가 있는 칸을 잡고 앵커로는 옆에 있는 ‘추천’을 잡는다. 앵커를 잡아야 다른 제품 상세페이지여도 후기수를 인식할 수 있다. 이는 stars라는 문자열 변수에 저장되도록 한다.
* **Get Text 액티비티**로 제품 상세페이지에서 좋아요수를 잡고 앵커로는 그 위의 하트를 잡는다. 이는 likes라는 문자열 변수에 저장되도록 한다.

**[예외사항]** 처음에는 구매하기인 파란색 아이콘(스크린샷, 일렉트릭 블루, 텍스트, 로고이(가) 표시된 사진

자동 생성된 설명)을 앵커로 잡고 진행했었는데 프로그램 구현 도중 제품 중 앱회원전용인 구매하기 파란색 아이콘()을 가진 제품이 있어 오류가 나 앵커로 를 잡아 진행하였다.

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

* **Add Data Row 액티비티**로 starlike 데이터테이블의 첫번째 열과 두번째 열에 각각 stars, likes의 행들을 추가한다.

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

* **Write Range Workbook 액티비티**로 ‘RPA 팀플 마케팅\_최종’ 엑셀파일의 무신사 시트에 **E2**에서부터 **starlike** 데이터 테이블의 데이터를 저장한다.

결과로 screenshot 폴더에는 검색어별 상위 4개의 제품들이 차례로 저장된다.

텍스트, 스크린샷, 웹사이트, 웹 페이지이(가) 표시된 사진

자동 생성된 설명

RPA 팀플 마케팅\_최종 엑셀파일의 무신사 시트에는 후기 수와 좋아요 수가 채워져 있는 것을 볼 수 있다.

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

**5. Chat GPT를 활용한 차트 생성 및 보고서 기입**

**차트생성\_보고서기입.xaml 전체 구조**

**텍스트, 스크린샷, 디자인이(가) 표시된 사진

자동 생성된 설명**

**텍스트, 스크린샷, 소프트웨어, 디자인이(가) 표시된 사진

자동 생성된 설명**

**5-1. Assign을 통한 변수 Defualt 값 부여**

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

1. todayStr : 오늘 날짜를 yyyyMMdd 형식의 String으로 저장한다.

2. filepath1 : 데이터랩, 무신사, 유튜브 등에서 수집된 데이터를 입력하는 엑셀 위치를 저장한다.

3. filePath2 : 차트 생성에 필요한 데이터가 입력된 csv파일의 위치를 저장한다.

4. uploadFile : Chat GPT에 업로드하는데 필요한 csv, xlsm파일의 위치를 합쳐 저장한다.

**※ 예외사항** : 폴더위치 입력창에서 “위치”의 형태로 입력하기 위해서는 “”위치””의 형태로 uploadFile 변수에 저장되어야 하나, Assign 문법상 불가능 하여 외부의 “”를 chr(34)로 대체한 후에 + 로 연결하여 해결하였다.

**5-2. Chat GPT 활용**

텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

Use Browser Chrome을 통해 미리 만들어둔 Chat GPT를 이용한다.

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

GPT Pro의 나의 GPT 만들기를 이용해 원하는 형식의 차트를 형성할 수 있도록 지침을 부여하고, 사용자 편의성을 위한 스타터 버튼을 구성하였다. Data visualization Bot은 사용자가 업로드하는 파일들을 지식으로 활용하여 차트를 제작하게 된다.

**※ 예외사항** : GPT가 의도치 않은 답변을 출력할 때마다 지침을 수정해 보완하였다.

**5-3. Try catch와 If를 이용한 로그인 ->14주차 Error and Exception Handling 이용**

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

Try catch를 통해 로그인을 시도한다. Elments Exists를 통해 회원가입 즉, 비로그인 상태일 때 발생하는 버튼 유무를 notloggedin 변수에 boolean형식으로 저장한다.

텍스트, 스크린샷, 디스플레이, 번호이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 디스플레이, 번호이(가) 표시된 사진

자동 생성된 설명

Notloggedin이 True일 때, 버튼을 클릭하고, 회원가입이 아닌 로그인 링크를 선택해 이동한다.

텍스트, 스크린샷, 디스플레이, 번호이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 디스플레이, 번호이(가) 표시된 사진

자동 생성된 설명

이후 ID와 PW를 Type into로 입력한 후에 “[k(enter)]”와 click을 이용해 로그인한다.

텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

로그인이 되어있었거나 로그인에 성공하였다면 Data visualization Bot에 기본적으로 탑재된 Upload files. 버튼을 눌러 상호작용을 시작한다. 이때 GPT는 사용자에게 파일을 업로드해달라는 대화문을 출력하도록 되어있다.

**5-4. Chat GPT에 파일 업로드**

텍스트, 스크린샷, 디스플레이, 번호이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 번호, 디스플레이이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 디스플레이, 번호이(가) 표시된 사진

자동 생성된 설명

이후 Click과 Type into를 반복적으로 사용해 파일을 업로드하는 절차를 마친다. 파일명 입력 후에 검색되는 시간과 파일이 로드 되는 시간을 고려해 Delay를 활용해 여유시간을 확보하였다.

**※ 예외사항** : ID와 마찬가지로 “[k(enter)]”를 활용해 파일업로드를 즉시 완료하려고 하였으나, 파일이 로드 되는 시간이 반드시 필요했기 때문에 Click과 Delay를 활용하였다.

텍스트, 스크린샷, 디스플레이, 번호이(가) 표시된 사진

자동 생성된 설명

파일이 로드 된 후에 커서는 자동적으로 대화 입력창에 있기 때문에 대화 입력창에 엔터키를 입력해 파일을 GPT에 업로드한다. 사용자가 업로드한 파일에 대해서 GPT가 받아들이고, 파일 업로드가 완료되었다는 문구를 출력하는 데까지 Delay 10초를 부여했다.

**5-5. Chat GPT를 통한 차트 제작 상호작용**

텍스트, 스크린샷, 번호, 디스플레이이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 번호, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

이후 GPT는 기존에 학습시킨 대로 차트 1~4까지 제작을 하게 된다. GPT에게 학습시킨 내용은 차트 1은 사용할 데이터의 위치, 차트 번호별 차트 형식 등이 있다. Type into를 통해 GPT에게 먼저 차트 1,2의 제작을 요청하고 약 3분의 delay를 부여한다. 차트 3,4와 차트 다운로드 링크 생성도 마찬가지로 3분의 delay를 부여한다.

**※ 예외사항** : GPT가 차트 생성에 소요되는 시간이 Try마다 다르다. 보통 1분내로 끝내지만 예외 사항으로 더 오랜 시간이 걸린다면 다음 요청사항을 입력할 수 없다. 그렇기 때문에 대기시간을 3분으로 충분히 부여하였다. Element exists를 활용해 대답을 마쳤을 때 등장하는 아이콘을 식별하여 진행할 수도 있었지만, Selector를 지정할 수 없었기에 시간적 여유를 부여했다.

텍스트, 스크린샷, 도표, 평행이(가) 표시된 사진

자동 생성된 설명

차트 생성이 완료되면 yyyyMMdd의 형식으로 각 차트들을 압축하여 다운로드 하이퍼 링크를 사용자에게 전달한다. Click 액티비티를 이용해 이를 다운로드 한다.

**※ 예외사항** : Modern 인터페이스의 Click에서 Ui explorer를 활용해 Selector를 수정하였으나 다운로드 링크를 클릭할 수 없었다. 이에 따라 Classic 인터페이스의 Anchor Base 액티비티를 활용하여 Find element로 앵커를 지정하고, Click으로 다운로드 링크를 클릭하는 것으로 구현했다.

**5-6. 다운로드된 zip파일 압축 해제**

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

filePath3 : 다운로드 된 zip파일의 위치를 저장

extractFilePath : 압축해제 후 저장할 폴더 위치를 저장

Ivoke Method 에서 Target type을 Systme.IO.Compression.Zipfile로 지정하여 압축해제를 진행한다.

**5-7. 완성된 차트 및 수집된 사진을 엑셀에 탑재**

텍스트, 스크린샷, 번호, 소프트웨어이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 대수학이(가) 표시된 사진

자동 생성된 설명

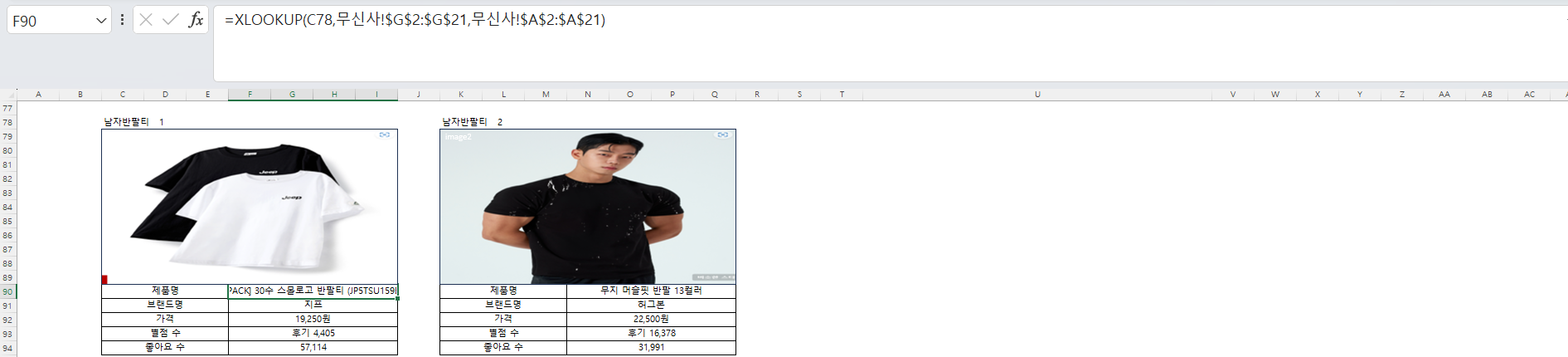
일련의 과정을 통해서 수집된 차트와 사진 데이터들을 보고서에 탑재한다. Use Excel File을 통해 엑셀을 연 후에 매크로가 지정되어 있는 도형을 클릭한다. 사진이 들어갈 위치에 알맞게 도형을 배치한 후에 도형에 Chart 1, Chart 2 ~ Chart 4와 image 1 ~ image 20까지 이름을 부여하였다. VBA코드를 통해서 각 이름에 맞는 도형을 for문을 통해 반복하여 삽입하도록 매크로를 구현하였다.

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

이후 Send hotkey를 통해 Ctrl S (저장), Alt F4 (종료)를 입력함으로써 워크플로우가 종료된다.

**추가. 함수와 VBA의 활용**

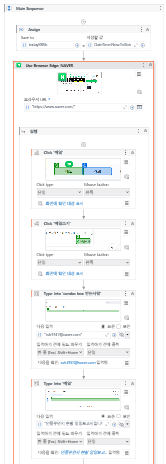
****

엑셀 보고서 작성 과정에서 함수와 VBA를 적극 활용하였다. 품목과 차트의 사진은 5-7번에서 사용된 VBA코드를 작성하여 제목에 매크로 버튼을 형성함으로써 자동적으로 채우는 기능을 구현하였다. 사진 좌측 상단의 카테고리와 인덱스 또한 함수로 자동 부여되며, 데이터랩 시트가 업데이트 됨에 따라 변한다. 인기 상품에 대한 정보도 무신사 시트의 정보가 업데이트됨에 따라 자동으로 변한다.

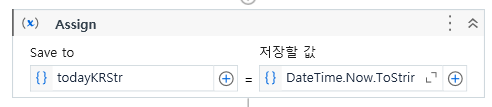


자동화를 위한 기초 파일이 오류 발생율이 높은 XLSM (매크로 포함 문서)이기 때문에 XLSX형식의 양식을 하나 더 만들어 공유링크로 연결하였다. 이에 따라 XLSX파일이 업데이트 되면 XLSM파일 또한 동시에 업데이트 된다. 데이터 랩도 마찬가지로 연결되어 있다.

**6. 메일 발송**

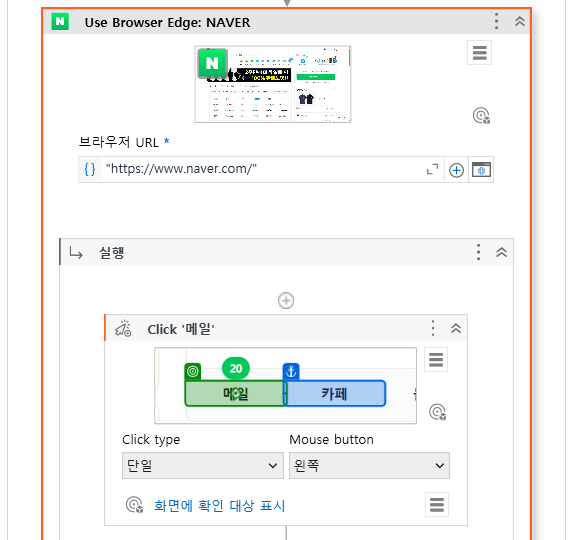
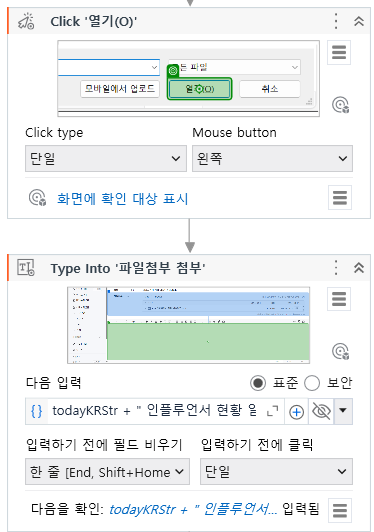
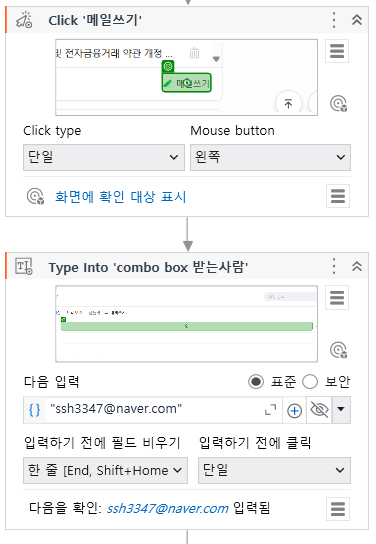
 

**6-1. Assign을 통한 날짜 변수 설정**



메일 작성 시, 제목 및 내용에 포함하기 위한 날짜 변수를 생성하였다.

**6-2. Use Browser를 이용한 메일 발송**

Use Browser 내의 실행에서 Click과 Type into를 통해서 메일 발송을 자동화 하였다. Click을 통해서 메일 탭을 눌러 들어가 메일쓰기를 클릭, 주소창을 클릭하고 보낼 이메일을 입력하였다. 그 다음 내 피씨를 눌러 보고서의 고정 저장 주소를 파일명에 입력하여 파일을 첨부하였다. 이후 20240617 인플루언서 현황 일일 보고서 입니다. 라는 문구를 내용에 적고 발송을 완료하였다.