

Microsoft Azure

# Azure Synapse Analytics

## Hands-on LAB

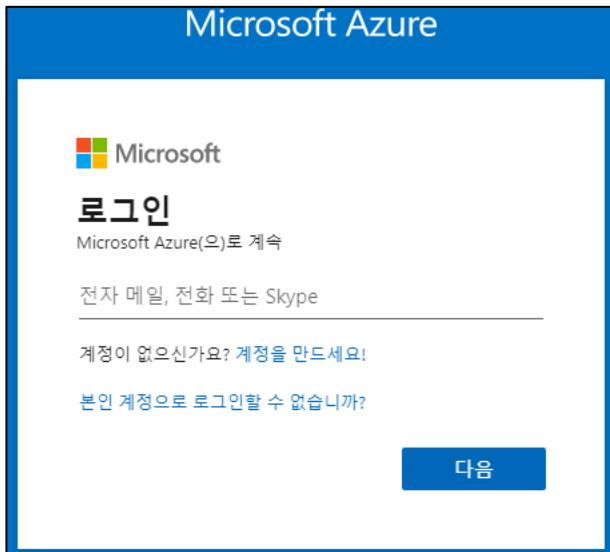
## 목차

Lab 1: HOL 환경 설정.....	2
Task 1. Azure Synapse Workspace 생성.....	2
Task 2. Cosmos DB 생성 .....	4
Task 3. ADLS Gen2에 컨테이너 및 데이터 생성.....	11
Task 4. SQLPOOL, SPARK POOL 생성.....	16
Task 5. Azure Synapse Workspace Studio 접속.....	21
Lab 2: External Table을 이용한 테이블 LOAD(Polybase).....	23
Task 1. 작업 Script 불러오기 .....	23
Task 2. 테이블 생성, 데이터 Load, 데이터 확인.....	25
Lab 3: Pipeline을 이용한 원격 DB 데이터 LOAD(ADF방식).....	31
Task 1. Source 연결 설정 .....	31
Task 2. Target 연결 설정 .....	35
Lab 4: 데이터 조회.....	42
Task 1. SQL Script를 이용한 데이터 조회 .....	42
Task 2. DataFrame을 이용한 데이터 조회 .....	44
Task 3. 외부 데이터 조회 .....	46
Lab 4: Cosmos DB 데이터 조회.....	51
Task 1. Cosmos DB 연결 .....	51

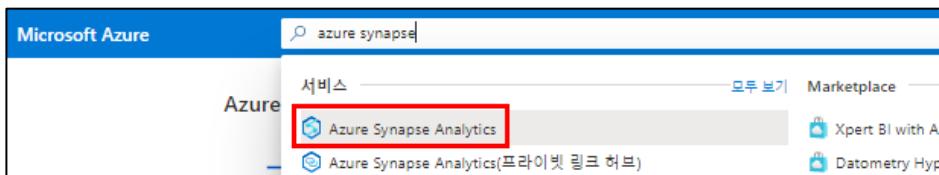
## Lab 1: HOL 환경 설정

### Task 1. Azure Synapse Workspace 생성

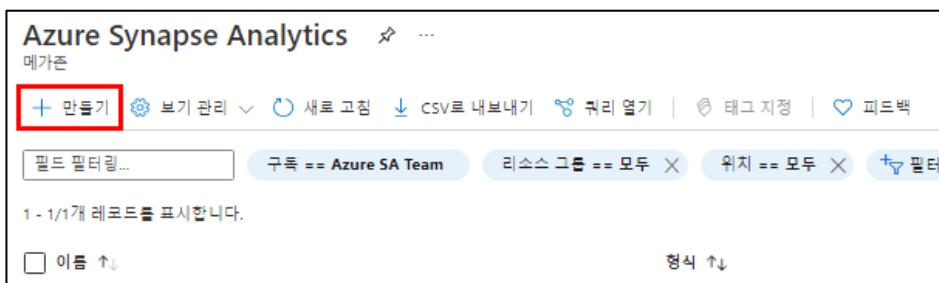
1. Azure Portal(<https://portal.azure.com/>)에 접속하여 개인 혹은 제공된 아이디로 로그인 합니다.



2. 상단의 둘보기에서 Azure Synapse Analytics를 입력하여 선택합니다.



3. Azure Synapse Analytics 만들기를 클릭하여 생성할 리소스의 정보를 입력합니다.



4. 기본사항에 필요한 정보를 입력합니다.

## [Azure Synapse Analytics Hands-on Labs]

\* 기본 사항 \* 보안 네트워킹 태그 검토 + 만들기

Synapse 작업 영역을 만들어 몇 번의 클릭만으로 엔터프라이즈 분석 솔루션을 개발할 수 있습니다.

프로젝트 정보

배포된 리소스와 비용을 관리할 구독을 선택합니다. 폴더 같은 리소스 그룹을 사용하여 모든 리소스를 정리 및 관리합니다.

구독 \* ○ Azure SA Team

리소스 그룹 \* ○ (신규) gildong-rg

관리되는 리소스 그룹 ○ 관리되는 리소스 그룹 이름 입력 ✓

작업 영역 정보

작업 영역의 이름을 지정하고 위치를 선택하고, 로그 및 작업 출력의 기본 위치로 사용할 기본 Data Lake Storage Gen2 파일 시스템을 선택합니다.

작업 영역 이름 \* gildong-synapse ✓

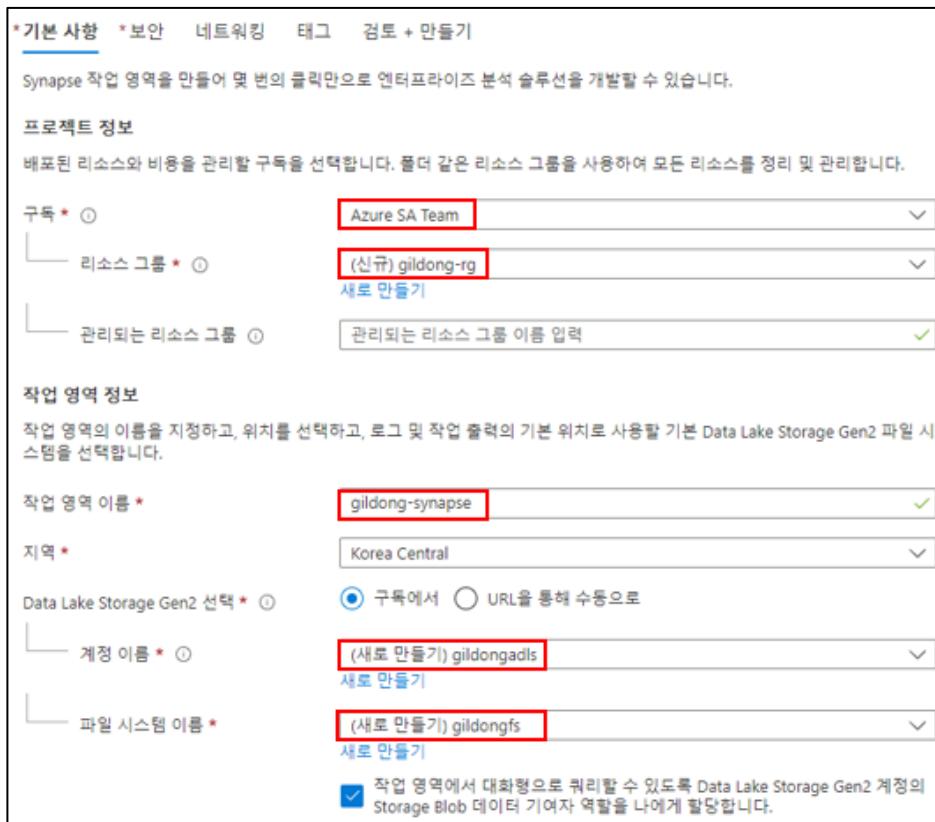
지역 \* Korea Central

Data Lake Storage Gen2 선택 \* ○ 구독에서 ○ URL을 통해 수동으로

계정 이름 \* ○ (새로 만들기) gildongadls

파일 시스템 이름 \* ○ (새로 만들기) gildongfs

작업 영역에 대화형으로 쿼리를 할 수 있도록 Data Lake Storage Gen2 계정의 Storage Blob 데이터 기여자 역할을 나에게 할당합니다.



리소스 그룹 : <username>-rg (새로 만들기)

작업 영역 이름 : <username>-synapse (고유 이름값)

지역 : 한국 중부

Data Lake Storage Gen2 계정 이름 : <username>adls (새로 만들기)(고유 이름값)

Data Lake Storage Gen2 파일 시스템 이름 : <username>fs (새로 만들기)

### 5. 보안에서 SQL admin 관리자 계정 정보를 입력합니다.

\* 기본 사항 \* 보안 네트워킹 태그 검토 + 만들기

작업 영역의 보안 옵션을 구성합니다.

인증

SQL 풀과 같은 작업 영역 리소스에 액세스하기 위한 인증 방법을 선택합니다. 인증 방법은 나중에 변경할 수 있습니다.  
[자세한 정보](#)

인증 방법 ○  로컬 및 Azure Active Directory(Azure AD) 인증 모두 사용  
 Azure Active Directory(Azure AD) 인증만 사용

SQL Server 관리자 로그인 \* ○ sqladminuser

SQL 암호 ○ ..... ✓

암호 확인 ○ .....

시스템이 할당한 관리 ID 권한

작업 영역 시스템 ID를 사용하여 작업 영역 네트워크 액세스 권한을 Data Lake Storage Gen 2 계정에 부여하려면 선택합니다.  
[자세한 정보](#)

Data Lake Storage Gen2 계정에 네트워크 액세스를 허용합니다. ○  
선택한 Data Lake Storage Gen2 계정이 네트워크 액세스 규칙을 사용하여 네트워크 액세스를 제한하지 않거나, 기본 탭 아래의 URL를 통해 스토리지 계정을 수동으로 선택했습니다. [자세한 정보](#)



## [Azure Synapse Analytics Hands-on Labs]

SQL Server 관리자 로그인 : sqladminuser

SQL 암호 : 각자 입력

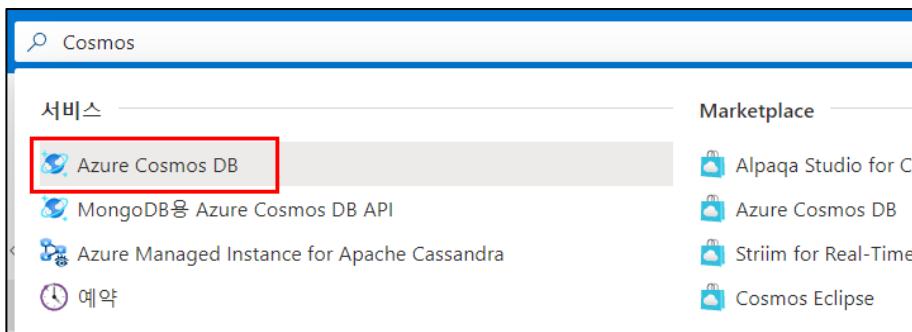
6. 검토 + 만들기를 클릭 후 하단의 만들기로 생성합니다.



\* 빨간 박스 외에 리소스 설정은 기본 값이므로 수정하지 마세요.

### Task 2. Cosmos DB 생성

1. 상단의 둘보기에서 Cosmos DB를 입력하여 선택합니다.



2. Azure Cosmos DB 만들기를 클릭하여 생성할 리소스의 정보를 입력합니다.

The screenshot shows the Azure Cosmos DB management portal. At the top, there's a toolbar with various icons: '+ 만들기' (Create), '복원' (Restore), '보기 관리' (View Management), '새로 고침' (Refresh), and 'CSV로 내보내기' (Export to CSV). Below the toolbar, there are two search/filter boxes: '필드 필터링...' (Field Filtering...) and '구독 == 모두' (Subscription == All) and '리소스 그룹 == 모두' (Resource Group == All). A message below the filters says '0 - 0/0개 레코드를 표시합니다.' (0 - 0/0 records displayed). At the bottom, there's a sorting option '이름 ↑↓' (Name ↑↓).

3. 코어(SQL) API를 선택합니다.

The screenshot shows the 'API 옵션 선택' (Select API Options) page. It lists several database models with their respective creation buttons and detailed information links. The 'Core (SQL) - 권장' (Recommended) section is highlighted with a red box. Other sections include 'Azure Cosmos DB for PostgreSQL', 'MongoDB용 Azure Cosmos DB API', 'Cassandra', 'Azure Table', and 'Gremlin(그래프)'. Each section has a '만들기' (Create) button and a '자세한 정보' (Detailed Information) link.

4. 기본에서 필요한 정보를 입력합니다. 검토 + 만들기를 클릭 후 하단의 만들기로 생성합니다.

## [Azure Synapse Analytics Hands-on Labs]

기본 사항 전역 배포 네트워킹 백업 정책 암호화 태그 검토 + 만들기

Azure Cosmos DB는 확장 가능한 고성능 애플리케이션을 구축하기 위한 완전 관리형 NoSQL 데이터베이스 서비스입니다.

프로젝트 세부 정보

배포된 리소스 및 비용을 관리할 구독을 선택합니다. 풀더와 같은 리소스 그룹을 사용하여 모든 리소스를 구성하세요.

구독 \* Azure SA Team

리소스 그룹 \*

- gildong-rg
- 새로 만들기

인스턴스 세부 정보

계정 이름 \* gildongcosmos

위치 \* (Asia Pacific) Korea Central

용량 모드 ⓘ

- 프로비저닝된 처리량
- Serverless

용량 모드에 대한 자세한 정보

With Azure Cosmos DB free tier, you will get the first 1000 RU/s and 25 GB of storage for free in an account. You can use this tier for testing and prototyping.

무료 계층 할인 적용

- 적용
- 적용 안 함

총 계정 처리량 제한

이 계정에서 프로비전할 수 있는 총 처리량 제한

i 이 제한은 프로비전된 처리량과 관련된 예상치 못한 요금이 발생하지 않을 것을 보증합니다.

**검토 + 만들기** **이전** **다음: 전역 배포**

리소스 그룹 : <username>-rg

리소스 이름 : <username>cosmos (고유 이름 값)

지역 : 한국 중부

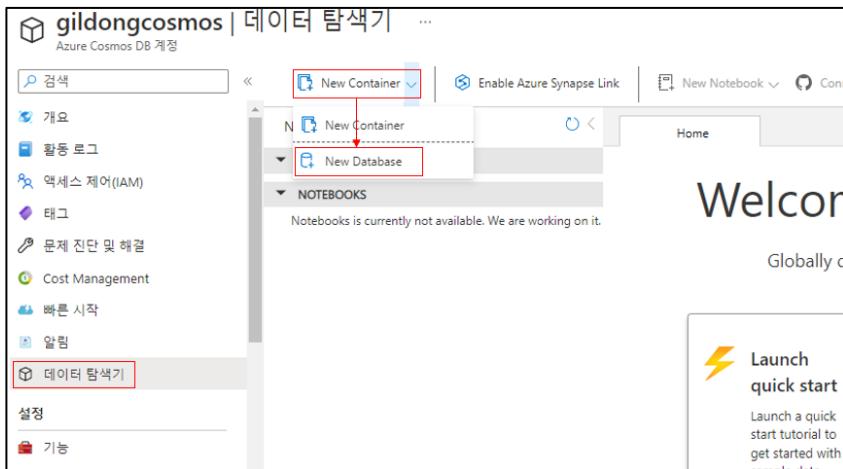
용량 모드 : 프로비저닝된 처리량

무료 계층 할인 적용 : 적용 (RU/s 1000 제한)

총 계정 처리량 제한 : (Default)체크

- 만들어진 Cosmos DB 계정으로 이동합니다. 왼쪽 메뉴에서 데이터 탐색기를 선택합니다. New Container 옆 드롭다운을 눌러 리스트를 확장 후 New Database 를 클릭합니다.

## [Azure Synapse Analytics Hands-on Labs]



6. Database id를 아래와 같이 입력합니다. Autoscale을 설정하고 Database Max RU/s는 기본 값인 4000으로 설정합니다. OK를 눌러 데이터베이스를 생성합니다.  
(Max RU 값이 1000으로 설정되어있다면 1000으로 설정하셔도 됩니다.)

The image shows two side-by-side 'New Database' configuration dialogs. Both dialogs have a warning message about billing if provisioning more than 1000 RU/s. The left dialog has 'Database id' set to 'CosmosDemo', 'Provision throughput' checked, and 'Autoscale' selected. The right dialog has 'Database id' set to 'CosmosDemo', 'Provision throughput' checked, 'Autoscale' selected, and 'Database Max RU/s' set to '1000'. Both dialogs have an 'OK' button at the bottom.

Database id: CosmosDemo

Database Max RU/s : 4000 (OR 1000)

7. 아래와 같이 데이터베이스가 생성된 것을 볼 수 있습니다. 상단의 Enable Azure Synapse Link를 클릭하여 Synapse Link를 허용합니다. (시간이 조금 소요됩니다)

## [Azure Synapse Analytics Hands-on Labs]

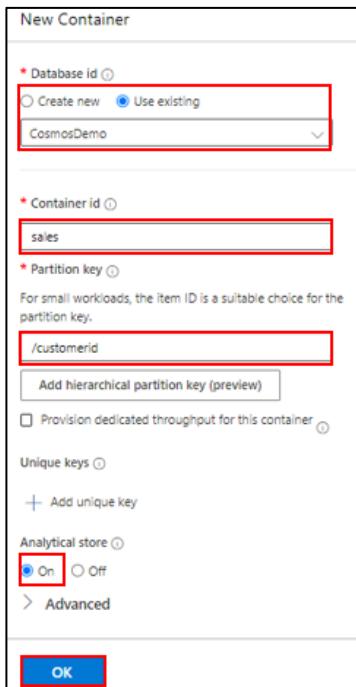
The screenshot shows the Azure Cosmos DB portal for the database 'gildongcosmos'. On the left, there is a sidebar with various navigation links: '개요' (Overview), '활동 로그' (Activity Log), '액세스 제어(IAM)', '태그', '문제 진단 및 해결', '빠른 시작', '알림', and '데이터 탐색기' (Data Explorer). The '데이터 탐색기' link is highlighted. On the right, under the 'SQL API' section, there is a container named 'CosmosDemo'. A red box highlights the 'Enable Azure Synapse Link' button. Below this, a modal dialog box titled 'Enable Azure Synapse Link on your Cosmos DB account' contains the following text: 'Enable Azure Synapse Link to perform near real time analytical analytics on this account, without impacting the performance of your transactional workloads. Azure Synapse Link brings together Cosmos Db Analytical Store and Synapse Analytics'. It includes a 'Learn more' link and two buttons: 'Enable Azure Synapse Link' (highlighted with a red box) and 'Cancel'.

8. CosmosDemo DB의 우측에 있는 메뉴바를 눌러 New Container를 클릭하여 컨테이너를 생성합니다.

The screenshot shows the Azure Cosmos DB portal for the database 'gildongcosmos'. On the left, there is a sidebar with 'NOSQL API' selected. Under the 'DATA' section, there is a database named 'CosmosDemo' (highlighted with a red box). On the right, there is a menu bar with 'New Container' (highlighted with a red box), 'New Notebook', and 'Connect to GitHub'. Below the menu bar, there are buttons for 'Scale', 'New Container' (highlighted with a red box), and 'Delete Database'. A note at the bottom says 'Notebooks is currently not available. We are working on it.' A red box also highlights the 'New Container' button in the menu bar.

9. Database id는 Use existing을 선택하여 기존에 생성한 CosmosDemo를 선택합니다. Container id와 Partition key, Analytical store 옵션을 아래와 같이 생성합니다.

## [Azure Synapse Analytics Hands-on Labs]



Database id : CosmosDemo

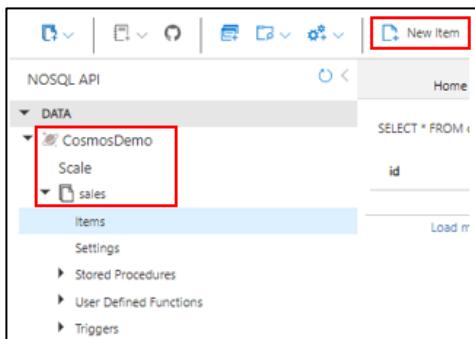
Container id : sales

Partition key : /customerid

Analytical store : On

10. CosmosDemo 데이터베이스 아래 sales 컨테이너가 생성된 것을 확인할 수 있습니다.

Sales 컨테이너의 Items를 선택 후 상단의 New Item을 누릅니다.



11. 에디터에 아래 Json 내용을 붙여 넣고 Save 버튼을 누릅니다.

```
{  
  "id": "S043701",  
  "orderdate": "2019-07-01",  
  "customerid": 123,  
  "customerdetails": {  
    "customername": "Christy Zhu",  
    "customeremail": "christy12@adventure-works.com"  
},
```

```

"product": "Mountain-100 Silver, 44",
"quantity": 1,
"price": 3399.99
}

```

12. 다시 New Item 버튼을 눌러 에디터 안에 아래 두 번째 Json 내용을 추가하고 Save 버튼을 누릅니다.

```

{
  "id": "S043704",
  "orderdate": "2019-07-01",
  "customerid": 124,
  "customerdetails": {
    "customername": "Julio Ruiz",
    "customeremail": "julio1@adventure-works.com"
  },
  "product": "Mountain-100 Black, 48",
  "quantity": 1,
  "price": 3374.99
}

```

13. 마지막으로 New Item 버튼을 눌러 에디터 안에 아래 세 번째 Json 내용을 추가하고 Save 버튼을 누릅니다.

```

{
  "id": "S043707",
  "orderdate": "2019-07-02",
  "customerid": 125,
  "customerdetails": {
    "customername": "Emma Brown",
    "customeremail": "emma3@adventure-works.com" },
  "product": "Road-150 Red, 48",
  "quantity": 1,
  "price": 3578.27
}

```

14. Sales의 Items를 확인하면 12~14에서 추가한 데이터를 볼 수 있습니다.

The screenshot shows the Azure Synapse Analytics Data Explorer interface. On the left, there's a navigation pane with 'DATA' expanded, showing 'CosmosDemo' selected. Under 'CosmosDemo', there are 'Scale', 'sales' (selected), 'Items', 'Settings', 'Stored Procedures', 'User Defined Functions', and 'Triggers'. On the right, a query result is displayed with a red box around the table. The table has columns 'id' and 'value'. The data rows are:

id	value
SO43701	123
SO43704	124
SO43707	125

### Task 3. ADLS Gen2에 컨테이너 및 데이터 생성

- 상단의 돌보기에서 ADLS를 입력하여 앞에서 생성한 <username>adls를 선택합니다.

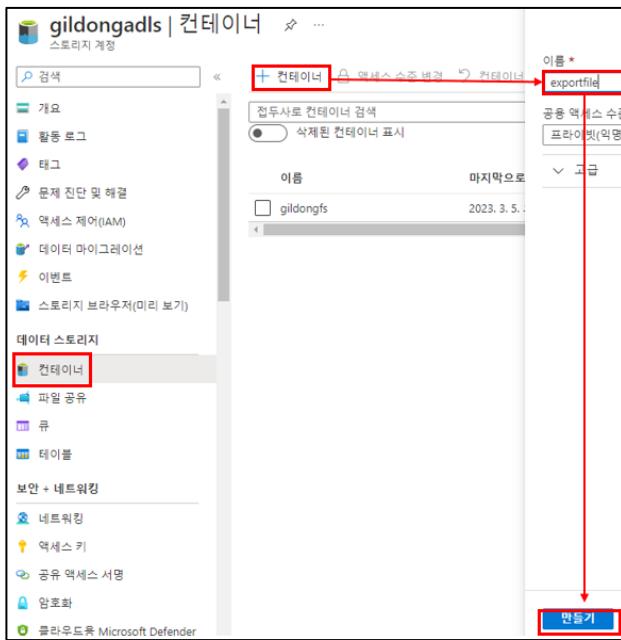
The screenshot shows the Azure portal search results for 'adls'. The search bar at the top contains 'adls'. Below it, there are two sections: '서비스' (Services) and '리소스' (Resources). In the '서비스' section, '스토리지 계정' (Storage Account) is listed. In the '리소스' section, 'gildongadls' is listed. Both items are highlighted with red boxes.

- 왼쪽 하단에서 컨테이너 클릭합니다. 앞에서 생성했던(Synapse Workspace) <username>fs 파일 시스템이 보이는지 확인합니다.

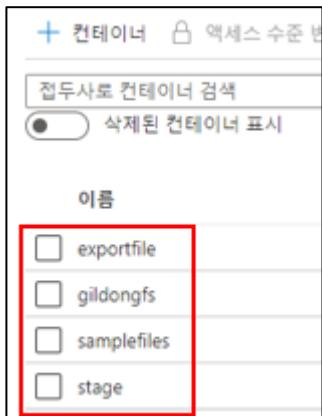
The screenshot shows the Azure Storage Explorer interface for the 'gildongadls' storage account. On the left, there's a sidebar with various options like '검색' (Search), '개요' (Overview), '활동 로그' (Activity Log), '태그' (Tags), etc. On the right, there's a '컨테이너' (Containers) section. A new container named 'gildongfs' is being created, and its name is highlighted with a red box. There's also a button labeled '+ 컨테이너' (Create Container).

- +컨테이너 버튼을 눌러 exportfile을 만듭니다.

## [Azure Synapse Analytics Hands-on Labs]



4. 동일한 과정으로 +컨테이너 버튼을 눌러 stage, samplefiles 도 동일하게 만듭니다.
5. 총 4 개의 컨테이너(exportfile, gildongfs, samplefiles, stage)가 구성되었는지 확인합니다.



6. 위 과정에서 구성한 컨테이너에 파일을 업로드하기 위해 Azure Storage Explorer 를 다운로드하여 설치합니다. 스토리지 브라우저(미리보기)를 선택하여 Azure Storage Explorer 를 다운로드 합니다.

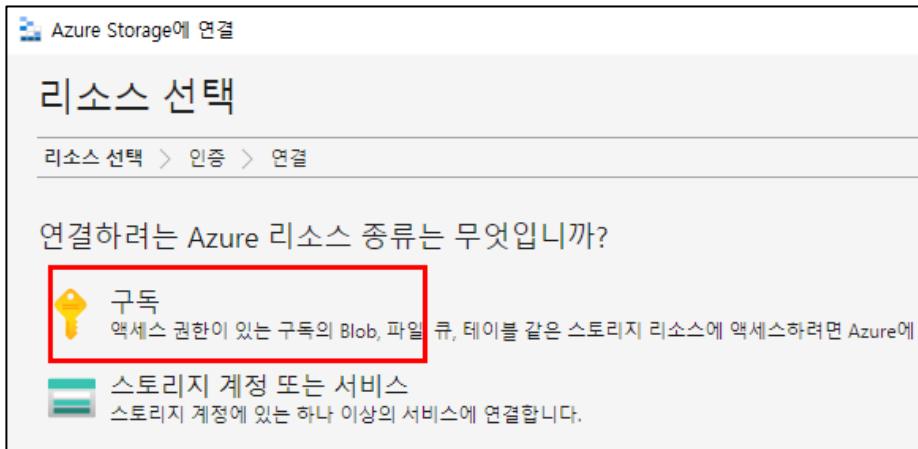
## [Azure Synapse Analytics Hands-on Labs]

The screenshot shows the Azure Storage Explorer landing page. At the top, there's a navigation bar with links like 'Explore', 'Products', 'Solutions', 'Pricing', 'Partners', and 'Resources'. Below the navigation, there's a main heading 'Azure Storage Explorer' and a sub-heading 'Free tool to conveniently manage your Azure cloud storage resources from your desktop.' There are two buttons: 'Try Azure for free' and 'Contact sales for assistance'. A red box highlights the 'Operating system' dropdown menu, which is expanded to show options for Windows, macOS, Linux-snapstore, and Linux-.tar. The 'Windows' option is selected.

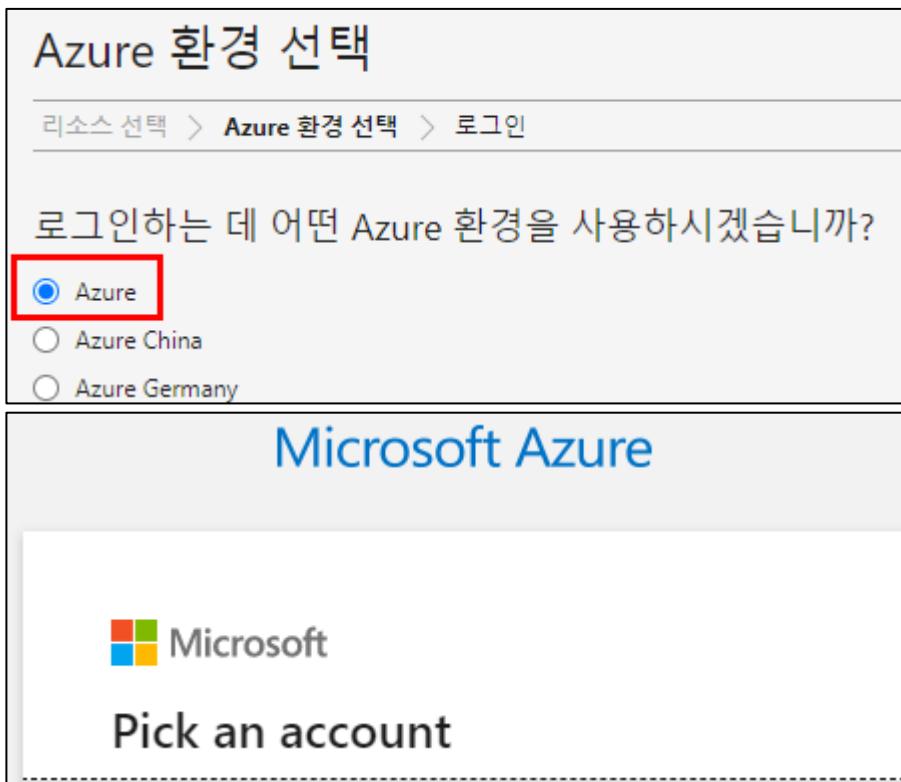
7. 계정 연결을 위해 콘센트 모양의 버튼을 클릭합니다.

The screenshot shows the Microsoft Azure Storage Explorer application interface. The left sidebar has a 'Storage' icon highlighted with a red box. The main pane shows a calendar for September 2021 and a welcome message for version 1.21.2. The message includes information about new features in version 1.21, such as changes to time and date display.

8. 리소스 선택에서 구독을 클릭합니다.



9. Azure 환경 선택에서 Azure 선택 후 다음을 클릭하여 계정을 연결합니다.



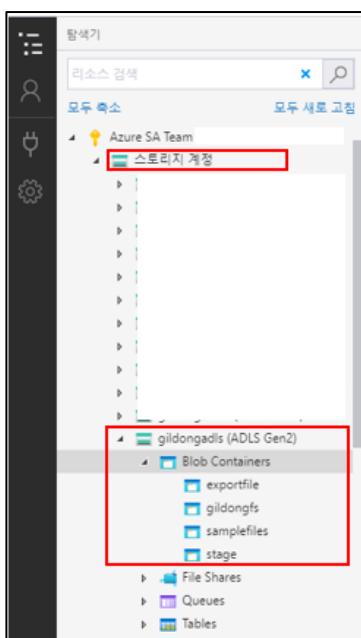
\* 연결이 완료되면 아래와 같은 문구가 나타납니다.

Authenticated. You can return to Storage Explorer. You might need to authenticate again if you close this browser tab.

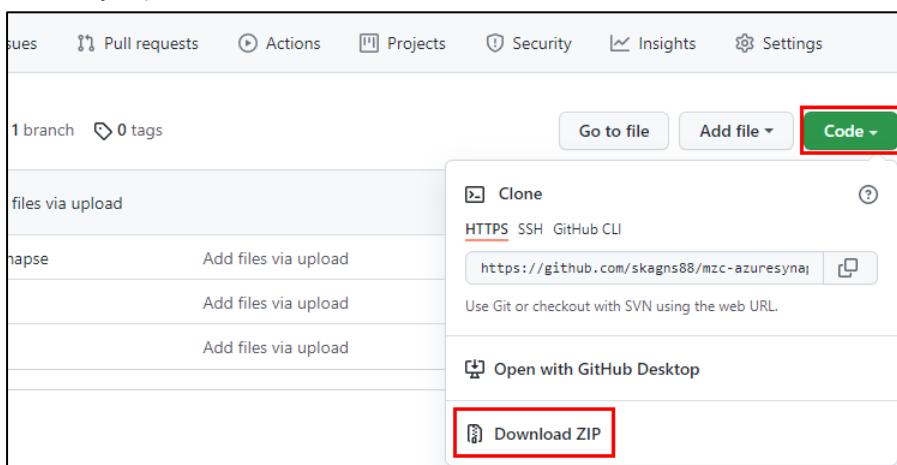
## [Azure Synapse Analytics Hands-on Labs]

10. 맨 위의 왼쪽 상단을 클릭하여 개별 구독 아래에 스토리지 계정에서 앞에서 생성했던 ADLS Gen2 가 보이는지 확인합니다. (리소스 그룹을 생성한 '구독'의 하위 디렉토리에 있습니다.)

11.

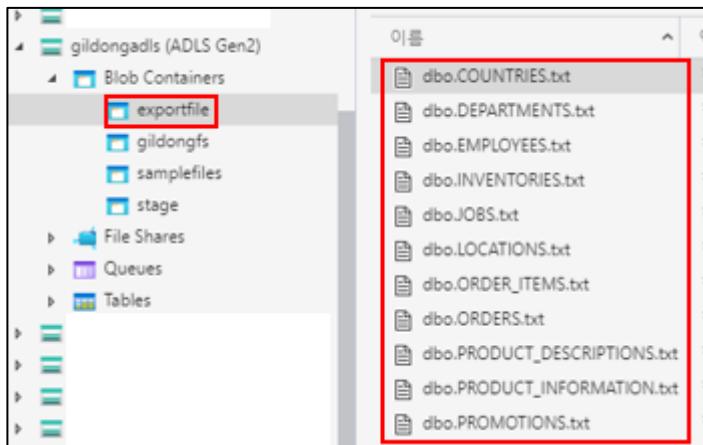


12. [Github](#)에서 실습에 사용할 자료를 다운받습니다. 이미 자료를 받았다면 압축을 풀어서 Azure Synapse 폴더를 확인합니다.

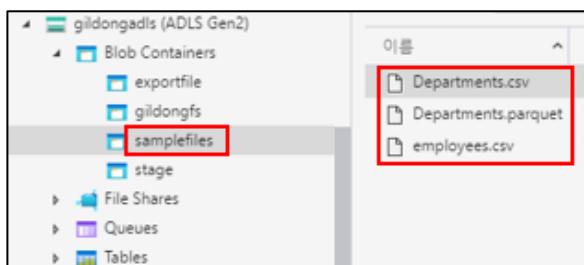


13. 다운로드한 자료에서 Azure Synapse LAB File exportfile 안의 파일들을 exportfile 컨테이너에 Drag & Drop 하여 넣어줍니다. 11 개의 텍스트 파일이 업로드 됩니다.

## [Azure Synapse Analytics Hands-on Labs]



14. Azure Synapse LAB File samplefiles 안의 폴더들을 samplefiles 컨테이너에 Drag & Drop 하여 넣어줍니다. 3 개의 파일이 업로드 됩니다.



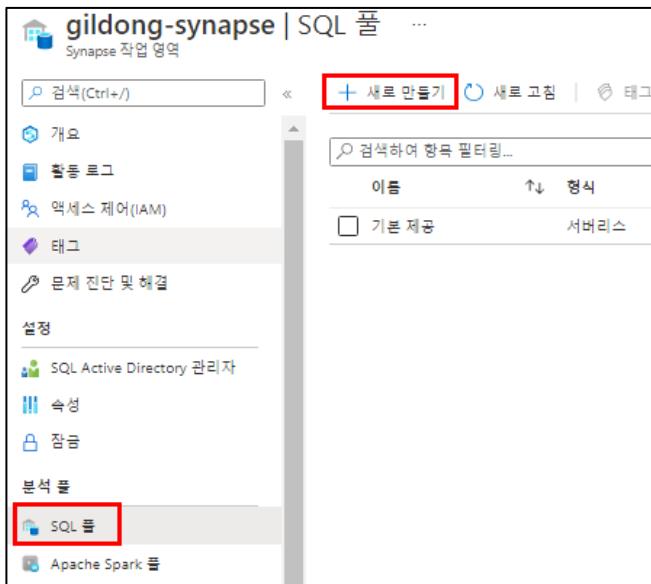
### Task 4. SQLPOOL, SPARK POOL 생성

1. Portal 상단의 돋보기에서 synapse를 입력하여 앞에서 생성한 <username>-synapse를 선택합니다.

이름	형식	리소스 그룹
gildong-synapse	Synapse 작업 영역	mzc-rg

2. 창의 왼쪽에서 SQL 풀로 이동하여 새로 만들기를 클릭합니다.

## [Azure Synapse Analytics Hands-on Labs]



### 3. 기본 사항에서 SQL풀 세부 정보를 입력합니다.

\* 기본 사항 \* 추가 설정 태그 검토 + 만들기

원하는 구성으로 전용 SQL 풀을 만듭니다. [기본] 템플릿을 완료한 후 [검토 + 만들기]로 이동하여 적절한 기본값으로 프로비저닝하거나, 각 템플릿을 방문하여 사용자 지정합니다. 자세한 정보 ↗

전용 SQL 풀 세부 정보

전용 SQL 풀 이름 \*

sqlpool

성능 수준 ⓘ

DW100c

예상 가격 ⓘ

시간당 예상 비용  
--  
가격 책정 세부 정보 보기

전용 SQL 풀 이름 : sqlpool

성능 수준 : DW100c

### 4. 검토 + 만들기를 클릭 후 하단의 만들기를 누릅니다.

\* 기본 사항 \* 추가 설정 태그 **검토 + 만들기**

제품 세부 정보

Azure Synapse Analytics 전용 SQL 풀 Microsoft 제공 사용 약관 | 개인정보처리방침

시간당 예상 비용  
--  
가격 책정 세부 정보 보기

\* 그 외에 리소스 설정은 기본 값이므로 수정하지 마세요.

## [Azure Synapse Analytics Hands-on Labs]

5. 리소스 생성이 완료되면 다시 Azure Synapse Analytics 리소스로 돌아갑니다. SQL 풀을 선택하여 sqlpool 이 생성되었는지 확인합니다.

The screenshot shows the Azure portal interface for the 'gildong-synapse' resource group. On the left, there's a sidebar with options like '개요', '활동 로그', '액세스 제어(IAM)', '태그', '문제 진단 및 해결', '설정' (with 'SQL Active Directory 관리자', '속성', and '잠금'), '분석 풀', and 'SQL 풀'. The 'SQL 풀' option is highlighted with a red box. The main area displays a table with columns '이름', '형식', and '상태'. A row for 'sqlpool' is selected and also highlighted with a red box; it shows '서비스' under '형식' and '온라인' under '상태'.

\* sqlpool 을 클릭하면 전용 SQL 풀에 대한 일시 중지가 가능합니다. 사용하지 않을 시 중지하여 비용 발생을 줄일 수 있습니다.

The screenshot shows the detailed view for the 'sqlpool' resource within the 'gildong-synapse' resource group. The top navigation bar includes '검색(Ctrl+ /)', '새로 만들기', '새로 고침', '태그 지정', and '삭제'. Below the navigation, there are tabs: '일시 중지' (highlighted with a red box), '스케일링', '복원', '+ 새 풀', and '...'. The main content area shows basic information: '리소스 그룹 (변경) : mzc-rg', '상태 : 온라인', and '위치 : 한국 중부'.

6. 창의 왼쪽에서 Apache Spark 풀로 이동하여 새로 만들기를 클릭합니다.

7. 기본 사항에서 Apache Spark 풀 세부 정보를 입력합니다.

\*기본 사항 \*주가 설정 태그 검토 + 만들기

원하는 구성으로 Synapse Analytics Apache Spark 풀을 만듭니다. [기본] 탭을 완료한 후 [검토] + [만들기]로 가서 적절한 기본 값으로 프로비저닝하거나, 각 탭을 방문하여 사용자 지정합니다.

Apache Spark 풀 세부 정보

Apache Spark 풀의 이름을 지정하고 초기 설정을 선택합니다.

Apache Spark 풀 이름 *	sparkpool
노드 크기 패밀리	MemoryOptimized
노드 크기 *	Medium(vCore 8기/64GB)
자동 크기 조정 *	<input checked="" type="radio"/> 사용 <input type="radio"/> 사용 안 함
노드 수 *	3 <input type="range" value="3"/> 10
동적으로 실행기 할당 *	<input type="radio"/> 사용 <input checked="" type="radio"/> 사용 안 함

Apache Spark 풀 이름 : sparkpool

노드 크기 : Medium

자동 크기 조정 : 사용

노드 수 : 3 ~ 10

동적으로 실행기 할당 : 사용 안 함

8. 추가 설정에서 유휴 시간 10, Apache Spark 2.4 버전을 선택합니다.

## [Azure Synapse Analytics Hands-on Labs]

\* 기본 사항 \* 추가 설정 태그 검토 + 만들기

자동 일시 중지 및 구성 요소 버전을 비롯한 추가 구성 매개 변수를 사용자 지정합니다.

자동 일시 중지

자동 일시 중지를 구성합니다. 사용하도록 설정하면 선택한 유휴 시간 후에 Apache Spark 풀이 자동으로 일시 중지됩니다.

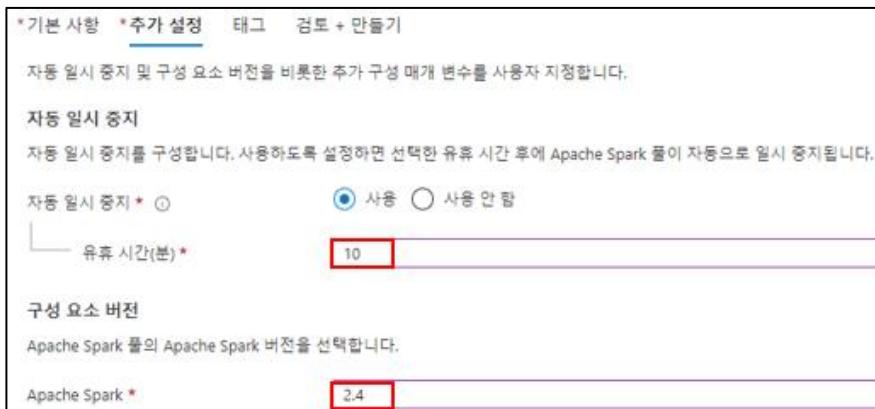
자동 일시 중지 \*  사용  사용 안 함

유휴 시간(분) \* 10

구성 요소 버전

Apache Spark 풀의 Apache Spark 버전을 선택합니다.

Apache Spark \* 2.4



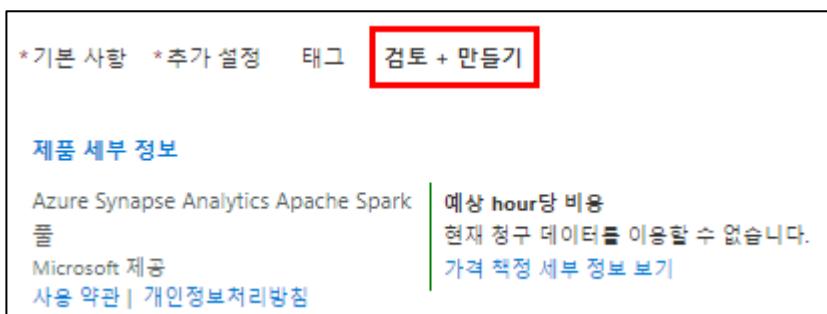
9. 검토 + 만들기를 클릭 후 하단의 만들기로 생성합니다.

\* 기본 사항 \* 추가 설정 태그 **검토 + 만들기**

제품 세부 정보

Azure Synapse Analytics Apache Spark 풀 Microsoft 제공 사용 약관 | 개인정보처리방침

예상 hour당 비용  
현재 청구 데이터를 이용할 수 없습니다.  
[가격 책정 세부 정보 보기](#)



\* 그 외에 리소스 설정은 기본 값이므로 수정하지 마세요.

10. 리소스 생성이 완료되면 다시 Azure Synapse Analytics 리소스로 돌아갑니다. Apache Spark 풀을 선택하여 sparkpool 이 보이는지 확인합니다.

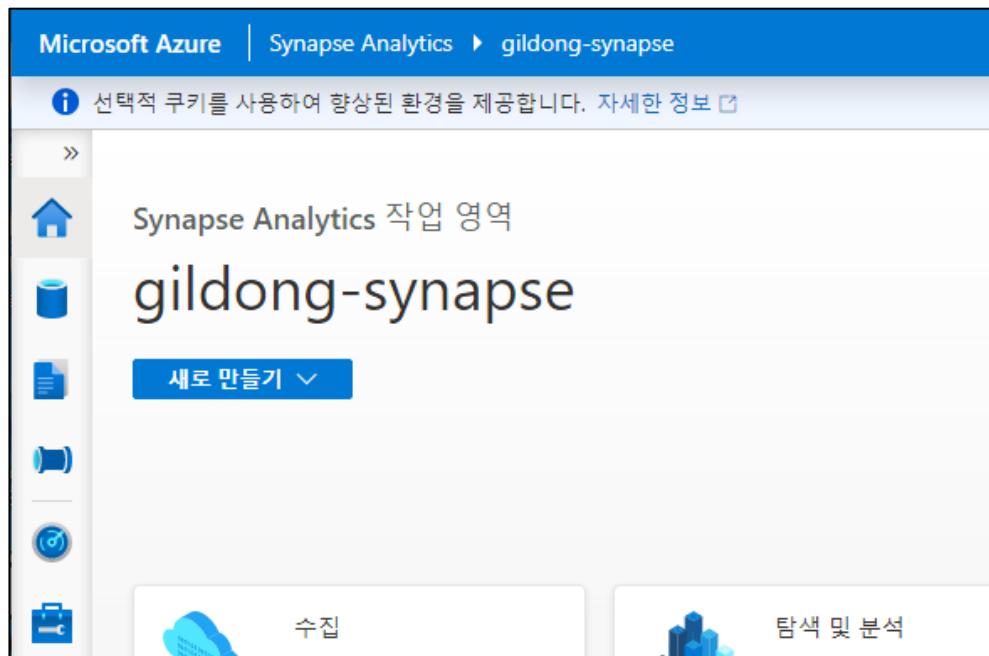
The screenshot shows the Azure Synapse Analytics workspace interface. On the left, there's a sidebar with various options like '개요', '활동 로그', '액세스 제어(IAM)', '태그', '문제 진단 및 해결', '설정' (with 'SQL Active Directory 관리자', '속성', '잠금'), '분석 풀' (with 'SQL 풀' and 'Apache Spark 풀'), and '보안'. The main area is titled 'gildong-synapse | Apache Spark 풀' and shows a list of pools. A search bar at the top has '(Ctrl+)/)' placeholder text. Below it are buttons for '+ 새로 만들기', '새로 고침', '태그 지정', and '삭제'. The pool list table has columns for '이름' (Name) and '노드 제품군' (Node Product Group). One row, 'sparkpool', is selected and highlighted with a red box. At the bottom right of the table are buttons for '메모리 최적화' (Memory Optimization) and '설정' (Configure). The 'Apache Spark 풀' item in the sidebar is also highlighted with a red box.

### Task 5. Azure Synapse Workspace Studio 접속

1. Azure Synapse Analytics 리소스의 개요에서 작업 영역 웹 URL을 클릭하거나 Synapse Studio 이미지의 개시 버튼을 눌러 Studio 화면으로 진입합니다.

The screenshot shows the Azure Synapse Analytics workspace overview page. On the left, there's a sidebar with '작업 영역', '잠금', '분석 풀' (with 'SQL 풀' and 'Apache Spark 풀'), '데이터 탐색기 풀(미리 보기)', '보안' (with '암호화', '네트워킹', 'Identity'), '프라이빗 엔드포인트 연결', '승인된 Azure AD 테넌트', 'Azure SQL 감사', and '클라우드용 Microsoft Defender'. The main area shows basic information about the workspace, including '리소스 그룹 (이동)' : 'gildong-synapse', '상태' : 'Succeeded', '위치' : 'Korea Central', '구독 (이동)' : 'Azure SA Team', '구독 ID' : 'c53bdcf6-8fac-491b-8692-9195df094803', '관리형 가상 네트워크' : '아니요', '관리 ID' : '5d9df0ef-c4b9-4f22-8d6f-64c86df2b085', '작업 영역 웹 URL' : '<https://web.azuresynthesize.net/ko/?workspace=%2bsubscriptions%2fc53bdcf6-8fac-491b-8692-9195df094803>' (this URL is highlighted with a red box), and '태그 (편집)' : '태그를 추가하려면 여기를 클릭'. Below this is a '시작' section with two cards: 'Synapse Studio 열기' (Open Synapse Studio) and '설명서 읽기' (Read the manual). The '개시' button in the 'Synapse Studio 열기' card is highlighted with a red box.

2. Azure Synapse Studio 웹 페이지가 나타납니다.



## Lab 2: External Table을 이용한 테이블 LOAD(Polybase)

### Task 1. 작업 Script 불러오기

- Azure Synapse 작업 영역의 개요에서 작업 영역 웹 URL Link 를 클릭하여 Synapse Studio를 실행합니다.

Synapse Analytics 작업 영역  
gildong-synapse

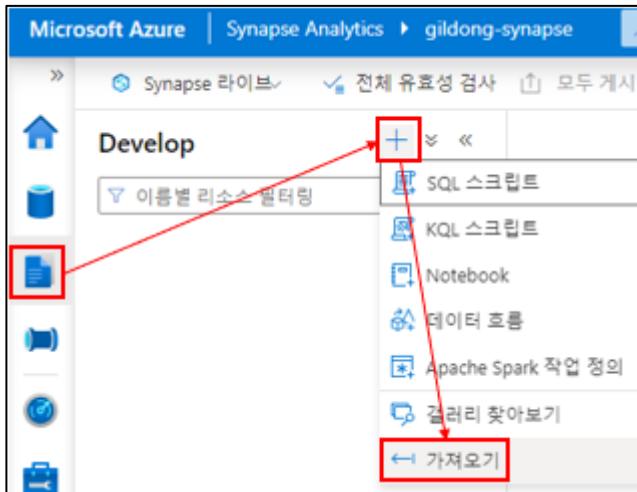
**새로 만들기 ▾**

**수집**  
일회성 또는 예약된 데이터 로드를 수행합니다.

**탐색 및 분석**  
데이터에서 인사이트를 얻는 방법을 알아봅니다.

**시각화**  
Power BI 기능을 사용하여 대화형 보고서를 빌드합니다.

- 개발 Hub 탭을 선택한 후 + 를 불러 가져오기를 선택합니다. 다운로드 받은 Azure Synapse\LabFiles\Scripts\SQL Script 안의 SQL Script 들을 선택하여 불러옵니다. (8 개 SQL 스크립트)



- 8 개의 SQL 스크립트가 정상적으로 Import 되었는지 확인합니다. 모두 게시를 누릅니다.

```

1 SELECT count(*)
2 | FROM [dbo].[CUSTOMERS];

```

- 모두 게시 → 게시를 선택하여 가져온 SQL 스크립트들을 자신의 Studio에 저장하도록 합니다.

보류 중인 모든 변경 내용을 라이브 환경에 게시하려고 합니다.

보류 중인 변경 내용 (8)

이름	변경
00_Create_Data_Loader	(신규)
01_Create_DW_Tables	(신규)
02_Create_Credential	(신규)
03_Create_External_Tables	(신규)
04_Count_Table_Rows	(신규)
05_Load_Data	(신규)
06_Create_Transaction_Table	(신규)
07_OLTP_Table_Count	(신규)

**게시** **취소**

## Task 2. 테이블 생성, 데이터 Load, 데이터 확인

- Manage 탭에서 SQL pools 를 선택 후 Sqlpool의 다시 시작 버튼을 누릅니다. Sqlpool 재기동시 시간이 조금 걸립니다. 재기동이 완료되면 상태에서 온라인으로 표기됩니다.

SQL 풀

서비스 SQL 풀인 기본 제공을 즉시 작업 영역에서 사용할 수 있습니다. 팀 또는 조직 요구 사항과 제약 조건에 맞게 전용 SQL 풀(들)을 구성할 수 있습니다.

이름	형식	상태	크기
기본 제공	서비스	온라인	자동
sqlpool	전용	온라인	DW100c

## [Azure Synapse Analytics Hands-on Labs]

2. Sqlpool 이 기동된 후 Develop 탭으로 이동하여 Data Loader 계정 생성을 위해 00\_CreateDataLoader 스크립트를 실행합니다.
3. 먼저 '연결 대상'에는 sqlpool 을 선택하고, 위의 2 줄은 master 데이터베이스를 선택하여 실행하고 아래 3 줄은 sqlpool 데이터베이스를 선택하여 실행합니다.

```

--Execute this line from the master database
1 CREATE LOGIN dataloader WITH PASSWORD = 'Demo@pass123';
2 CREATE USER dataloader FOR LOGIN dataloader;
3 --Execute the remainder of these lines from the sqlpool database
4
5 CREATE USER dataloader FOR LOGIN dataloader;
6 GRANT CONTROL ON DATABASE::sqlpool TO dataloader;
7 EXEC sp_addrolemember 'largerc', 'dataloader';
8
9

```

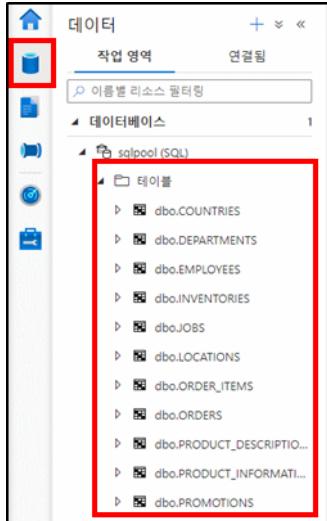
4. Oracle Server VM 의 DW 테이블들의 데이터 이관을 위해 Oracle DW 테이블 스키마에 맞춰 SQL Pool 안에 테이블들을 생성할 것 입니다. 01\_CreateDWTables 스크립트를 수행합니다. (연결 대상 : sqlpool / 데이터베이스 사용 : sqlpool)

```

1 CREATE TABLE dbo.LOCATIONS (LOCATION_ID int not null, STREET_ADDRESS nVARCHAR(100) null, POSTAL_CODE nVARCHAR(100) null, CITY nVARCHAR(100) null, STATE nVARCHAR(100) null, COUNTRY_ID nVARCHAR(100) null, ZIP_CODE nVARCHAR(100) null);
2 CREATE TABLE dbo.JOBS (JOB_ID nVARCHAR(100) not null, JOB_TITLE nVARCHAR(100) not null, MIN_SALARY int null, MAX_SALARY int null);
3 CREATE TABLE dbo.EMPLOYEES (EMPLOYEE_ID int not null, FIRST_NAME nVARCHAR(100), LAST_NAME nVARCHAR(100) null, HIRE_DATE date not null, JOB_ID nVARCHAR(100) not null, MIDDLE_NAME nVARCHAR(100) null, PHONE_NUMBER nVARCHAR(100) null, STAFF_LEVEL int not null, STORE_ID int null);
4 CREATE TABLE dbo.DEPARTMENTS (DEPARTMENT_ID int not null, DEPARTMENT_NAME nVARCHAR(100) null, MANAGER_ID int not null, LOCATION_ID int not null);
5 CREATE TABLE dbo.COUNTRIES (COUNTRY_ID nVARCHAR(10) not null, COUNTRY_NAME nVARCHAR(100) null, REGION_ID int not null);
6 CREATE TABLE dbo.PROMOTIONS (PROMO_ID int not null, PROMO_NAME nVARCHAR(100)) WITH(CLUSTERED COLUMNSTORE INDEX);
7 CREATE TABLE dbo.PRODUCT_DESCRIPTIONS (PRODUCT_ID int NOT NULL, LANGUAGE_ID nVARCHAR(3) NOT NULL, TRANSLATED bit not null);
8 CREATE TABLE dbo.PRODUCT_INFORMATION (PRODUCT_ID INT NOT NULL, PRODUCT_NAME nVARCHAR(50), CATEGORY_ID INT, WEIGHT decimal(5,2));
9 CREATE TABLE dbo.ORDERS (ORDER_ID INT not null, ORDER_DATE datetime not null, ORDER_MODE nVARCHAR(8), CUSTOMER_ID int not null, EMPLOYEE_ID int not null, SHIPMENT_ID int not null);
10 CREATE TABLE dbo.ORDER_ITEMS (ORDER_ID int not null, LINE_ITEM_ID int not null, PRODUCT_ID int not null, UNIT_PRICE decimal(10,2));
11 CREATE TABLE dbo.INVENTORIES (PRODUCT_ID int not null, WAREHOUSE_ID int not null, QUANTITY_ON_HAND int not null);

```

5. SQL Pool 안에 테이블들이 정상적으로 생성되었는지 확인하기 위해 데이터 Hub 탭을 선택하고 sqlpool 안의 테이블 List 를 확인합니다.



6. 02\_CreateCredential 스크립트를 수행 절차.

(주의 : 해당 스크립트를 바로 실행하지 마세요.)

- Sqlpool에서 exportfile 컨테이너에 접근하기 위한 Master Key, Credential, Data Source, File Format을 정의하는 과정입니다.
- Oracle DW 테이블에서 Export 받은 파일들을 ADLS Gen2의 'exportfile' 컨테이너 안에 옮겨 놓았습니다.
- 개발 Hub로 돌아와서 02\_CreateCredential 스크립트를 선택합니다.
- 02\_CreateCredential 스크립트 내용 중 <ADLS Gen2 Account Name>, <ADLS Gen2 Key>를 자신의 환경에 맞게 내용을 바꾼 후 스크립트를 실행해야 합니다.

```

1 CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'Demo@pass123';
2 CREATE DATABASE SCOPED CREDENTIAL MigrationCredential
3 WITH IDENTITY = <ADLS Gen2 Account Name>, SECRET = <ADLS Gen2 Key>
4
5 CREATE EXTERNAL DATA SOURCE MigrationStor WITH (TYPE = HADOOP,
6 LOCATION=
7 'wasbs://exportfile<ADLS Gen2 Account Name>.blob.core.windows.net',
8 CREDENTIAL = MigrationCredential);
9
10 CREATE EXTERNAL FILE FORMAT MigrationFiles WITH(FORMAT_TYPE = DelimitedText,
11 FORMAT_OPTIONS (FIELD_TERMINATOR = '|'));

```

- <ADLS Gen2 Account Name>, <ADLS Gen2 Key>를 확인하기 위하여 Azure Portal의 Storage Account를 선택 후 액세스 키 탭을 선택합니다. 스토리지 계정 이름을 확인 후 Key1의 표시 버튼을 눌러 키 값을 확인합니다.

## [Azure Synapse Analytics Hands-on Labs]

The screenshot shows the Azure Key Vault interface for the 'gildongadls' storage account. The 'Access Keys' section is highlighted. The 'key1' key is selected, and its value is shown in the 'Value' field. The 'Copy' button next to the value is highlighted with a red box.

- 위에서 확인한 스토리지 계정 이름과 Key1의 키 값을 복사하여 02\_CreateCredential 스크립트의 <ADLS Gen2 Account Name>, <ADLS Gen2 Key> 값을 수정하여 스크립트를 실행합니다.

```

CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'Demo@pass123';
CREATE DATABASE SCOPED CREDENTIAL MigrationCredential
WITH IDENTITY = 'gildongadls', SECRET = 's0VVd/c8yTzVM8X0i3G0sx2A+bND5ifz5NbCHKHQYfih76Ja6TTyHKRdQMnOPGk5...';
CREATE EXTERNAL DATA SOURCE MigrationStor WITH (TYPE = HADOOP,
LOCATION= 'wasbs://exportfile@gildongadls.blob.core.windows.net',
CREDENTIAL = MigrationCredential);
CREATE EXTERNAL FILE FORMAT MigrationFiles WITH(FORMAT_TYPE = DelimitedText,
FORMAT_OPTIONS (FIELD_TERMINATOR = '|'));

```

- 다음으로, 03\_CreateExternalTables 스크립트를 실행하여 Export 받은 파일들을 Read 할 External 테이블들을 생성합니다.

## [Azure Synapse Analytics Hands-on Labs]

```

1 CREATE external TABLE dbo.LOCATIONS_External (LOCATION_ID int not null, STREET_ADDRESS nVARCHAR(100) null, P
2 CREATE external TABLE dbo.JOBS_External (JOB_ID nVARCHAR(100) not null, JOB_TITLE nVARCHAR(100) not null, MIN
3 CREATE external TABLE dbo.EMPLOYEES_External (EMPLOYEE_ID int not null, FIRST_NAME nVARCHAR(100), LAST_NAME n
4 CREATE external TABLE dbo.DEPARTMENTS_External (DEPARTMENT_ID int not null, DEPARTMENT_NAME nVARCHAR(100) nul
5 CREATE external TABLE dbo.COUNTRIES_External (COUNTRY_ID nvarchar(10) not null, COUNTRY_NAME nVARCHAR(100)) WITH(LOCATION
6 CREATE external TABLE dbo.PROMOTIONS_External (PROMO_ID int not null, PROMO_NAME nVARCHAR(100)) WITH(LOCATION
7 CREATE external TABLE dbo.PRODUCT_DESCRIPTIONS_External (PRODUCT_ID INT NOT NULL, LANGUAGE_ID nVARCHAR(50),
8 CREATE external TABLE dbo.PRODUCT_INFORMATION_External (PRODUCT_ID INT NOT NULL, PRODUCT_NAME nVARCHAR(50),
9 CREATE external TABLE dbo.ORDERS_External (ORDER_ID INT not null, ORDER_DATE datetime not null, ORDER_MODE nV
10 CREATE external TABLE dbo.ORDER_ITEMS_External (ORDER_ID int not null, LINE_ITEM_ID int not null, PRODUCT_ID
11 CREATE external TABLE dbo.INVENTORIES_External (PRODUCT_ID int not null, WAREHOUSE_ID int not null, QUANTITY_
12

```

8. **04\_Count\_Table\_Rows** 스크립트를 실행하여 sqlpool 의 테이블과 External 테이블의 건수를 비교 합니다.

```

1 select 'LOCATIONS' Table_Name, (SELECT COUNT(*) FROM dbo.locations) ASIS_CNT, 'LOCATIONS_EXTERNAL' Table_Name
2 select 'JOBS' Table_Name, (SELECT COUNT(*) FROM dbo.JOBs) ASIS_CNT, 'JOBS_EXTERNAL' Table_Name, (SELECT COUNT(*)
3 select 'EMPLOYEES' Table_Name, (SELECT COUNT(*) FROM dbo.EMPLOYEES) ASIS_CNT, 'EMPLOYEES_EXTERNAL' Table_Name
4 select 'DEPARTMENTS' Table_Name, (SELECT COUNT(*) FROM dbo.DEPARTMENTS) ASIS_CNT, 'DEPARTMENTS_EXTERNAL' Tabl
5 select 'COUNTRIES' Table_Name, (SELECT COUNT(*) FROM dbo.COUNTRIES) ASIS_CNT, 'COUNTRIES_EXTERNAL' Table_Name
6 select 'PROMOTIONS' Table_Name, (SELECT COUNT(*) FROM dbo.PROMOTIONS) ASIS_CNT, 'PROMOTIONS_EXTERNAL' Table_N
7 select 'PRODUCT_DESCRIPTIONS' Table_Name, (SELECT COUNT(*) FROM dbo.PRODUCT_DESCRIPTIONS) ASIS_CNT, 'PRODUCT_
8 select 'PRODUCT_INFORMATION' Table_Name, (SELECT COUNT(*) FROM dbo.PRODUCT_INFORMATION) ASIS_CNT, 'PRODUCT_IN
9 select 'ORDERS' Table_Name, (SELECT COUNT(*) FROM dbo.ORDERS) ASIS_CNT, 'ORDERS_EXTERNAL' Table_Name, (SELECT
10 select 'ORDER_ITEMS' Table_Name, (SELECT COUNT(*) FROM dbo.ORDER_ITEMS) ASIS_CNT, 'ORDER_ITEMS_EXTERNAL' T
11 select 'INVENTORIES' Table_Name, (SELECT COUNT(*) FROM dbo.INVENTORIES) ASIS_CNT, 'INVENTORIES_EXTERNAL' Tab
12

```

- 04\_Count\_Table\_Rows 스크립트 실행 결과입니다.
- SQL Pool 에 생성된 테이블은 아직 Data 가 Load 되지 않아 0 건으로 나오는 반면 External 테이블은 ADLS Gen2 의 파일(외부 데이터)에서 바로 Data 를 Load 하기 때문에 0 건 이상의 row 가 나옵니다.

Table_Name	ASIS_CNT	Table_Name	TOBE_CNT
LOCATIONS	0	LOCATIONS_EXTERNAL	23
JOBS	0	JOBS_EXTERNAL	19
EMPLOYEES	0	EMPLOYEES_EXTERNAL	107
DEPARTMENTS	0	DEPARTMENTS_EXTERNAL	27
COUNTRIES	0	COUNTRIES_EXTERNAL	25
PROMOTIONS	0	PROMOTIONS_EXTERNAL	2
PRODUCT_DESCRIPTIONS	0	PRODUCT_DESCRIPTIONS_EXTERNAL	8640
PRODUCT_INFORMATION	0	PRODUCT_INFORMATION_EXTERNAL	288
ORDERS	0	ORDERS_EXTERNAL	105
ORDER_ITEMS	0	ORDER_ITEMS_EXTERNAL	665
INVENTORIES	0	INVENTORIES_EXTERNAL	1112

9. **05\_LoadData** 스크립트를 실행하여 External 테이블의 Data 를 테이블로 Insert 합니다.

## [Azure Synapse Analytics Hands-on Labs]

```

1 insert into dbo.locations select LOCATION_ID,STREET_ADDRESS,POSTAL_CODE,CITY,STATE_PROVINCE,COUNTRY_ID from
2 insert into dbo.jobs select JOB_ID,JOB_TITLE,MIN_SALARY,MAX_SALARY from dbo.JOBs_External;
3 insert into dbo.employees select EMPLOYEE_ID,FIRST_NAME,LAST_NAME,EMAIL,PHONE_NUMBER,HIRE_DATE,JOB_ID,SALARY
4 insert into dbo.departments select DEPARTMENT_ID,DEPARTMENT_NAME,MANAGER_ID,LOCATION_ID from dbo.DEPARTMENTS
5 insert into dbo.countries select COUNTRY_ID,COUNTRY_NAME,REGION_ID from dbo.COUNTRIES_External;
6 insert into dbo.PROMOTIONS select PROMO_ID,PROMO_NAME from dbo.PROMOTIONS_External;
7 insert into dbo.PRODUCT_DESCRIPTIONS select PRODUCT_ID,LANGUAGE_ID,TRANSLATED_NAME,TRANSLATED_DESCRIPTION fr
8 insert into dbo.PRODUCT_INFORMATION select PRODUCT_ID,PRODUCT_NAME,CATEGORY_ID,WEIGHT_CLASS,WARRANTY_PERIOD,
9 insert into dbo.ORDERS select ORDER_ID,ORDER_DATE,ORDER_MODE,CUSTOMER_ID,ORDER_STATUS,ORDER_TOTAL,SALES_REP
10 insert into dbo.ORDER_ITEMS select ORDER_ID,LINE_ITEM_ID,PRODUCT_ID,UNIT_PRICE,QUANTITY from dbo.ORDER_ITEMS
11 insert into dbo.INVENTORIES select PRODUCT_ID,WAREHOUSE_ID,QUANTITY_ON_HAND from dbo.INVENTORIES_External;
12
13

```

10. 다시 **04\_Count\_Table\_Rows** 스크립트 수행하여 각 테이블의 Rows 를 비교합니다.  
 05\_Load\_Data 가 정상적으로 수행되었다면 각 테이블의 ASIS\_CNT 와 TOBE\_CNT 의 ROW 건수가 동일할 것 입니다.

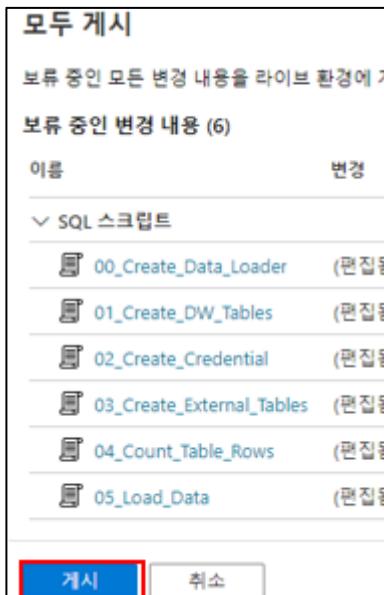
Table_Name	ASIS_CNT	Table_Name	TOBE_CNT
LOCATIONS	23	LOCATIONS_EXTERNAL	23
JOBs	19	JOBs_EXTERNAL	19
EMPLOYEES	107	EMPLOYEES_EXTERNAL	107
DEPARTMENTS	27	DEPARTMENTS_EXTERNAL	27
COUNTRIES	25	COUNTRIES_EXTERNAL	25
PROMOTIONS	2	PROMOTIONS_EXTERNAL	2
PRODUCT_DESCRIPTIONS	8640	PRODUCT_DESCRIPTIONS_EXTER...	8640
PRODUCT_INFORMATION	288	PRODUCT_INFORMATION_EXTER...	288
ORDERS	105	ORDERS_EXTERNAL	105
ORDER_ITEMS	665	ORDER_ITEMS_EXTERNAL	665
INVENTORIES	1112	INVENTORIES_EXTERNAL	1112

11. 지금까지 Lab 2 를 수행하면서 변경된 내용을 반영하기 위해 모두 계시 버튼을 눌러 변경된 내용을 반영 합니다.

```

1 insert in
2 insert in
3 insert in
4 insert in
5 insert in
6 insert in
7 insert in
8 insert in
9 insert in
10 insert in
11 insert in

```



## Lab 3: Pipeline을 이용한 원격 DB 데이터 LOAD(ADF방식)

### Task 1. Source 연결 설정

- 개발 Hub 탭에서 06\_CreateTransactionTable 스크립트를 실행하여 테이블을 생성합니다.

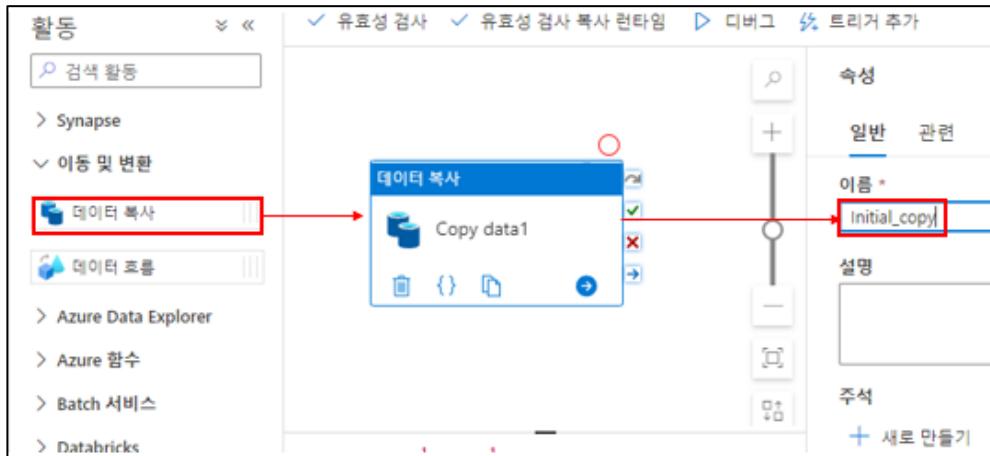
```

CREATE TABLE DBO.CUSTOMERS
(
    CUST_ID INT NOT NULL,
    CUST_FIRST_NAME NVARCHAR(20) NOT NULL,
    CUST_LAST_NAME NVARCHAR(40) NOT NULL,
    CUST_GENDER CHAR(1) NOT NULL,
    CUST_YEAR_OF_BIRTH INT NOT NULL,
    CUST_MARITAL_STATUS NVARCHAR(20),
    CUST_STREET_ADDRESS NVARCHAR(40) NOT NULL,
    CUST_POSTAL_CODE NVARCHAR(10) NOT NULL,
    CUST_CITY NVARCHAR(30) NOT NULL,
    CUST_CITY_ID INT NOT NULL,
    CUST_STATE_PROVINCE NVARCHAR(40) NOT NULL,
    CUST_STATE_PROVINCE_ID INT NOT NULL,
    COUNTRY_ID INT NOT NULL,
)

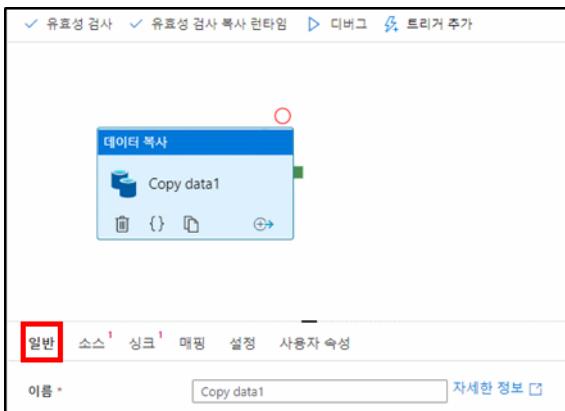
```

- 통합(Integrate) Hub 탭으로 이동하여 + 를 누른 후 파이프라인을 선택합니다.

3. 이동 및 변환 탭에서 데이터 복사 항목을 캔버스 중앙으로 Drag & Drop 합니다. Activity의 이름을 Initial\_copy로 입력합니다.

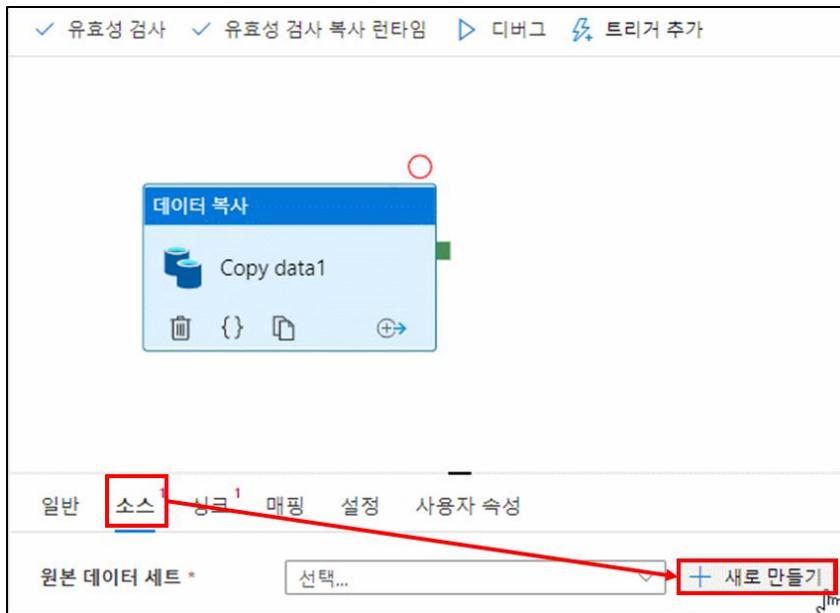


4. 데이터 복사를 선택 후 하단의 일반 탭은 변경 내용 없이 소스 탭으로 이동합니다.

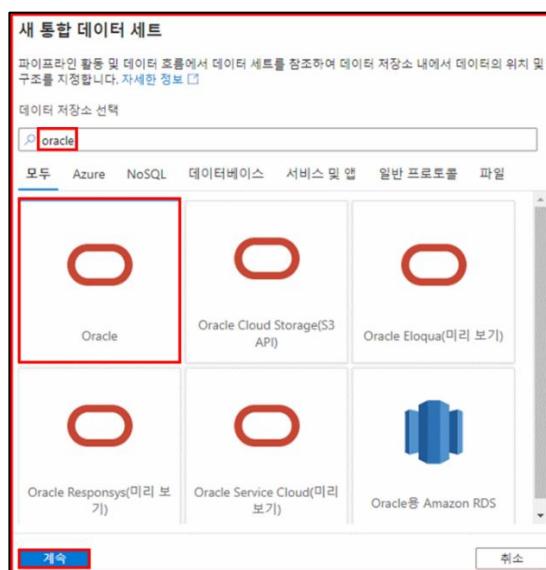


5. 원본 테이블(Oracle Server VM)에 접근하기 위한 데이터 세트를 만드는 과정입니다. 소스 탭의 원본 데이터 세트 항목에서 새로 만들기를 선택합니다.

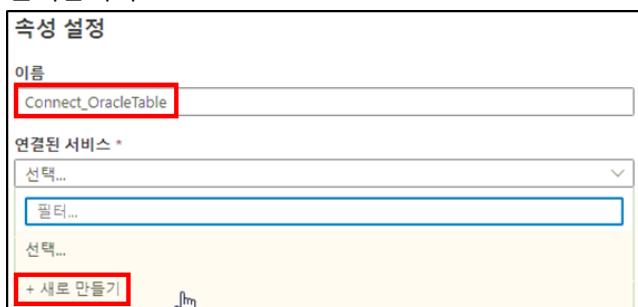
## [Azure Synapse Analytics Hands-on Labs]



6. Oracle 을 선택 후 계속 을 누릅니다.



7. 속성의 이름을 Connect\_OracleTable 로 입력하고 연결된 서비스에서 새로 만들기를 선택합니다.



8. 새 연결된 서비스에서 각 항목을 아래와 같이 기입 후 **연결 테스트**를 하여 **연결 성공**이 된 것을 확인 후 **만들기**를 누릅니다. (오라클 IP 고정하여 셋팅)

항목	값
이름	Oracle_server
호스트	20.214.139.183
포트	1521
연결 형식	Oracle SID
SID	orcl
사용자 이름	scott
암호	oracle

**새 연결된 서비스**

Oracle 자세한 정보

이름 \*

Oracle\_server

설명

통합 런타임을 통해 연결 \*

AutoResolveIntegrationRuntime

연결 문자열 Azure Key Vault

호스트 \*

20.214.139.183

포트 \*

1521

연결 형식

Oracle SID

SID \*

orcl

사용자 이름 \*

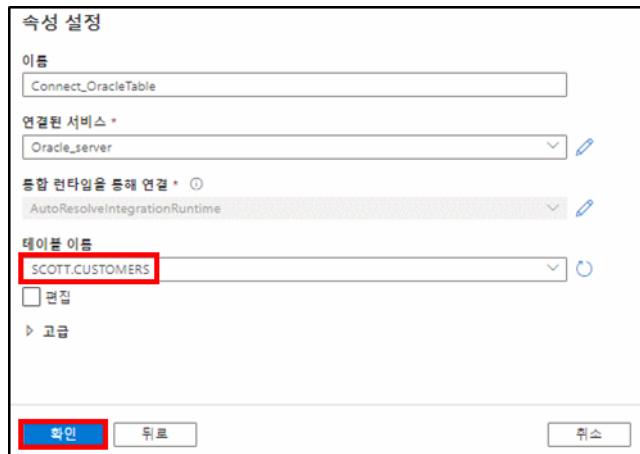
scott

암호 \*

연결 성공

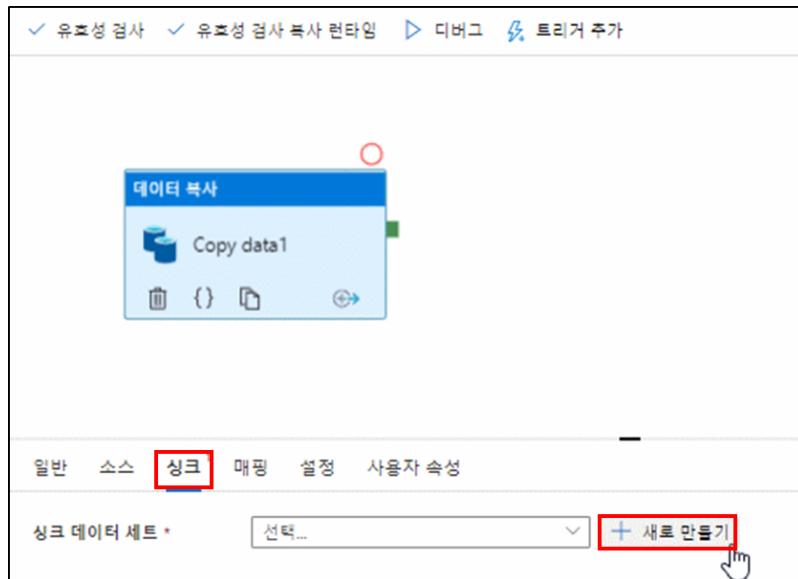
만들기 취소 연결 테스트

- 다시 속성 설정으로 돌아와서 테이블 이름을 SCOTT.CUSTOMERS 를 선택하거나 입력 후 확인을 누릅니다. 이것으로 소스 테이블(Oracle Server VM 에 있는 Transaction Table)에 접속하기 위한 설정을 마쳤습니다.



## Task 2. Target 연결 설정

- 이제 소스 테이블을 LOAD 하기 위한 Synapse dedicated SQLPOOL 내 SYNC 테이블 데이터 셋을 설정해야 합니다. 싱크 탭의 싱크 데이터 세트 항목에서 새로 만들기를 선택합니다.

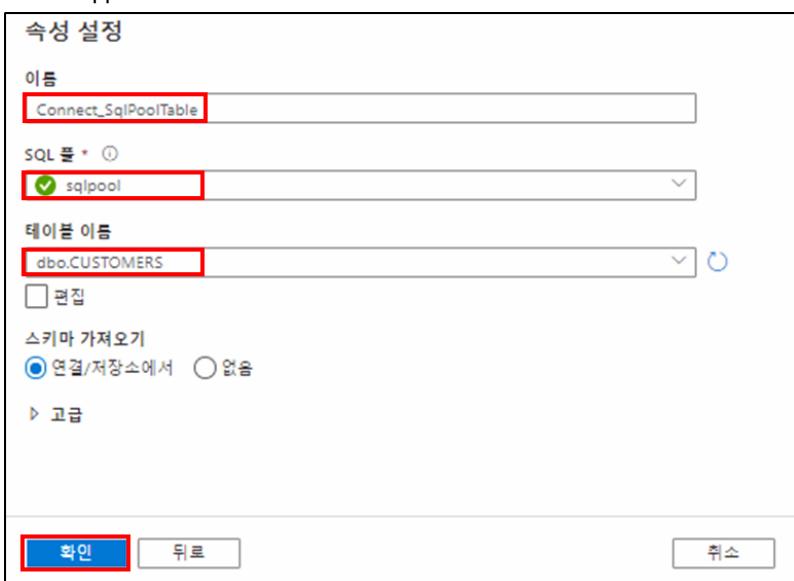


- 새 통합 데이터 세트에서 데이터 저장소 선택을 Azure Synapse 전용 SQL 풀을 선택하고 계속을 누릅니다.

## [Azure Synapse Analytics Hands-on Labs]

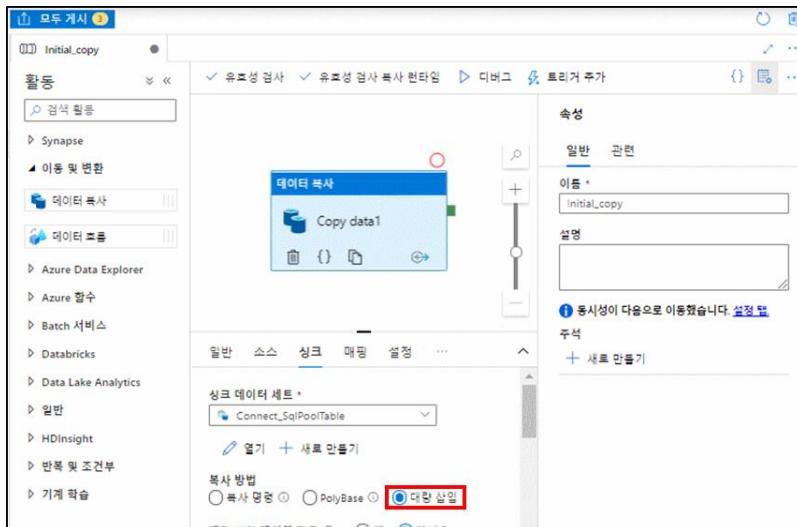


- 속성 설정에서 속성 이름을 Connect\_SqlPoolTable 로 입력하고 SQL 풀은 미리 생성해 놓은 sqlpool 을 선택하고 테이블 이름은 dbo.CUSTOMERS 를 선택합니다.

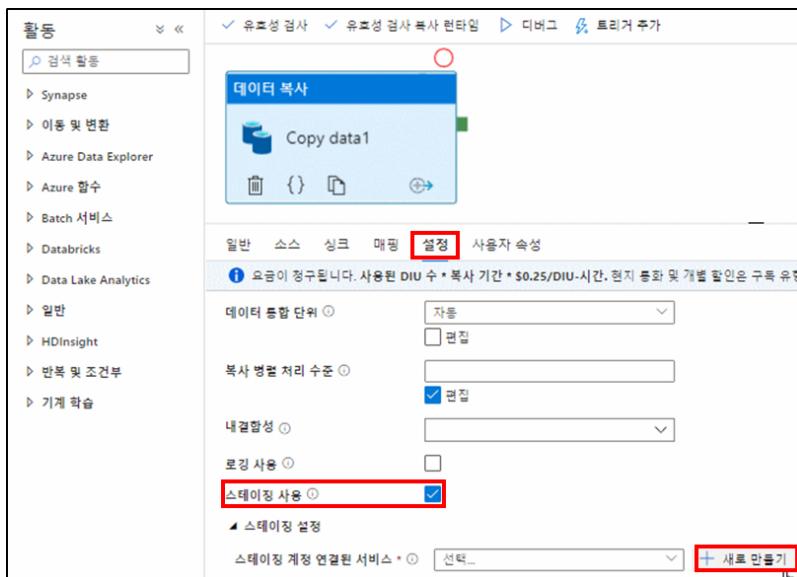


- 싱크 데이터 세트를 생성했습니다. 추가로 싱크 탭에서 복사 방법을 대량 삽입을 선택하고 설정 탭으로 이동합니다.

## [Azure Synapse Analytics Hands-on Labs]



5. 설정 탭에서 스테이징 사용을 Check 하고 새로 만들기를 누릅니다.

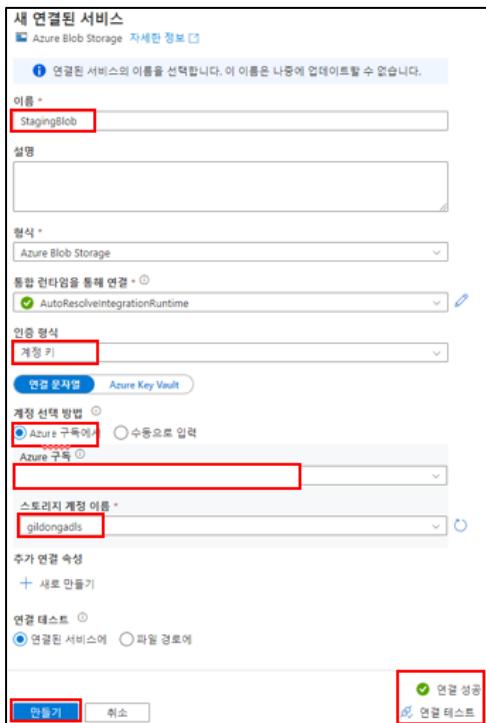


6. 새 연결된 서비스의 의 각 항목을 아래와 같이 기입, 선택합니다.

항목	값
이름	StagingBlob
형식	Azure Blob Storage
Azure 구독	개인 구독 선택
스토리지 계정 이름	개인별 생성한 Storage Account 선택

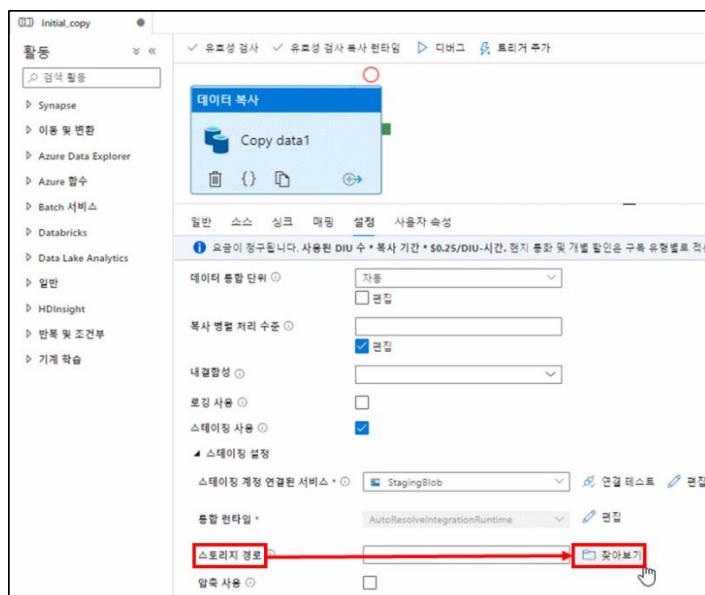
각 항목을 적절히 입력, 선택 후 연결테스트를 수행하여 연결 성공이 된 것을 확인하고 만들기를 누릅니다.

## [Azure Synapse Analytics Hands-on Labs]

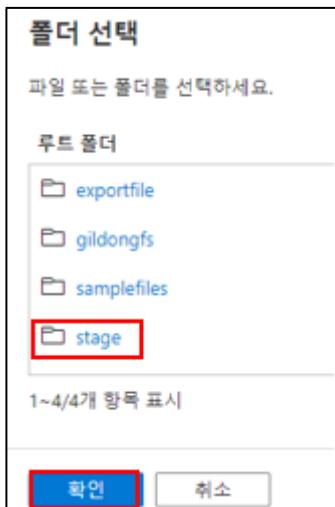


7. 마지막으로 작업이 수행될 스테이징에 사용할 스토리지 경로를 지정합니다.

스토리지 경로 항목에서 찾아보기를 누릅니다.



8. 폴더 선택에서 stage 를 선택 후 확인을 누릅니다.



- 설정 탭에서 모든 설정을 마쳤습니다. 새로 생성된 항목들(파이프 라인, 데이터 세트)을 모두 게시하여 반영합니다.

모두 게시 4

활동

▶ Synapse

◀ 이동 및 변환

▶ 데이터 복사

▶ 데이터 흐름

▶ Azure Data Explorer

데이터 복사

Copy data1

보류 중인 모든 변경 내용을 라이브 환경에 게시합니다.

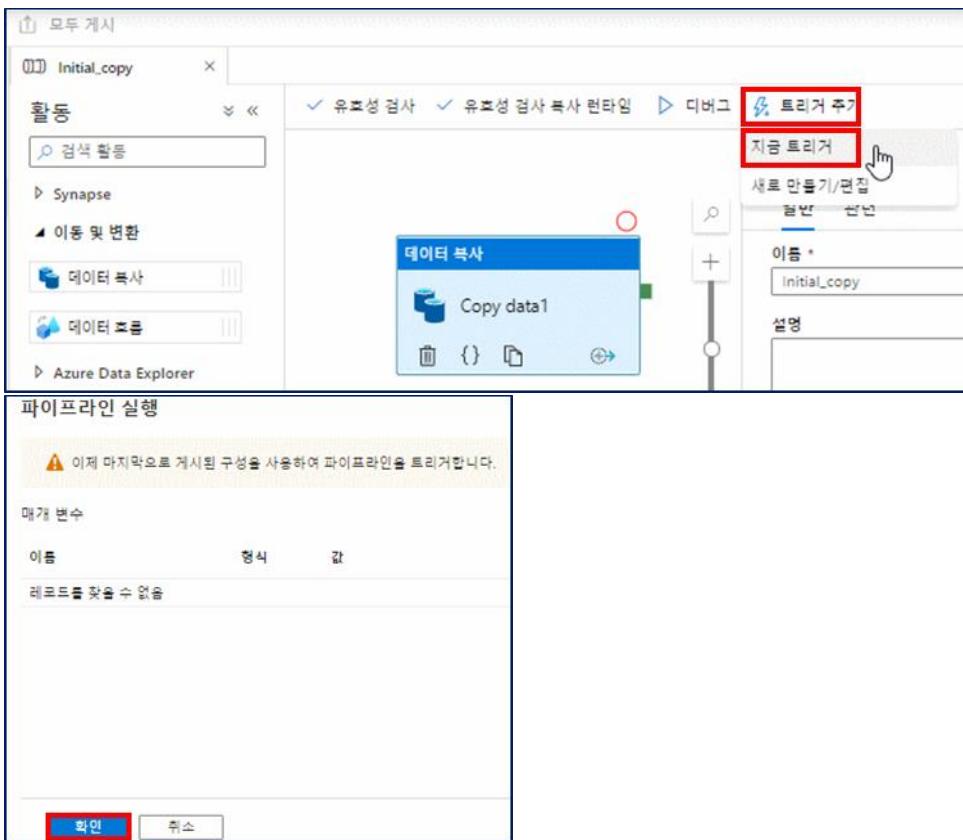
보류 중인 변경 내용 (4)

이름	변경
파이프라인	
Initial_copy	(신규)
데이터 세트	
Connect_OracleTable	(신규)
Connect_SqlPoolTable	(신규)
SQL 스크립트	
06_Create_Transaction_Table	(편집됨)

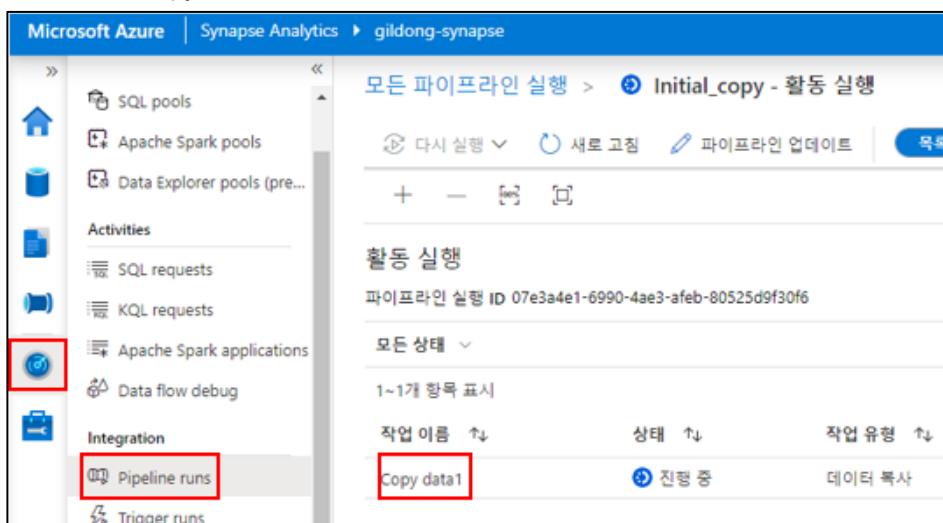
게시 취소

## [Azure Synapse Analytics Hands-on Labs]

10. 게시가 완료되면, 트리거를 실행하여 파이프 라인을 실행합니다.

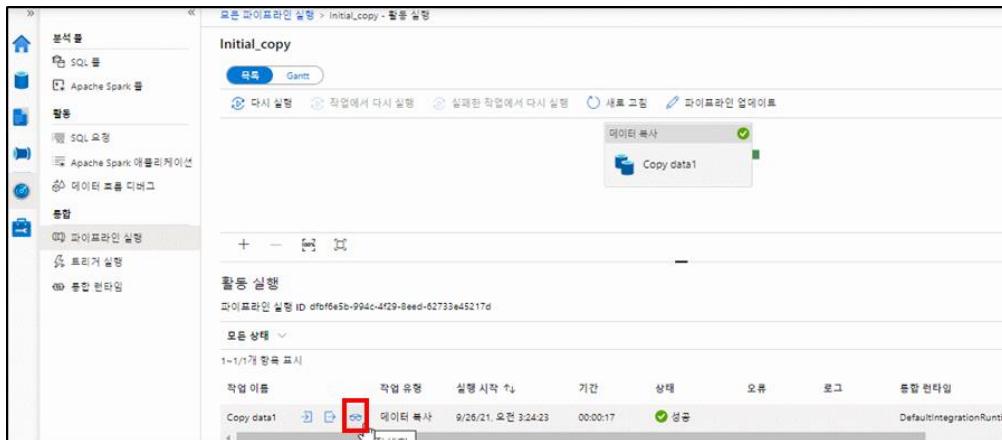


11. 실행중인 파이프 라인을 모니터합니다. 모니터 Hub 탭에서 파이프라인 실행 항목을 선택한 후 Copy\_data1 을 선택합니다.

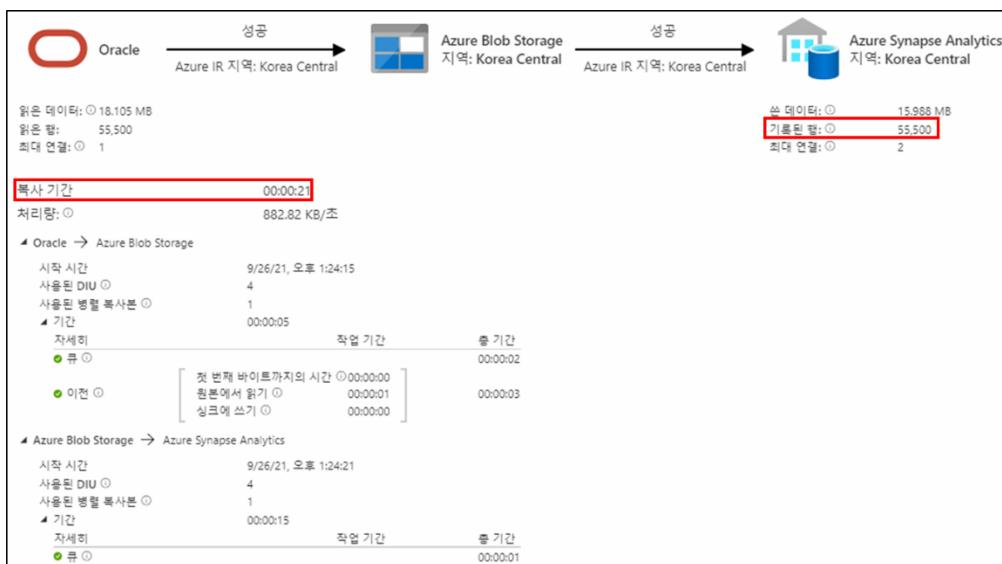


12. 파이프 라인에서 생성했던 작업 이름 Copy data1 항목에서 자세히(안경 이미지) 아이콘을 클릭합니다.

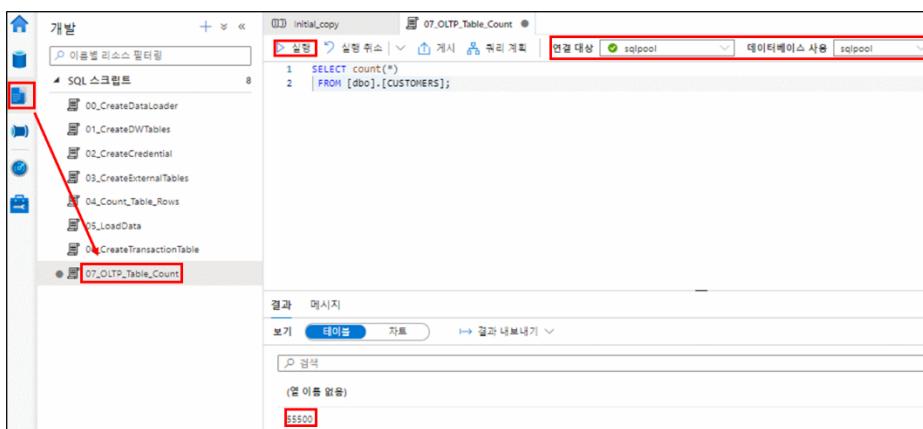
## [Azure Synapse Analytics Hands-on Labs]



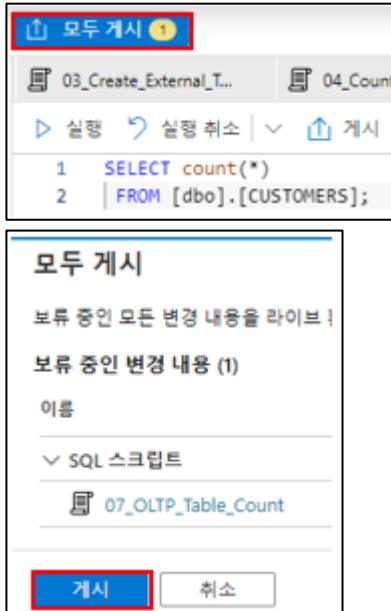
13. 파이프 라인이 수행된 프로세스와 내용을 확인할 수 있습니다.



14. 마지막으로 개발 Hub 로 돌아와서 07\_OLTP\_Table\_Count 스크립트를 수행하여 결과 건수를 확인합니다. 55500 값이 나오면 파이프라인을 사용한 데이터 Load 가 정상적으로 수행된 것 입니다.



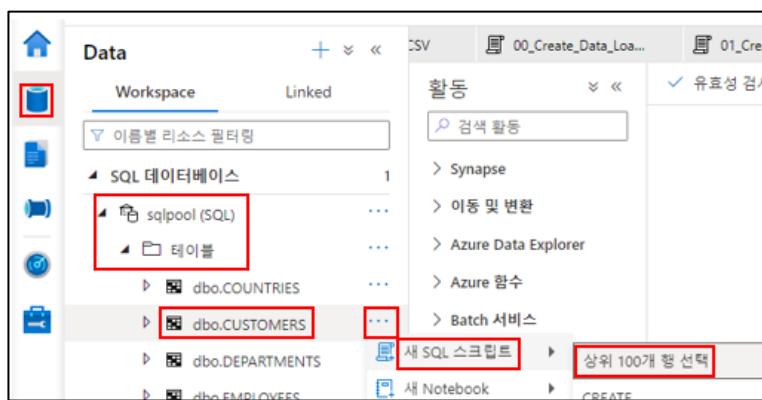
15. 07\_OLTP\_Table\_Count 의 변경내용을 게시합니다.



## Lab 4: 데이터 조회

### Task 1. SQL Script를 이용한 데이터 조회

- 데이터 Hub로 이동하여 sqlpool의 dbo.CUSTOMERS 테이블의 우측 옵션을 누릅니다. (dbo.CUSTOMERS 테이블이 보이지 않을 경우, sqlpool (SQL)에 마우스를 올려놓고 우클릭 후, '새로고침'을 누릅니다.)
- 새 SQL 스크립트를 눌러 상위 100개 행 선택을 합니다.



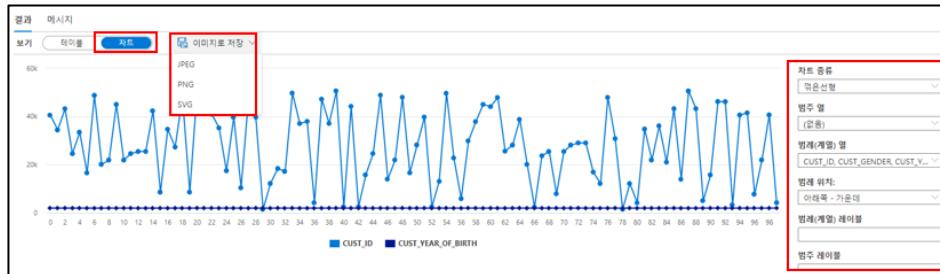
- dbo.CUSTOMERS 테이블의 Top100 rows를 조회하기 위한 쿼리가 자동 생성되며 수행된 결과 데이터가 노출됩니다.

CUST_ID	CUST_FIRST_N...	CUST_LAST_N...	CUST_GENDER	CUST_YEAR_O...	CUST_MARITA...	CUST_STREET_...	CUST_POSTAL_...	CUST_CITY	CUST_CITY_ID
40571	Annie	Kotzman	F	1968	married	107 South Pass...	46063	Long Meadow	51805
34425	Angela	Gentile	F	1982	single	17 North Kenton...	78558	Los Angeles	51806
43242	Armand	Ling	M	1969	(NULL)	117 West Gratia...	72860	Koeln	51735
24573	Antony	Chinn	M	1962	single	77 North Patriotic...	38878	Hilversum	51662
33462	Archibald	Patterson	M	1966	single	97 West Meriweth...	31132	Saint Marks	52279
16579	Arthur	Poe	M	1944	single	67 Perkins Avenue...	57128	Barry	51166
48802	Ashley	Oxford	F	1938	single	27 West Union Street	47195	Shipman	52310

결과 내보내기 버튼을 눌러 CSV, JSON, XML 의 파일형식을 개인의 Local PC 에 저장할 수 있습니다.

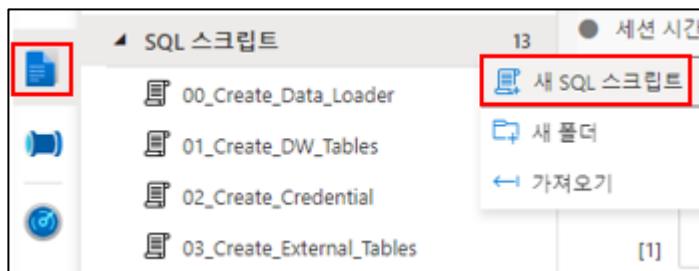
- 또한, 차트 버튼을 눌러 결과 데이터에 대한 차트를 나타낼 수 있습니다.

차트의 종류, 범주, 범례를 선택하여 제한적이나 자신에게 원하는 차트의 형태를 표현할 수 있으며, 차트 결과를 JPEG, PNG, SVG 의 결과로 저장할 수 있습니다.



- Synapse Studio 의 자동 쿼리 생성 기능 외에도, 사용자의 자유로운 T-SQL 작성에 의해 데이터 조회가 가능합니다.

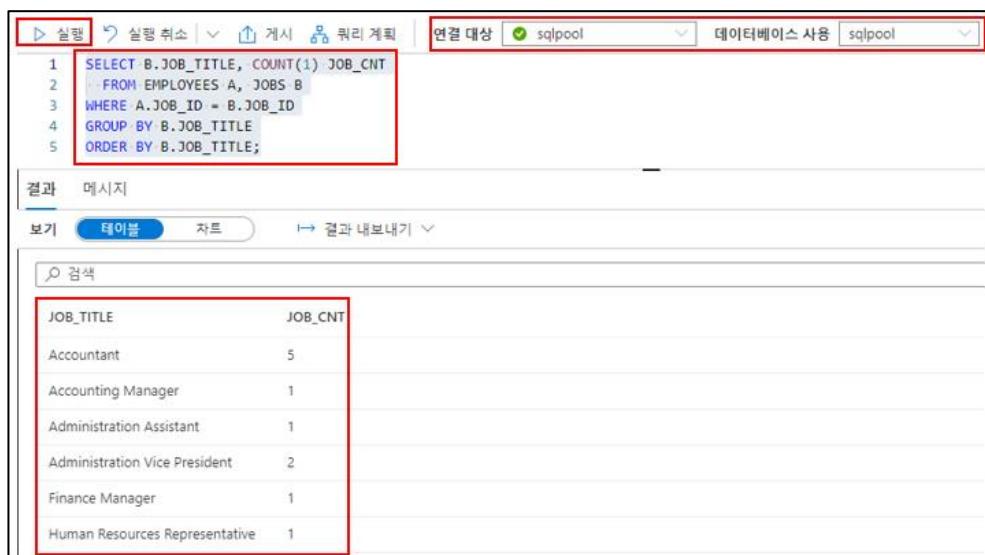
Develop Hub 로 이동하여 새 SQL 스크립트를 누릅니다.



6. 새로운 SQL 스크립트 창에서 아래 쿼리 스크립트를 적습니다.  
 (연결 대상 'sqlpool' / 데이터베이스 사용 'sqlpool')  
 연결대상으로 sqlpool, 데이터베이스를 sqlpool로 선택한 뒤 실행 버튼을 누릅니다

```
SELECT B.JOB_TITLE, COUNT(1) JOB_CNT
FROM EMPLOYEES A, JOBS B
WHERE A.JOB_ID = B.JOB_ID
GROUP BY B.JOB_TITLE
ORDER BY B.JOB_TITLE;
```

7. 결과를 확인합니다. 아래와 같이 결과가 나오면 정상적으로 T-SQL 을 이용한 데이터 조회가 수행된 것 입니다.

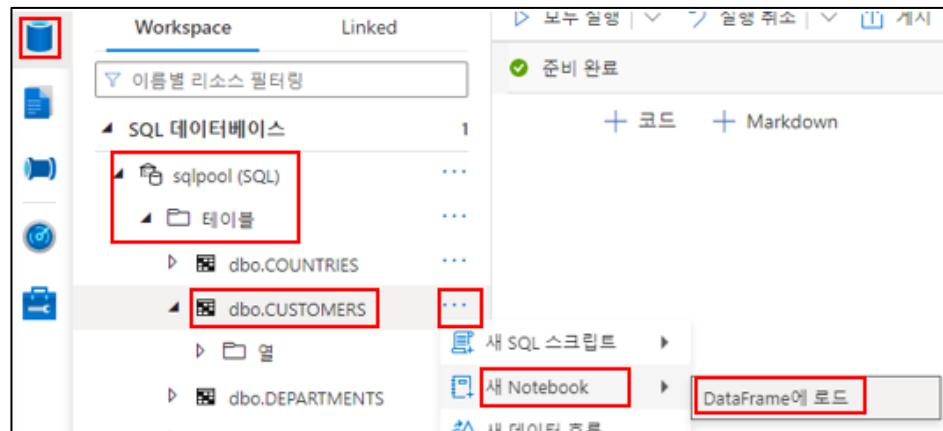


JOB_TITLE	JOB_CNT
Accountant	5
Accounting Manager	1
Administration Assistant	1
Administration Vice President	2
Finance Manager	1
Human Resources Representative	1

## Task 2. DataFrame을 이용한 데이터 조회

1. dbo.CUSTOMERS 테이블을 Spark Notebook 의 DataFrame 에서도 조회할 수 있습니다.
2. 데이터 Hub 로 이동하여 sqlpool 의 dbo.CUSTOMERS 테이블의 우측 옵션을 누릅니다.
3. DataFrame 에 로드를 선택합니다.

## [Azure Synapse Analytics Hands-on Labs]



4. 새로운 Notebook의 DataFrame에 dbo.customers 데이터를 읽고 default.t1 테이블에 데이터를 저장하는 code가 자동 생성됩니다. 코드를 수행할 연결대상을 사전에 생성해 둔 sparkpool을 선택 후, 셀 실행 버튼을 누릅니다.  
(Notebook을 처음 수행 시, 초기 인스턴스를 생성하는 시간이 필요하여 코드를 수행하는데 2~3분의 시간이 소요될 수 있습니다.)

```

1 %%spark
2 val df = spark.read.synapsesql("sqlpool.dbo.CUSTOMERS")
3 df.write.mode("overwrite").saveAsTable("default.t1")

```

[1] ✓ 2분 45초 - 2 분 9초 889 ms에서 Apache Spark 세션이 시작되었습니다. 1:36  
작업 실행 성공 Spark 실행기 2개 코어 16개

... df: org.apache.spark.sql.DataFrame = [CUST\_ID: int, CUST\_FIRST\_NAME:

+ 코드 + Markdown

5. 위 코드는 default.t1의 테이블에 dbo.CUSTOMER 테이블 데이터를 Load 만 했지만 결과를 노출하진 않습니다. + 코드 를 눌러 새로운 셀을 만들고 아래 명령어를 입력합니다.

```

1 %%spark
2 val df = spark.read.synapsesql("sqlpool.dbo.CUSTOMERS")
3 df.write.mode("overwrite").saveAsTable("default.t1")
Shift +Enter를 눌러 실행

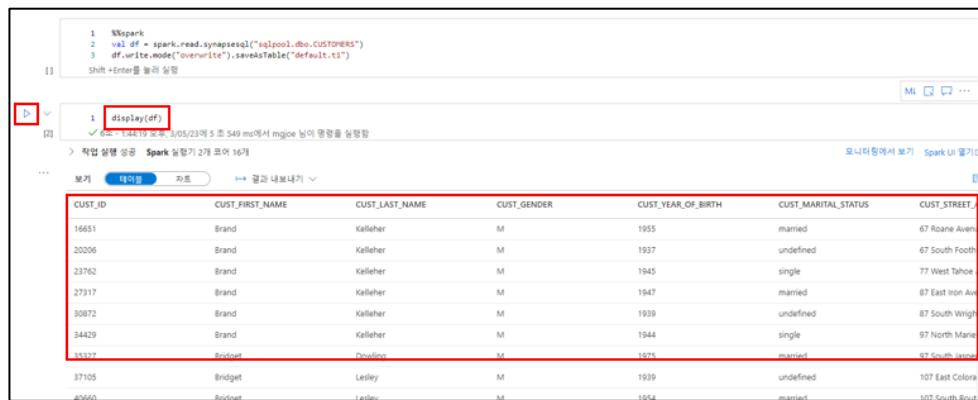
```

+ 코드 + Markdown

display(df)

## [Azure Synapse Analytics Hands-on Labs]

### 6. 2 번 셀을 실행하고 결과를 확인합니다.



```

1 %spark
2 val df = spark.read.synapsesql("sqlpool.dbo.CUSTOMERS")
3 df.write.mode("overwrite").saveAsTable("default.t1")
Shift +Enter를 눌러 실행

```

The screenshot shows the execution results of the first cell. A red box highlights the command `display(df)`. The output shows a table with 10 rows of customer data. A red box highlights the entire table.

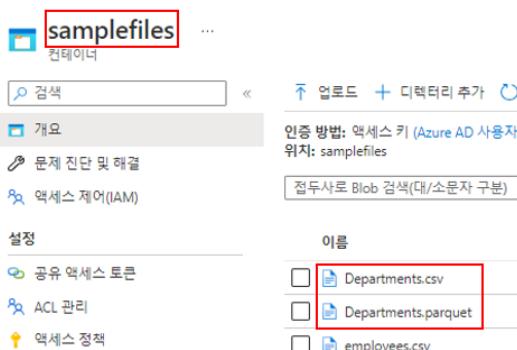
CUST_ID	CUST_FIRST_NAME	CUST_LAST_NAME	CUST_GENDER	CUST_YEAR_OF_BIRTH	CUST_MARITAL_STATUS	CUST_STREET_...
16651	Brand	Kelleher	M	1955	married	67 Roane Aven...
20206	Brand	Kelleher	M	1937	undefined	67 South Footh...
23762	Brand	Kelleher	M	1945	single	77 West Tahoe...
27317	Brand	Kelleher	M	1947	married	87 East Iron Av...
30872	Brand	Kelleher	M	1939	undefined	87 South Wrig...
34429	Brand	Kelleher	M	1944	single	97 North Marie...
35337	Bridget	Dowling	M	1975	married	97 South Japan...
37105	Bridget	Lesley	M	1939	undefined	107 East Colora...
40560	Brigid	Lesley	M	1954	married	107 South Bou...

### 7. 수행한 Notebook 창을 닫거나 인스턴스의 세션을 중지합니다.



### Task 3. 외부 데이터 조회

1. Serverless SQL Pool 과 Spark Dataframe 을 이용하여 여러분이 생성한 ADLS Gen2 스토리지의 sampleFiles 컨테이너 내부에 있는 Departments.csv, Departments.parquet 파일의 데이터를 조회하겠습니다.



The screenshot shows the Azure Storage Explorer interface. A red box highlights the 'samplefiles' container. The container list shows three files: `Departments.csv`, `Departments.parquet`, and `employees.csv`.

## [Azure Synapse Analytics Hands-on Labs]

2. Develop hub → + 버튼 → SQL 스크립트로 이동하여 새로운 SQL 스크립트 창을 엽니다.



3. Departments.csv 를 조회하기 위해 연결대상은 기본제공, 데이터베이스는 master 를 선택한 다음 아래 쿼리를 기재 후 실행 버튼을 누릅니다.  
(연결대상 '기본제공' / 데이터베이스 사용 'master')

```
SELECT *
FROM OPENROWSET(
    BULK 'https://<username>adls.dfs.core.windows.net/samplefiles/Departments.csv',
    FORMAT = 'CSV',
    PARSER_VERSION = '2.0',
    HEADER_ROW = TRUE
) AS [result]
```

4. 실행 결과 데이터를 확인합니다.

The screenshot shows the Azure Synapse Analytics Query Editor. At the top, there's a toolbar with various icons. Below it, the '연결 대상' dropdown is set to '기본 제공' and the '데이터베이스 사용' dropdown is set to 'master'. The main area contains a code editor with the previously provided T-SQL script. A red box highlights the entire code block. Below the code editor, there's a table titled '결과' (Results) with a single row labeled '테이블' (Table). The table displays the data from the 'Departments' CSV file:

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing	114	1700
40	Human Resources	203	2400
50	Shipping	121	1500

## [Azure Synapse Analytics Hands-on Labs]

5. 다음으로 Departments.parquet 파일을 조회하겠습니다. Develop hub → + 버튼 → SQL 스크립트로 이동하여 새로운 SQL 스크립트 창을 엽니다.



6. Parquet 파일을 조회하기 위해 연결대상은 기본제공, 데이터베이스는 master 를 선택한 다음 아래 쿼리를 기재 후 실행 버튼을 누릅니다.  
(연결대상 '기본제공' / 데이터베이스 사용 'master')

```
SELECT *
FROM OPENROWSET(
    BULK 'https://<username>adls.dfs.core.windows.net/samplefiles/Departments.parquet',
    FORMAT = 'PARQUET'
) AS [result];
```

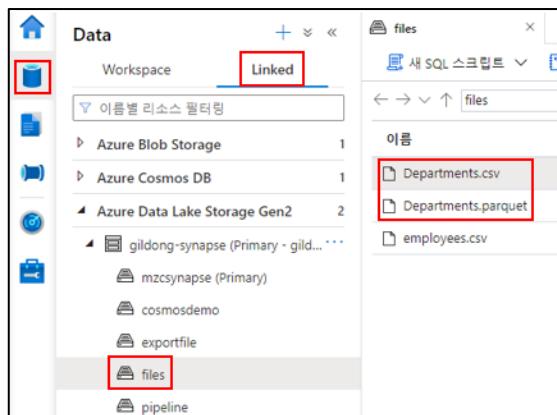
7. 결과를 확인합니다.

A screenshot of the Azure Synapse Analytics Query Editor. At the top, there are several buttons: '실행' (Run), '실행 취소' (Cancel), '게시' (Publish), '모두' (All), '쿼리 계획' (Query Plan), '연결 대상' (Connection Target) set to '기본 제공' (Default), '데이터베이스 사용' (Database Usage) set to 'master', and a dropdown for 'master'. Below the toolbar, the query text is shown in a code editor with syntax highlighting. The results section shows a table with the following data:

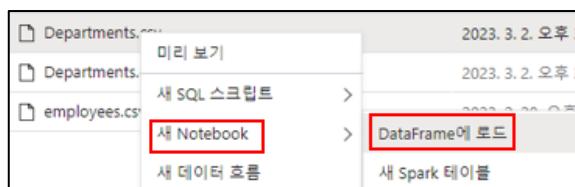
DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing	114	1700
40	Human Resources	203	2400
50	Shipping	121	1500
60	IT	103	1400
70	Public Relations	204	2700
80	Sales	145	2500

8. 위와 같이 SQL 스크립트를 직접 작성하지 않고 GUI의 스크립트 자동 생성기능을 이용하여 쉽게 데이터를 조회할 수 있습니다.

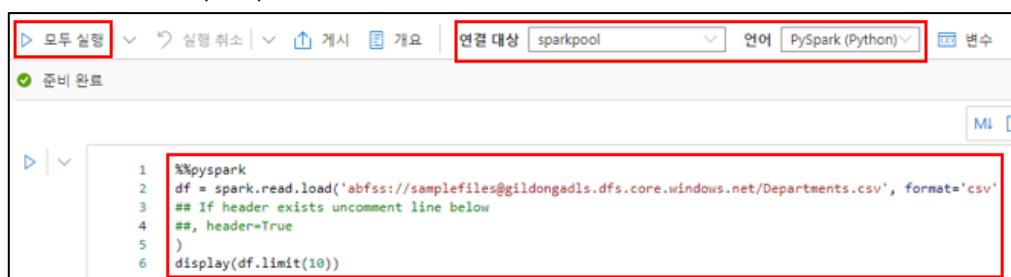
9. Data hub → Linked → Azure Data Lake Storage Gen2 SQL → samplefiles 컨테이너로 이동합니다. 컨테이너 안의 Departments.csv, Departments.parquet 파일을 확인합니다.



10. 위에서는 SQL 스크립트를 이용하였으니, 이번에는 Dataframe 을 사용하여 데이터 조회를 하겠습니다. Departments.csv 파일에서 마우스 우측버튼 → 새 Notebook → DataFrame에 로드를 선택합니다.



11. 새로운 Notebook 창에 Dataframe 을 사용한 데이터 조회 코드가 자동생성 됩니다. 연결 대상으로 sparkpool 로 선택하고 실행 버튼을 누릅니다.



12. 조회 결과를 확인합니다.

[Azure Synapse Analytics Hands-on Labs]

_c0	_c1	_c2	_c3
DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing	114	1700
40	Human Resources	203	2400
50	Shipping	121	1500
60	IT	103	1400
70	Public Relations	204	2700

13. 컬럼 헤더 정보가 `_c0`, `_c1`, `_c2..`로 나왔습니다. 해당 컬럼 헤더 정보를 정확한 컬럼명으로 변경하기 위해서 `##, header=True`의 주석을 제거하고 다시 실행 합니다.

```
1 %%pyspark
2 df = spark.read.load('abfss://samplefiles@gildongadls.dfs.core.windows.net/Departments.csv', format='csv')
3 ## If header exists uncomment line below
4 , header=True
5 )
6 display(df.limit(10))
```

✓ <1초 - 2:20:27 이후, 3/05/23에 576 ms에서 mgjoe 님이 명령을 실행함

> 작업 실행 성공 Spark 실행기 2개 코어 16개 모니터링에서 보기

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing	114	1700
40	Human Resources	203	2400
50	Shipping	121	1500
60	IT	103	1400
70	Public Relations	204	2700
80	Sales	145	2500

14. Departments.csv 의 컬럼과 데이터가 정상적으로 출력된 것을 확인할 수 있습니다.

15. 현재 수행하는 Notebook 의 세션을 중지합니다.

```
준비 완료

> | < 1 %%pyspark
2 df = spark.read.load('abfss://samplefiles@gildongadls.dfs.core.windows.net/Departments.csv', format='csv')
3 ## If header exists uncomment line below
4 , header=True
5 )
6 display(df.limit(10))
[5] ✓ <1초 - 2:20:27 오후, 3/05/23에 576 ms에서 mgjoe님이 명령을 실행함
```

16. Departments.parquet 도 10~14 번을 순차적으로 수행하여 아래 결과를 확인할 수 있습니다.

The screenshot shows the Azure Synapse Analytics Studio interface. At the top, there's a file tree with 'Departments.parquet' selected. A context menu is open with options: '새 SQL 스크립트', '새 Notebook' (highlighted with a red box), '새 데이터 흐름', 'DataFrame에 토드' (highlighted with a red box), and '새 Spark 테이블'. Below the file tree, a code editor window displays the following Python code:

```

1 %%pyspark
2 df = spark.read.load('abfss://files@gildongadls.dfs.core.windows.net/Departments.parquet', format='parquet')
3 display(df.limit(10))

```

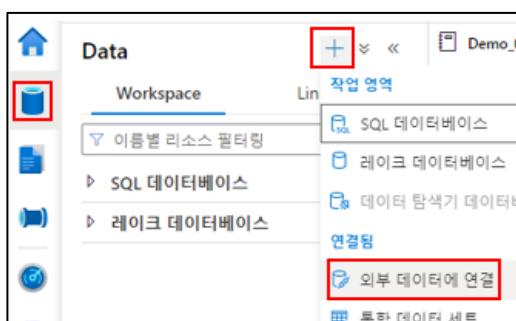
A green checkmark indicates the command was successful: "2초 - 5:45:40 오후, 3/02/23에 1 초 952 ms에서 mgjoe님이 명령을 실행함". Below the code editor is a log message: "작업 실행 성공 Spark 실행기 2개 코어 16개". At the bottom, there's a table preview with the following data:

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing	114	1700
40	Human Resources	203	2400
50	Shipping	121	1500
60	IT	103	1400
70	Public Relations	204	2700
80	Sales	145	2500
90	Executive	100	1700

## Lab 4: Cosmos DB 데이터 조회

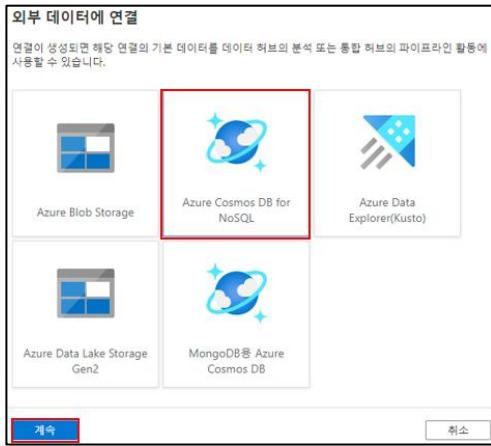
### Task 1. Cosmos DB 연결

1. Synapse Link 기능을 사용하여 Cosmos DB 와 Synapse Analytics 를 연결하고 Cosmos DB 의 데이터를 조회하는 방법입니다.
2. 데이터 허브로 이동합니다. 상단의 +를 클릭하고 외부 데이터에 연결을 클릭합니다.

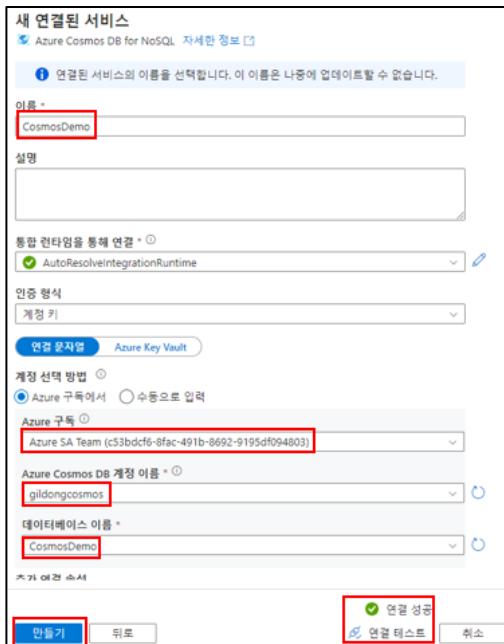


## [Azure Synapse Analytics Hands-on Labs]

3. Azure Cosmos DB(SQL API)를 선택합니다. 계속을 클릭합니다.



4. Linked Service의 이름과 Cosmos DB의 정보를 입력합니다. 연결 테스트를 진행하고 성공적으로 연결이 되는 것을 확인한 후, 만들기를 클릭합니다.



이름 : CosmosDemo

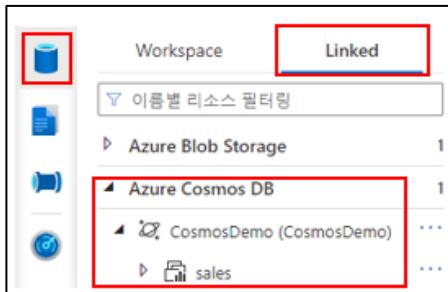
Azure 구독 : Cosmos DB 를 만든 구독 선택(현재 구독)

Azure Cosmos DB 계정 이름 : <username>cosmos

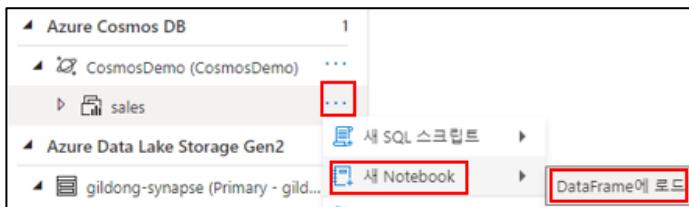
데이터베이스 이름 : CosmosDemo

5. 페이지를 새로고침(Refresh)한 후 Data Hub → Linked 로 들어가 생성된 Azure Cosmos DB 를 확인합니다.

## [Azure Synapse Analytics Hands-on Labs]



6. Sales 컨테이너 옵션을 누릅니다. → 새 Notebook → DataFrame에 로드를 누릅니다.



7. 스크립트가 자동생성 됩니다. 연결대상을 sparkpool로 선택하고 셀을 실행하여 결과를 확인합니다.

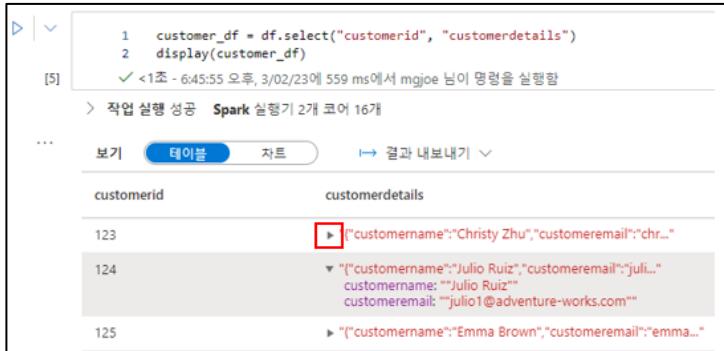
_rid	_ts	id	orderdate	cus
nK\$XAKVUSpcCAAAAAAAA==	1677749023	SO43704	2019-07-01	124
nK\$XAKVUSpcBAAAAAAA==	1677749001	SO43701	2019-07-01	125
nK\$XAKVUSpcDAAAAAAA==	1677749192	SO43707	2019-07-02	125

8. 결과에는 \_rid, \_ts 와 같은 일부 메타데이터가 포함되어 반환됩니다. + 버튼을 눌러 새로운 셀을 추가하여 아래 코드를 입력 후 셀을 실행합니다.

```
customer_df = df.select("customerid", "customerdetails")
display(customer_df)
```

9. 위 쿼리는 customerid, customerdetails 의 개별 컬럼의 정보만 조회합니다. Customerdetails 컬럼의 경우 컬럼 항목 내 다중 데이터에 대한 JSON 구조가 포함되어 있는지 확인합니다. 결과 항목에서 ▶ 버튼을 눌러 개별 데이터를 확인할 수 있습니다.

## [Azure Synapse Analytics Hands-on Labs]

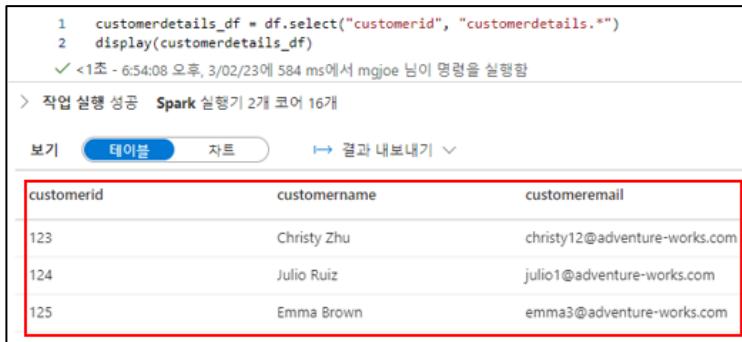


```
customerdetails = df.select("customerid", "customerdetails")
display(customerdetails)
```

10. 다시 + 버튼을 눌러 새로운 셀을 추가 후 아래 코드를 넣어 실행합니다.

```
customerdetails_df = df.select("customerid", "customerdetails.*")
display(customerdetails_df)
```

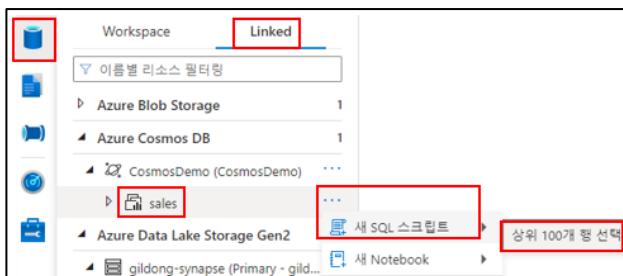
11. 결과를 확인합니다. 위 쿼리의 경우 customerdetails 컬럼내에 존재하는 모든 컬럼에 대하여 나열하는 결과를 반환합니다.



```
customerdetails_df = df.select("customerid", "customerdetails.*")
display(customerdetails_df)
```

12. 이번에는 Serverless sqlpool 로 cosmos DB 의 데이터를 조회하겠습니다.

Data Hub → Linked → sales 옵션 → 새 SQL 스크립트 → 상위 100 개 행 선택을 누릅니다.



## [Azure Synapse Analytics Hands-on Labs]

13. 아래와 같이 자동스크립트가 생성됩니다. SQL 풀은 Cosmos DB에 액세스할 때 사용할 자격 증명이 필요합니다. 때문에 수정 없이 스크립트를 실행하면 오류가 발생합니다.

```

1 IF (NOT EXISTS(SELECT * FROM sys.credentials WHERE name = 'gildongcosmos'))
2     THROW 50000, 'As a prerequisite, create a credential with Azure Cosmos DB key in SECRET option:
3     CREATE CREDENTIAL [gildongcosmos]
4     WITH IDENTITY = ''SHARED ACCESS SIGNATURE'', SECRET = ''<Enter your Azure Cosmos DB key here>'', 0
5 GO
6
7 SELECT TOP 100 *
8 FROM OPENROWSET(PROPRIETOR = 'CosmosDB',
9                  CONNECTION = 'Account=gildongcosmos;Database=CosmosDemo',
10                 OBJECT = 'sales',
11                 SERVER_CREDENTIAL = 'gildongcosmos'
12 ) AS [sales]

```

자격증명을 위한 Key 값을 얻기 위해, portal에서 자신이 생성한 Azure Cosmos DB 계정으로 들어간 후 키 → 기본키에서 key 값을 클립보드로 복사합니다.



14. 전체 스크립트를 아래와 같이 변경합니다.

13 번에서 복사한 기본키 값을 <Enter your Azure Cosmos DB key here>에 붙여넣고 전체 쿼리를 실행합니다.

```

--변경 전
IF (NOT EXISTS(SELECT * FROM sys.credentials WHERE name = 'gildongcosmos'))
    THROW 50000, 'As a prerequisite, create a credential with Azure Cosmos DB key in SECRET option:
CREATE CREDENTIAL [gildongcosmos]
WITH IDENTITY = ''SHARED ACCESS SIGNATURE'', SECRET = ''<Enter your Azure Cosmos DB key here>'', 0
GO

SELECT TOP 100 *
FROM OPENROWSET(PROPRIETOR = 'CosmosDB',
                CONNECTION = 'Account=gildongcosmos;Database=CosmosDemo',
                OBJECT = 'sales',
                SERVER_CREDENTIAL = 'gildongcosmos'
) AS [sales]

```

[Azure Synapse Analytics Hands-on Labs]

```
--변경 후

CREATE CREDENTIAL [<username>cosmos]
WITH IDENTITY = 'SHARED ACCESS SIGNATURE',
SECRET = 'niF6snar7dGSD5q0sbGQJCbXeLOpZelh9go5sk0UD3DbyvIJRxFovXMeyVM579XxDG87Cc5jw00
ACDbQ73VdA=='
GO

SELECT TOP 100 *
FROM OPENROWSET(PROPRIETOR = 'CosmosDB',
                CONNECTION = 'Account=gildongcosmos;Database=CosmosDemo',
                OBJECT = 'sales',
                SERVER_CREDENTIAL = 'gildongcosmos'
) AS [sales]
```

- ## 15. 쿼리 결과 값을 확인합니다.

```
▷ 실행 🔍 실행 취소 ✎ 게시 ⌂ 워크 계획 연결 대상 기본 제공 데이터베이스 사용 master

1 CREATE CREDENTIAL [gildongcosmos]
2 WITH IDENTITY = 'SHARED ACCESS SIGNATURE',
3 SECRET = 'n1f6snar7dQSD5q0sbGQJcbXeL0p2elh9go5sk0u0D3byv1JRxFovXMeYVM579XxD5G87Cc5jw00ACDbQ73VdA=='
4 GO
5
6
7 SELECT TOP 100 *
8 FROM OPENROWSET(PROVIDER = 'CosmosDB',
9 CONNECTION = 'Account=gildongcosmos;Database=CosmosDemo',
10 OBJECT = 'sales',
11 SERVER_CREDENTIAL = 'gildongcosmos'
12 ) AS [sales]
13
```

16. `_rid`, `quantity`, `_ts` 와 같이 필요하지 않은 메타데이터의 컬럼이 함께 출력됩니다. 출력되기  
    를 원하는 컬럼과 데이터 타입을 지정하여 쿼리를 수행할 수 있습니다.

17. SQL 스크립트의 모든 내용을 삭제하고 아래 스크립트를 기입하여 쿼리를 실행합니다.  
(Credential 은 한 번만 수행하면 되므로, 스크립트내 해당 내용을 삭제하셔도 됩니다.)

```
SELECT TOP 100 *
FROM OPENROWSET(PROPRIETOR = 'CosmosDB',
                CONNECTION = 'Account=<username>cosmos;Database=CosmosDemo',
                OBJECT = 'sales',
                SERVER_CREDENTIAL = '<username>cosmos'
)
with (
```

```

OrderID VARCHAR(10) '$.id',
OrderDate VARCHAR(10) '$.orderdate',
CustomerID INTEGER '$.customerid',
CustomerName VARCHAR(40) '$.customerdetails.customername',
CustomerEmail VARCHAR(30) '$.customerdetails.customeremail',
Product VARCHAR(30) '$.product',
Quantity INTEGER '$.quantity',
Price FLOAT '$.price'
)
AS [sales]

```

18. 결과를 확인합니다.

The screenshot shows the Azure Synapse Analytics Studio interface. At the top, there are tabs for '실행' (Run), '실행 취소' (Cancel), '계시' (Help), and '쿼리 계획' (Query Plan). Below these are connection settings: '연결 대상' (Connection Target) set to '기본 제공' (Default), '데이터베이스 사용' (Database Usage) set to 'master', and a 'master' server icon. The main area contains a T-SQL query:

```

1 SELECT TOP 100 *
2 FROM OPENROWSET(PVODIER = 'CosmosDB',
3                   CONNECTION = 'Account=gildongcosmos;Database=CosmosDemo',
4                   OBJECT = 'sales',
5                   SERVER_CREDENTIAL = 'gildongcosmos'
6 )
7 with (
8     OrderID VARCHAR(10) '$.id',
9     OrderDate VARCHAR(10) '$.orderdate',
10    CustomerID INTEGER '$.customerid',
11    CustomerName VARCHAR(40) '$.customerdetails.customername',
12    CustomerEmail VARCHAR(30) '$.customerdetails.customeremail',
13    Product VARCHAR(30) '$.product',
14    Quantity INTEGER '$.quantity',
15    Price FLOAT '$.price'
16 )
17 AS [sales]

```

The results pane shows a table with the following data:

OrderID	OrderDate	CustomerID	CustomerName	CustomerEmail	Product	Quantity	Price
SO43707	2019-07-02	125	Emma Brown	emma3@adventure-works.com	Road-150 Red, 48	1	3578.27
SO43704	2019-07-01	124	Julio Ruiz	julio1@adventure-works.com	Mountain-100 Black, 48	1	3374.99
SO43701	2019-07-01	123	Christy Zhu	christy12@adventure-works.com	Mountain-100 Silver, 44	1	3399.99