

※ Linux 환경에서 AgensGraph 설치

① 환경변수 설정 (선택사항)

`/usr/local/AgensGraph/`에 패키지(`AgensGraph_v2.1.0_linux_CE.tar.gz`)를 다운받고 압축을 해제했다고 가정하고 `.bash_profile`과 같은 쉘 시작파일에 다음 3줄을 추가합니다.

```
export LD_LIBRARY_PATH=/usr/local/AgensGraph/lib:$LD_LIBRARY_PATH
export PATH=/usr/local/AgensGraph/bin:$PATH
export AGDATA=/path/to/make/db_cluster
```

```
[agens@:/usr/local/AgensGraph]#cat .bash_profile
export LD_LIBRARY_PATH=/usr/local/AgensGraph/lib:$LD_LIBRARY_PATH
export PATH=/usr/local/AgensGraph/bin:$PATH
export AGDATA=/usr/local/AgensGraph/db_cluster
```

② 데이터베이스 클러스터 생성

다음 명령을 통해 데이터베이스 클러스터를 생성합니다.

```
initdb [-D /path/to/make/db_cluster]
```

```
[agens@:/usr/local/AgensGraph]#initdb -D /usr/local/AgensGraph/db_cluster
The files belonging to this database system will be owned by user "agens".
This user must also own the server process.

The database cluster will be initialized with locale "ko_KR.UTF-8".
The default database encoding has accordingly been set to "UTF8".
initdb: could not find suitable text search configuration for locale "ko_KR.UTF-8"
The default text search configuration will be set to "simple".

Data page checksums are disabled.

creating directory /usr/local/AgensGraph/db_cluster ... ok
creating subdirectories ... ok
selecting default max_connections ... 100
selecting default shared_buffers ... 128MB
selecting dynamic shared memory implementation ... posix
creating configuration files ... ok
running bootstrap script ... ok
performing post-bootstrap initialization ... ok
syncing data to disk ... ok

WARNING: enabling "trust" authentication for local connections
You can change this by editing pg_hba.conf or using the option -A, or
--auth-local and --auth-host, the next time you run initdb.

Success. You can now start the database server using:

    ag_ctl -D /usr/local/AgensGraph/db_cluster -l logfile start
```

③ 서버 구동

아래의 명령어를 이용하여 AgensGraph를 구동합니다.

```
ag_ctl start [-D /path/created/by/initdb]
```

```
[agens@:/usr/local/AgensGraph]#ag_ctl -D /usr/local/AgensGraph/db_cluster -l logfile start  
server starting
```

④ 데이터베이스 생성 및 터미널 실행

아래의 명령어를 이용하여 AgensGraph상에 DB를 생성하고 해당 DB로 접속하는 AgensGraph 터미널을 실행합니다.

```
createdb [dbname]
```

```
agens [dbname]
```

```
username=#
```

```
[agens@:bitnine01]$ createdb  
[agens@:bitnine01]$ agens  
agens (AgensGraph 2.1.0, based on PostgreSQL 10.4)  
Type "help" for help.  
agens=#
```

Super User인 경우 프롬프트에 "#=#"으로 표시되고 그외의 유저는 "=>"으로 표시됩니다.

```

[agens@:bitnine01]$ agens
agens (AgensGraph 2.1.0, based on PostgreSQL 10.4)
Type "help" for help.

agens=# CREATE GRAPH test_graph1;
CREATE GRAPH
agens=# SET graph_path = test_graph1;
SET
agens=# CREATE (testNode:Test {name: 'TestNode'});
GRAPH WRITE (INSERT VERTEX 1, INSERT EDGE 0)
agens=# MATCH (testNode:Test {name: 'TestNode'})
agens=# RETURN testNode;
          testnode
-----
 test[3.1]{"name": "TestNode"}
(1 row)

```

※ Windows 환경에서 AgensGraph 설치

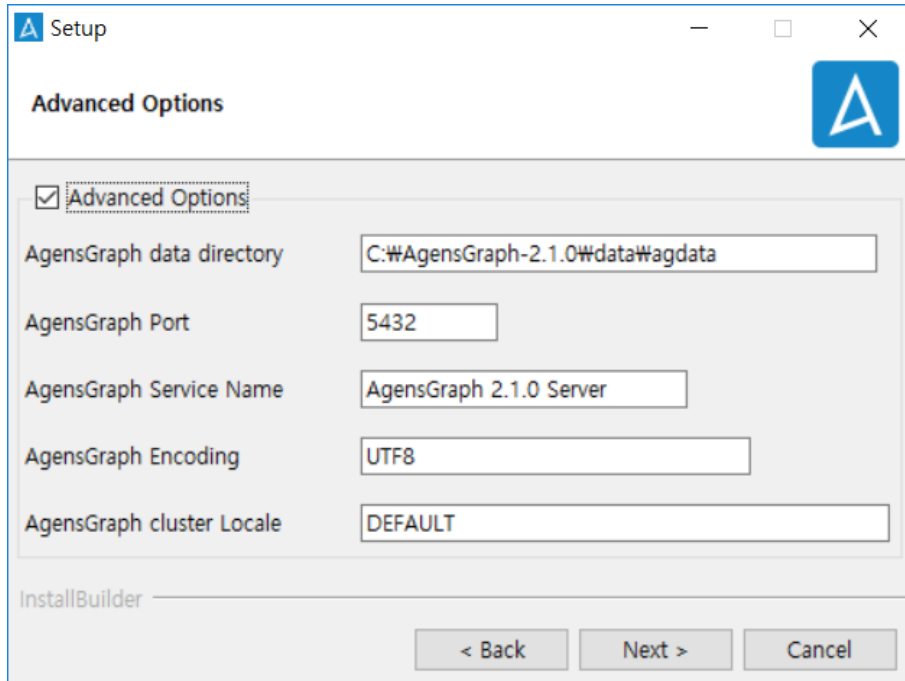
- ① 설치 파일을 관리자 권한으로 실행

다운받은 설치파일(예: AgensGraph-2.1.0-windows-installer)를 관리자 권한으로 실행합니다.

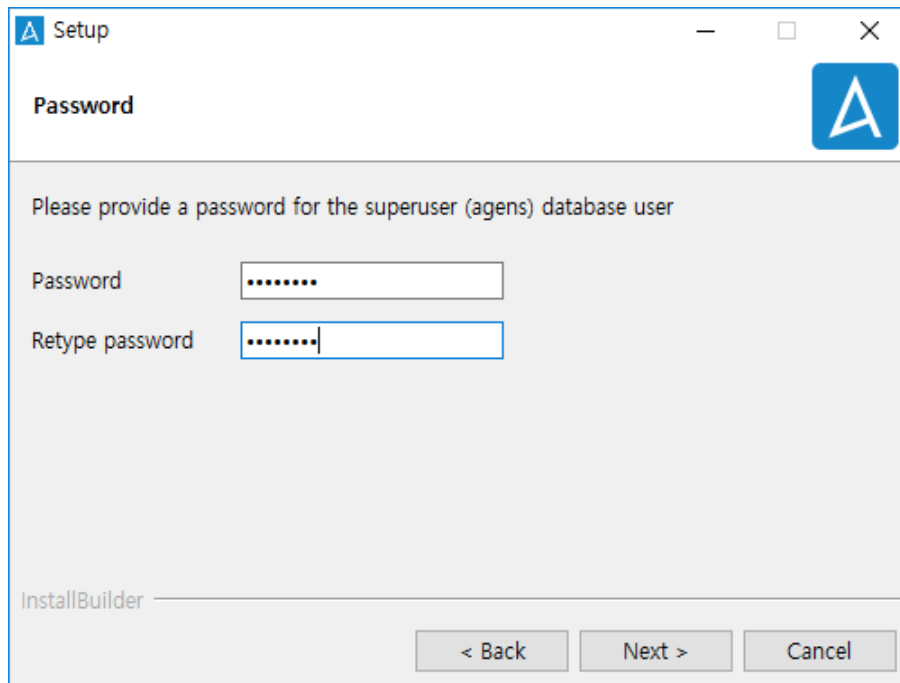
- ② 라이선스 조건에 동의(Accept)하고 설치 경로를 지정합니다.



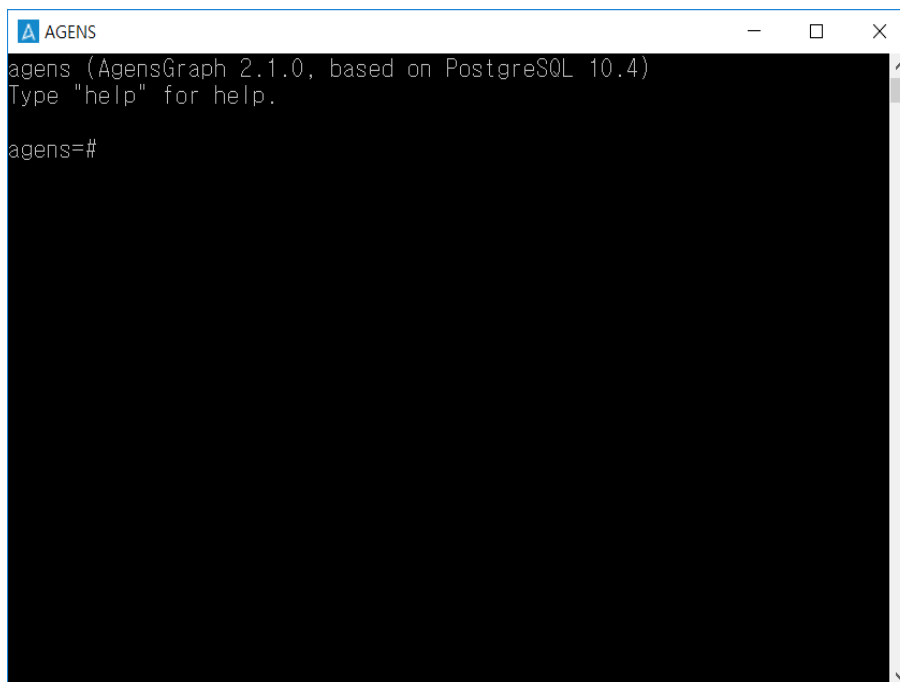
- ③ 설치 경로, 사용할 인코딩, DB에 할당할 포트를 변경하는 등 설치 설정을 변경합니다(변경없이 설치하려면 Advanced Options 체크박스를 해제하고 진행).



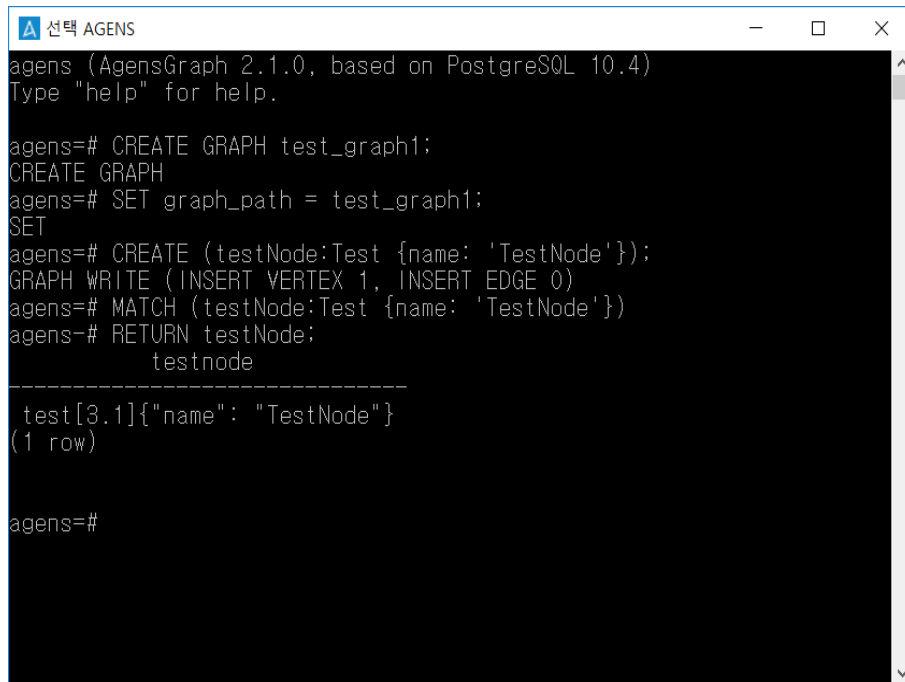
- ④ Superuser(기본 설정된 Superuser는 agens) 계정이 사용할 비밀번호를 설정하고 다음을 누릅니다. 설치 옵션을 최종 확인하는 화면이 나타나고 다음을 누르면 설치가 진행됩니다.



⑤ 설치가 완료된 후 윈도우(Windows) 시작 프로그램에 등록된 **AGENS** 아이콘(📌)을 클릭하면 아래와 같이 AgensGraph가 구동됩니다.



⑥ 정상적으로 AgensGraph가 구동하는지 확인하기 위해 샘플 그래프를 생성하고, 그래프 Path를 설정한 다음, Vertex 1개를 만들어 반환하는 쿼리를 수행했습니다. 아래 화면과 같이 모든 과정이 정상적으로 실행됨을 확인할 수 있습니다.



```
선택 AGENS
agens (AgensGraph 2.1.0, based on PostgreSQL 10.4)
Type "help" for help.

agens=# CREATE GRAPH test_graph1;
CREATE GRAPH
agens=# SET graph_path = test_graph1;
SET
agens=# CREATE (testNode:Test {name: 'TestNode'});
GRAPH WRITE (INSERT VERTEX 1, INSERT EDGE 0)
agens=# MATCH (testNode:Test {name: 'TestNode'})
agens-# RETURN testNode;
          testnode
-----
 test[3.1>{"name": "TestNode"}
(1 row)

agens=#
```

각 설치방법에 관한 자세한 내용을 알기 원하시면 상기 링크 또는 [AgensGraph 개발자 매뉴얼](#)을 참조하시기 바랍니다.