

# Backend.AI Console User's Guide

(19.09)

Lablup Inc.



---

## 목차

---

<b>0</b>	<b>면책조항</b>	<b>2</b>
<b>1</b>	<b>개요</b>	<b>4</b>
1.1	주요 개념 . . . . .	5
<b>2</b>	<b>설치</b>	<b>8</b>
<b>3</b>	<b>가입 및 로그인</b>	<b>9</b>
3.1	가입 . . . . .	9
3.2	로그인 . . . . .	11
3.3	비밀번호를 잊어버렸을 경우 . . . . .	12
<b>4</b>	<b>Summary 페이지</b>	<b>14</b>
4.1	자원 사용 현황 모니터링 . . . . .	14
4.2	비밀번호 변경 . . . . .	16
4.3	SSH Keypair 변경 . . . . .	17
<b>5</b>	<b>연산 세션 조회하기</b>	<b>19</b>
<b>6</b>	<b>연산 세션 실행하기</b>	<b>20</b>
6.1	새로운 세션 시작하기 . . . . .	20

6.2	이미 실행 중인 세션을 사용하고 관리하기 . . . . .	23
6.3	웹 터미널 고급 사용법 . . . . .	26
<b>7</b>	<b>폴더 다루기</b>	<b>29</b>
7.1	폴더 생성 . . . . .	30
7.2	폴더 내용 조회하기 . . . . .	31
7.3	폴더 이름 변경 . . . . .	32
7.4	폴더 삭제하기 . . . . .	32
7.5	자동 마운트 폴더 . . . . .	33
<b>8</b>	<b>폴더를 마운트 하여 연산 세션 생성</b>	<b>34</b>
8.1	자동 마운트 폴더를 이용하여 연산 세션 환경 구성하기 . . . . .	36
<b>9</b>	<b>연산 세션에 SSH/SFTP 접속하기</b>	<b>39</b>
9.1	연산 세션에 SSH/SFTP 접속하기 (Linux/Mac) . . . . .	39
9.2	연산 세션에 SSH/SFTP 접속하기 (Windows / FileZilla) . . . . .	42
<b>10</b>	<b>관리자 기능</b>	<b>45</b>
10.1	새로운 사용자 생성 및 정보 갱신 . . . . .	46
10.2	사용자 계정 비활성화 . . . . .	48
10.3	사용자의 키페어 관리 . . . . .	49
10.4	리소스 정책 관리 . . . . .	51
10.5	이미지 관리 . . . . .	54
10.6	도커 레지스트리 관리 . . . . .	56
10.7	리소스 프리셋 설정 . . . . .	58
10.8	Agent 노드 조회 . . . . .	60
10.9	리소스 그룹 관리 . . . . .	61
10.10	시스템 설정 조회 . . . . .	62
10.11	서버 관리 메뉴 . . . . .	62
<b>11</b>	<b>GUI 콘솔 사용시 문제 해결</b>	<b>64</b>
11.1	SFTP 연결이 끊기는 경우 . . . . .	64

11.2 세션이 실제 사용중인 자원과 표시되는 자원이 불일치하는 경우 . . . . .	65
11.3 도커 레지스트리에 이미지 등록 후 세션 생성 환경에 이미지가 보이지 않을 때 . . . . .	65
<b>12 사용권 계약 조건</b>	<b>67</b>
12.1 Backend.AI 라이선스 (소프트웨어) . . . . .	67
<b>13 참고 자료</b>	<b>70</b>



Backend.AI GUI Console의 사용자 가이드입니다.

Backend.AI GUI Console은 웹 또는 앱 형태의 GUI 클라이언트로, Backend.AI 서버에 접속하여 연산 자원을 활용하고 자신의 환경을 관리할 수 있는 편리한 그래픽 인터페이스를 제공합니다.

이 문서의 최신 버전은 아래 사이트에서 볼 수 있습니다:

- <https://console.docs.backend.ai/en/stable> (영문)
- <https://console.docs.backend.ai/ko/stable> (한글)

## CHAPTER 0

### 면책조항

본 설명서의 정보와 내용은 정보 전달 및 참고 목적으로만 제공되며, 상업성, 특정 목적에의 적합성 그리고 비침해성에 대한 묵시적 보증을 포함하되 이에 국한되지 않고 명시적이거나 묵시적으로든 어떠한 종류의 보증도 없이 "있는 그대로" 제공됩니다. Lablup Inc.는 결과적 손해를 포함하여 본 설명서를 사용함으로써 인해 발생할 수 있는 어떠한 종류의 손해에 대해서도 책임을 지지 않습니다. 제공되는 데이터의 정확성을 보장하기 위해 많은 노력을 기울였음에도 불구하고 Lablup Inc. 는 정보의 완벽성, 정확성, 최신성 또는 적절성에 대해 어떠한 주장, 약속 또는 보증도 하지 않으며 오자, 시대에 뒤떨어진 정보 또는 오류에 대해서도 책임을 지지 않습니다. Lablup Inc. 는 본 설명서에 수록된 정보의 정확성이나 완벽성에 대해, 명시적이거나 묵시적으로든 보증을 하지 않으며 법적 책임을 지지 않습니다.

본 설명서에 사실과 다른 오류가 있다고 생각하신다면 연락해 주십시오. 최대한 빨리 검토하도록 하겠습니다.

이 사용 설명서 내용 속 제품은 Lablup Inc. 및 오픈소스 커뮤니티의 지속적인 개발을 이유로 사전 고지 없이 변경될 수 있습니다.

여기에 나와 있는 모든 제품 및 회사 이름은 해당 소유자의 상표일 수 있습니다.

Backend.AI는 Lablup Inc.의 등록 상표입니다.

Backend.AI™에 대한 저작권은 Lablup Inc.에서 소유하고 있습니다.

©2015-2020 Lablup Inc. All rights reserved.

문서 버전: v19.09.200912

최종 갱신 일자 (년/월/일): 2020/09/12

# CHAPTER 1

---

## 개요

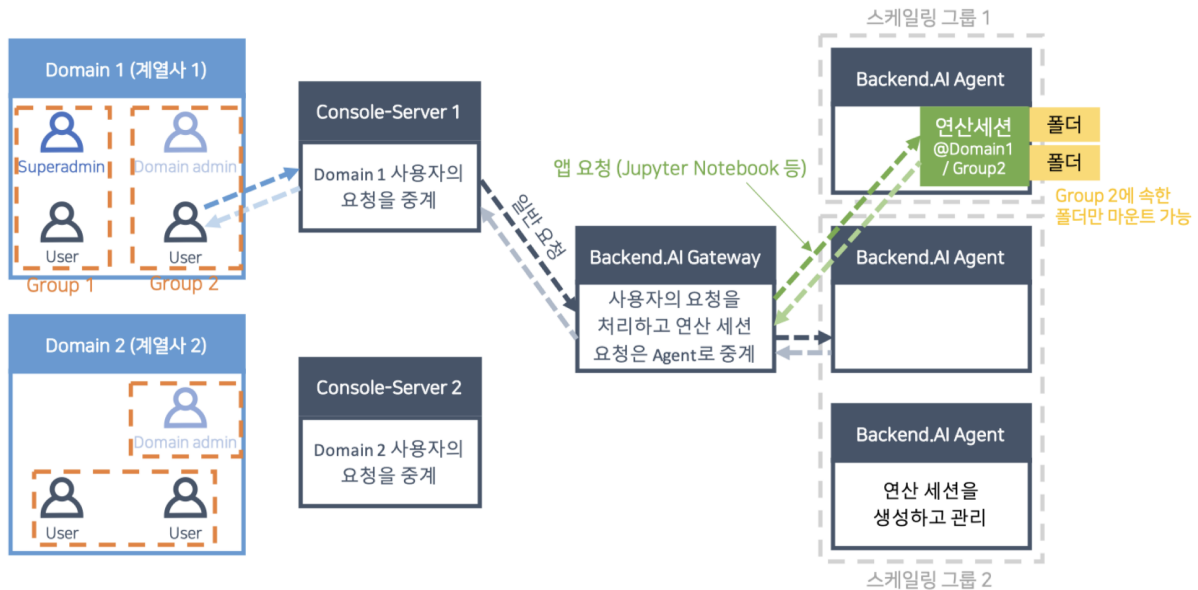
---

Backend.AI는 오픈소스 클라우드 리소스 관리 플랫폼입니다. Backend.AI는 클라우드 또는 온-프레미스 클러스터 환경에서 연산 자원을 효율적으로 관리하며 언제 어디서나 on-demand로 계산에 필요한 가상화된 연산 환경을 제공합니다. GPU 분할 가상화를 제공하여, 과학자, 데브옵스, 기업 및 AI 애호가들이 효율적으로 규모를 확장할 수 있게 돕습니다.

Backend.AI는 머신러닝 및 고성능 컴퓨팅 클러스터에 적합한 성능 향상 위주의 다양한 최적화들과 함께, 연구원, 관리자 및 데브옵스등 다양한 사용자 층을 지원하는 관리 기능 및 연구 편의 기능을 제공하고 있습니다. 엔터프라이즈 버전은 멀티 도메인 관리 및 슈퍼어드민 전용 허브 앱, GPU 분할 가상화 플러그인 기능을 추가적으로 지원합니다.

Backend.AI 서버가 지원하는 기능을 손쉽게 활용할 수 있도록 GUI 클라이언트 패키지도 함께 제공하고 있습니다. Backend.AI Console은 웹 또는 앱 형태의 GUI 클라이언트로, Backend.AI 서버에 접속하여 연산 자원을 활용하고 자신의 환경을 관리할 수 있는 편리한 그래픽 인터페이스를 제공합니다. 대부분의 작업을 마우스 클릭과 짧은 타이핑으로 할 수 있어 보다 직관적으로 사용할 수 있습니다.

## 1.1 주요 개념



- **사용자:** 사용자는 Backend.AI에 접속하여 작업을 수행하는 주체입니다. 사용자는 가지는 권한에 따라 일반 사용자, 도메인 어드민(domain admin), 수퍼 어드민(super-admin)으로 구분됩니다. 일반 사용자는 자신의 연산 세션과 관련된 작업만을 수행할 수 있는 데 반해, 도메인 어드민은 한 도메인 내에 속하는 작업을 수행할 수 있는 권한을 가지고 있으며, 수퍼 어드민은 시스템 전체에 걸쳐 거의 대부분의 작업을 수행할 수 있습니다. 사용자는 하나의 도메인에 속하게 되며, 도메인 내의 여러 그룹에 동시 속할 수 있습니다.
- **연산 세션, 컨테이너:** 자신의 연산 코드가 실행되는 격리된 가상 환경입니다. 자신이 온전한 사용자 권한을 가진 실제 리눅스 서버처럼 보이며, 물리적으로는 다른 사용자의 코드와 같은 서버에서 실행되더라도 다른 사용자를 서로 볼 수 없습니다. Backend.AI는 이러한 가상 환경을 컨테이너(container)라는 기술을 통해 구현합니다. 사용자는 자신이 속한 도메인과 그룹 내에서만 연산 세션을 생성할 수 있습니다.
- **도메인:** Backend.AI에서 지원하는 권한 및 자원 제어를 위한 최상위 계층입니다. 회사나 조직의 경우, 도메인을 하나의 계열사로 보고 도메인 별 (또는 계열사 별) 권한 및 자원 정책 등을 설정할 수 있습니다. 사용자는 반드시 하나의 도메인에 속하며, 자신의 도메인에서만 세션을 생성하거나 관련 정보를 확인할 수 있습니다. 도메인에는

하나 이상의 도메인 관리자가 있을 수 있으며, 도메인 관리자는 도메인 내의 정책을 설정하거나 세션을 관리할 수 있습니다. 예를 들어, 도메인 내에서 사용할 수 있는 총 자원량을 설정할 경우, 도메인 내 사용자가 생성한 모든 컨테이너의 자원은 설정된 양보다 클 수 없습니다.

- **그룹:** 도메인 하위에 속하는 계층으로, 한 도메인에는 여러 개의 그룹이 존재할 수 있습니다. 그룹은 하나의 프로젝트 단위라고 생각할 수 있습니다. 사용자는 한 도메인 내에 존재하는 여러 개의 그룹(또는 프로젝트)에 동시에 속할 수 있습니다. 연산 세션은 반드시 하나의 그룹에 속해야 하며, 사용자는 자신이 속한 그룹 내에서만 세션을 생성할 수 있습니다. 도메인 관리자는 도메인 내에 속한 그룹의 정책을 설정하거나 세션을 관리할 수 있습니다. 예를 들어, 그룹 내에서 사용할 수 있는 총 자원량을 설정할 경우, 그룹 내 사용자가 생성한 모든 컨테이너의 자원은 설정된 양보다 클 수 없습니다.
- **이미지:** 각 컨테이너에는 미리 설치된 언어별 런타임과 각종 연산 프레임워크들이 설치되어 있는데, 실행되기 전의 그러한 스냅샷 상태를 이미지(image)라고 합니다. 사용자는 클러스터 관리자가 제공하는 이미지를 골라서 실행하거나, 자신이 추가로 사용하고 싶은 소프트웨어가 설치된 이미지를 직접 만들거나 관리자에게 생성 또는 등록을 요청할 수 있습니다.
- **Virtual Folder (vfolder; 가상 폴더):** 사용자 별로 자신의 컨테이너가 어느 노드에서 실행 되든 관계없이 항상 접근 및 컨테이너에 탑재 가능한 "클라우드" 폴더입니다. 자신만의 가상 폴더를 생성한 후 미리 자신의 프로그램 코드, 데이터 등을 업로드해두고 연산 세션 실행 시 탑재하여 로컬 디스크에 있는 파일인 것처럼 읽고 쓸 수 있으며, 다른 사용자와의 공유도 가능합니다.
- **애플리케이션 서비스, 서비스 포트(service port):** 연산 세션 내에서 실행되는 다양한 사용자 애플리케이션(예: DIGITS, Jupyter Notebook, shell 터미널, TensorBoard 등)에 접속할 수 있게 해주는 기능입니다. 사용자는 컨테이너의 주소 및 포트 번호를 직접 알 필요 없이, 제공되는 CLI client 또는 GUI console을 이용하여 원하는 세션의 데몬에 바로 접속할 수 있습니다.
- **콘솔 앱:** 웹(web) 또는 앱(app) 형태로 서비스 되는 GUI 클라이언트. Backend.AI

서버의 주소를 지정하고 사용자 계정 정보를 입력해 로그인 한 후 서비스를 사용할 수 있습니다.

- 로컬 wsproxy: 콘솔 앱에서 내장하고 있는 프록시 서버. 콘솔 앱에서 사용 가능한 Jupyter Notebook, Terminal 등의 서비스는 websocket을 통해 서버와 통신하는데, 콘솔 앱과 주고받는 일반적인 HTTP 요청을 websocket 형태로 변환하여 메시지를 전달해 주는 역할을 합니다.
  - 콘솔 앱과 wsproxy와의 연결이 끊기거나 wsproxy 서버가 죽은 경우 Notebook, Terminal 등의 서비스 접속이 불가능합니다.
- 웹 wsproxy: 웹 형태로 제공되는 콘솔 앱의 경우 브라우저의 특성상 별도 내장 서버를 운영할 수 없습니다. 이 경우 wsproxy 서버를 별도의 웹 서버도 독립 운영하여 콘솔 앱에서 웹 wsproxy를 바라보게 만듦으로써 Jupyter Notebook, Terminal 등과 같은 서비스를 웹 환경에서도 사용할 수 있습니다.

## CHAPTER 2

---

### 설치

---

GUI 콘솔은 두 가지 형태로 사용할 수 있습니다. 관리자가 설치한 별도의 웹 주소로 접속하여 웹 서비스 형태로 이용할 수 있고, 별도 설치가 필요 없는 standalone 실행파일로 제공되는 앱 형태로 이용할 수도 있습니다. 앱 형태의 경우 데스크톱 운영체제의 보안 설정에 따라 서명되지 않은 실행파일로 인식되어 권한 허용 절차가 필요할 수 있습니다.



## CHAPTER 3

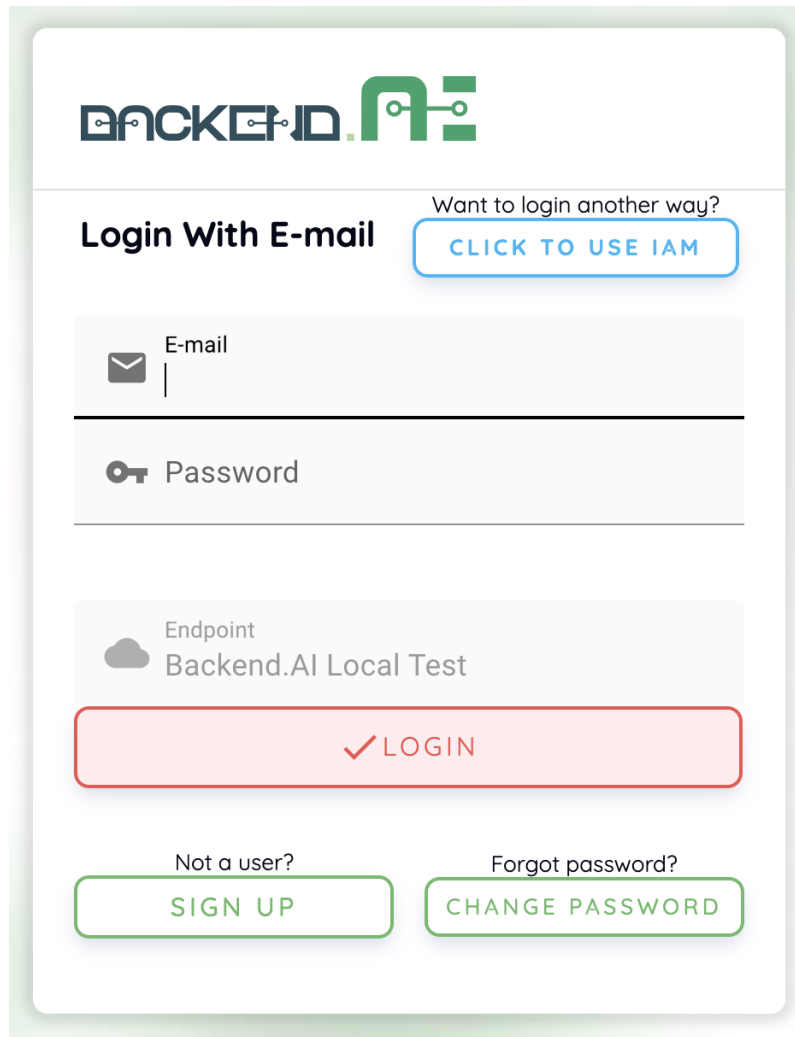
---

### 가입 및 로그인

---

#### 3.1 가입

GUI 콘솔을 실행하면 다음과 같은 로그인 대화창이 뜹니다. 아직 가입을 하지 않은 경우에는 SIGN UP 버튼을 눌러주세요.



The image shows a login form for Backend.AI. At the top is the Backend.AI logo. Below it, the text "Login With E-mail" is displayed. To the right of this text is a link that says "Want to login another way?" with a button labeled "CLICK TO USE IAM". Below the "Login With E-mail" text are two input fields: "E-mail" with an envelope icon and "Password" with a key icon. Below these fields is a section labeled "Endpoint" with a cloud icon and the text "Backend.AI Local Test". Below this section is a large red button with a checkmark and the text "LOGIN". At the bottom of the form are two buttons: "SIGN UP" with the text "Not a user?" above it, and "CHANGE PASSWORD" with the text "Forgot password?" above it.

이메일과 사용자 이름, 비밀번호 등의 정보를 입력하고 약관과 개인정보보호정책을 읽고 동의한 뒤 SIGNUP 버튼을 클릭합니다. 시스템 설정에 따라 가입하기 위해 별도의 초대 토큰을 입력해야할 수도 있습니다. 또한, 이메일이 본인의 것이 맞는지 검증하는 이메일이 전송될 수도 있습니다. 검증 이메일이 전송되는 경우, 이메일을 읽고 확인 링크를 클릭해서 검증을 통과해야만 가입한 계정으로 로그인 할 수 있습니다.

Signup (Beta invitation only) X

E-mail  
tester@test.com

UserName  
tester

Invitation Token  
invitation-token

Password  
.....

Password (again)  
.....

☒ I have read and agree to the [Terms of service](#) and [Privacy Policy](#).

✓ SIGNUP

**주석:** 서버 설정이나 플러그인 설정에 따라 사용자에게 의한 가입이 막혀있을 수 있습니다. 이 경우에는 시스템 관리자에게 문의 하십시오.

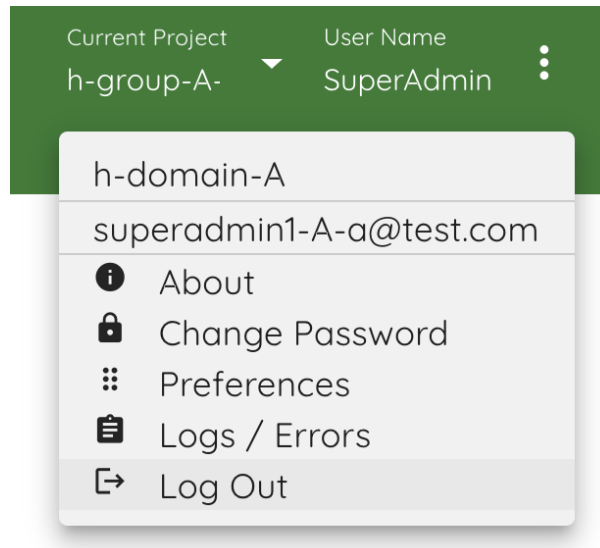
## 3.2 로그인

ID와 암호를 입력하고 LOGIN 버튼을 눌러주세요. ENDPOINT에는 Manager로의 연결을 중계하는 Backend.AI Console-Server가 작동하고 있는 URL을 넣어주시면 됩니다.

**주석:** Console server의 설치 및 설정 환경에 따라, Endpoint가 고정되어 있을 수 있습니다.

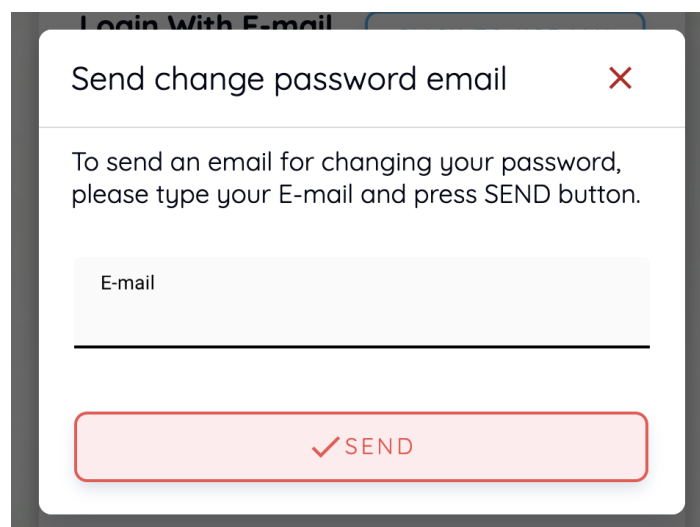
로그인이 완료되면 Summary 탭에서 현재 사용하고 있는 자원량 등의 정보를 확인할 수 있습니다.

우측 상단의 아이콘을 클릭하면 하위 메뉴가 나옵니다. Log Out 버튼을 클릭해서 로그아웃할 수 있습니다.



### 3.3 비밀번호를 잊어버렸을 경우

비밀번호를 잊어버렸을 경우, 로그인 패널의 CHANGE PASSWORD 버튼을 클릭하여 비밀번호 변경 링크를 이메일로 전송할 수 있습니다. 이메일을 읽고 지시대로 따라하면 비밀번호를 변경할 수 있습니다. 서버 설정에 따라 비밀번호 변경 기능이 비활성화 되어있을 수 있습니다. 이 경우에는 관리자에게 문의 하십시오.



---

**주석:** 이 기능도 모듈화되어 있어 어떤 환경에서는 지원되지 않을 수 있습니다.

---

## CHAPTER 4

---

### Summary 페이지

---

Summary 페이지에서는 사용자의 자원 및 세션 사용 현황을 확인할 수 있으며, 자주 사용되는 기능의 단축 링크가 있습니다.

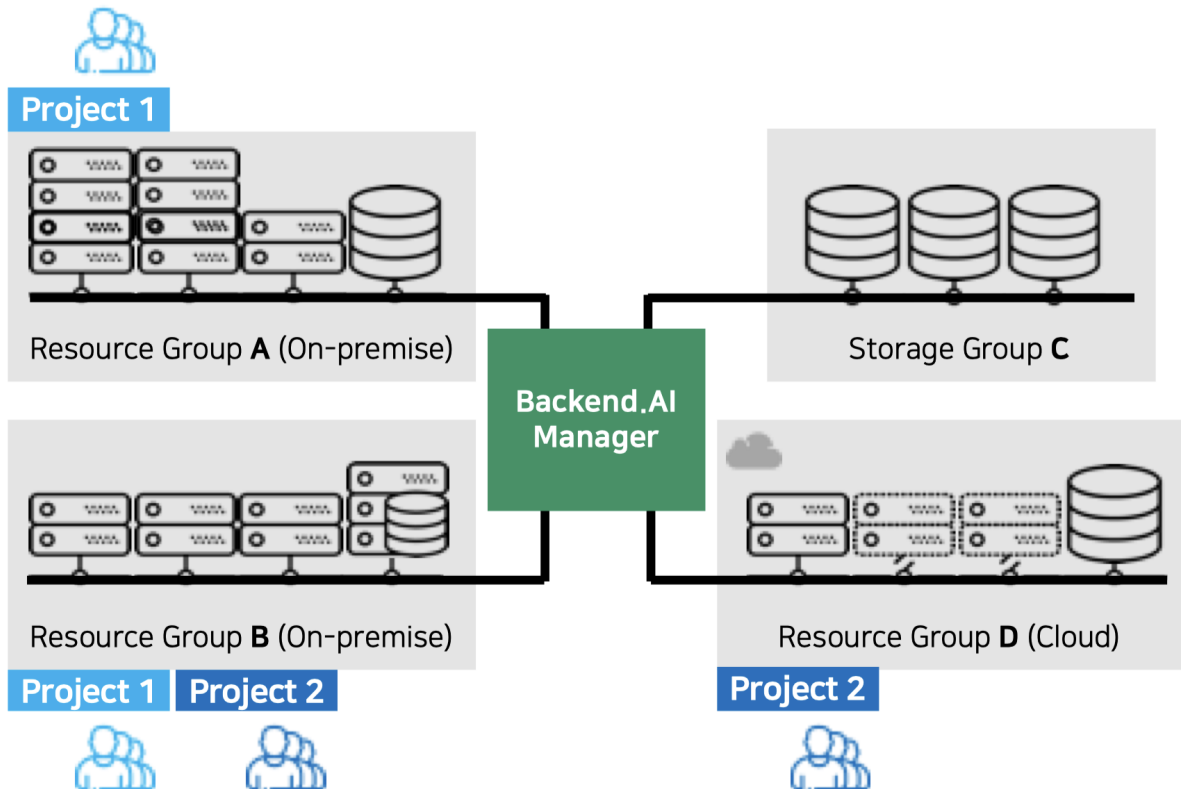
#### 4.1 자원 사용 현황 모니터링

Start menu에서는 사용자가 할당할 수 있는 자원의 총량과 현재 사용하고 있는 양을 확인할 수 있습니다. CPU, RAM, GPU 점유 현황 및 현재 실행 중인 세션의 개수 등을 확인할 수 있으며, +START 버튼을 클릭해서 세션의 바로 생성해볼 수 있습니다 (세션 생성 방법은 Sessions 페이지를 참조하시기 바랍니다).

사용자는 하나 이상의 그룹에 속할 수 있습니다. 사용하고자 하는 그룹을 변경하려면 우측 상단에 있는 Project 버튼을 클릭하여 원하는 그룹을 선택할 수 있습니다. 그룹이 바뀌게 되면 그룹에 설정된 자원 사용 정책에 따라 사용 가능한 자원의 총량도 바뀌게 되고, 현재 사용하고 있는 자원의 양도 해당 그룹 내에서 돌아가고 있는 세션의 점유 자원량으로 업데이트 됩니다. 사용자가 속하는 그룹은 관리자가 설정하는 부분으로, 그룹을 변경하고

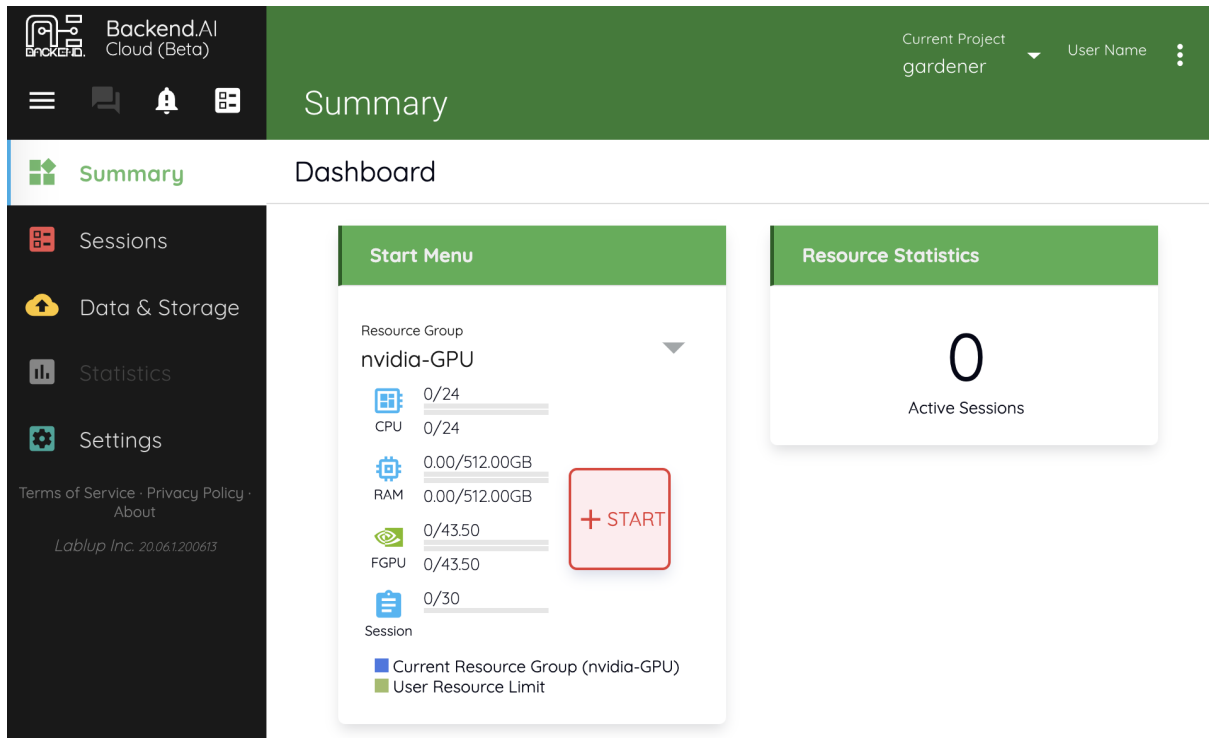
싶다면 담당 관리자에게 문의해야 합니다.

사용자는 관리자에 의해 여러 개의 리소스 그룹을 사용하도록 설정될 수 있습니다. 리소스 그룹은 사용자, 그룹 또는 도메인 별로 접근 가능한 워커 노드를 설정할 수 있는 관리자 기능입니다. 예를 들어, V100 GPU 서버 여러 대를 하나의 리소스 그룹으로 묶고, P100 GPU 서버 여러 대를 또 다른 리소스 그룹으로 묶은 후 특정 사용자에게 P100 리소스 그룹만 접근할 수 있도록 설정하면, 해당 사용자는 P100 GPU 서버의 자원만 사용할 수 있고 V100 GPU 서버의 자원은 사용할 수 없습니다. Start Menu 내의 Resource Group을 클릭하여 자신이 접근 가능한 리소스 그룹을 선택할 수 있습니다. 리소스 그룹이 변경될 때마다 가용한 자원량도 따라서 바뀌게 됩니다. 리소스 그룹 정책은 관리자가 설정하는 부분으로, 특정 자원을 사용하고 싶다면 담당 관리자에게 문의해야 합니다.



하단의 Project 섹션에서는 사용자가 속한 그룹 내의 총 사용량 및 자신의 해당 그룹 내 사용량을 확인할 수 있습니다.

Resource Statistics 패널에서는 현재 사용자가 실행하고 있는 총 세션의 수를 보여줍니다. 사용자의 권한에 따라 기타 세션 관련 정보가 추가로 출력됩니다.

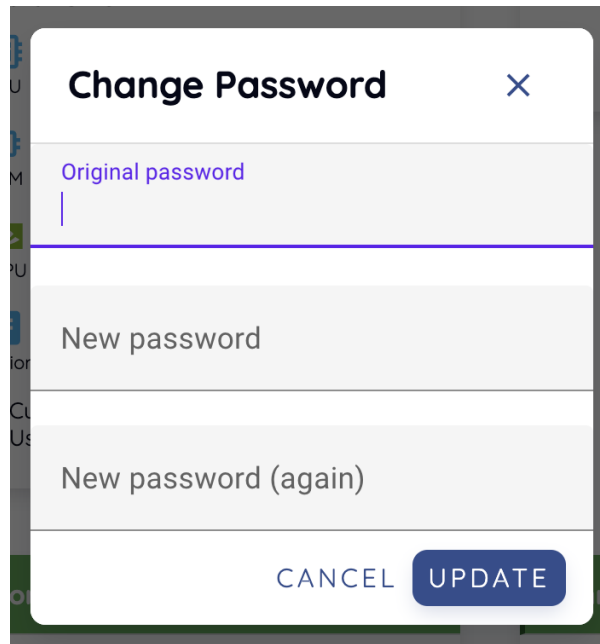


## 4.2 비밀번호 변경

우측 상단의 설정 버튼을 클릭하면 나오는 Change Password 메뉴를 클릭해서 비밀번호를 변경할 수 있습니다. Original password 필드에 현재 비밀번호를, New password와 New password (again) 필드에 새로 설정할 비밀번호를 입력한 후 UPDATE 버튼을 눌러 비밀번호를 갱신합니다. 이 때, 비밀번호는 다음과 같은 규칙을 따라야만 합니다:

- 8 글자 이상
- 하나 이상의 알파벳
- 하나 이상의 특수문자
- 하나 이상의 숫자



A modal dialog box titled "Change Password" with a close button (X) in the top right corner. It contains three input fields: "Original password" (with a purple label), "New password", and "New password (again)". At the bottom, there are two buttons: "CANCEL" and "UPDATE".

Change Password

Original password

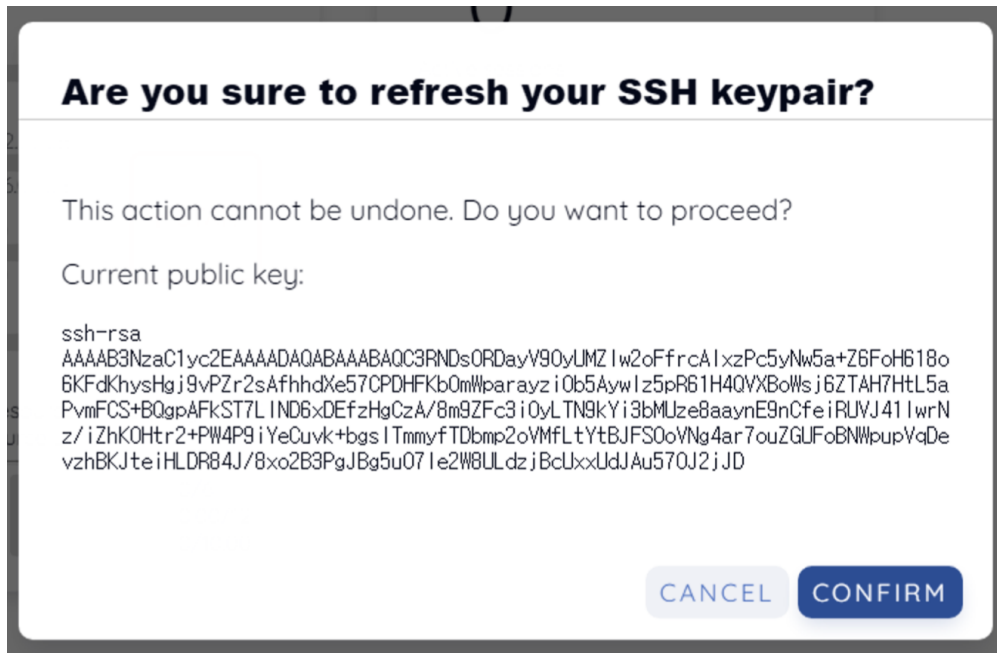
New password

New password (again)

CANCEL UPDATE

### 4.3 SSH Keypair 변경

앱 형태의 GUI 콘솔을 사용하는 경우 컨테이너로 직접 SSH/SFTP 접속을 할 수 있습니다. (주의: 웹 기반 콘솔에서는 아직 SSH/SFTP 접속을 지원하지 않습니다.) 이 때 사용되는 SSH keypair를 조회하거나 새로 생성할 수 있습니다. Preferences 패널의 Refresh SSH Keypair 링크를 클릭하면 다음과 같은 다이얼로그가 뜹니다. 현재 설정된 public key를 조회할 수 있으며 (private key는 조회되지 않습니다), CONFIRM 버튼을 클릭하여 새로운 public key, private key를 생성할 수 있습니다.



**주석:** 웹기반 콘솔은 아직 SSH/SFTP 연결을 지원하지 않습니다.

**주석:** Backend.AI는 OpenSSH에 기반한 SSH keypair를 사용합니다. 윈도우즈에서는 PPK 기반 키로 변환해야 할 수 있습니다.

# CHAPTER 5

## 연산 세션 조회하기

연산 세션을 조회하려면 왼쪽 사이드바의 Sessions를 클릭합니다. 우측 내용의 Running 탭에는 현재 실행되고 있는 세션의 정보를 확인할 수 있습니다. Finished 탭을 클릭하면 과거에 실행되었다가 종료된 세션의 정보를 볼 수 있습니다. 각 세션 별로 ID, 시작일, 사용 시간, 자원 설정, 사용량 등의 정보를 확인할 수 있습니다.

Sessions

TOTAL RESOURCE MY

CPU

4/24

RAM

8.00/512.00GB

FGPU

2/43.50

Session

1/30

Current Project

gardener

User Name

Running

Finished

Others

+ START

#	SessionInfo	Status	Control	Configuration	Usage	Starts
1	CC1Wp2j-console TensorFlow 2.2 PY36 CUDA10.1	RUNNING		nvidia-GPU <sup>RG</sup> 4core  8.00GB 2.00GPU	No data 0MB  0MB READ  WRITE	2020. 6. 13. 오후 10

수퍼어드민의 경우 현재 클러스터에서 실행 중인 (또는 종료된) 모든 세션 정보를 확인할 수 있고, 사용자의 경우에는 자신이 사용한 세션만 조회 가능합니다.

## CHAPTER 6

---

### 연산 세션 실행하기

---

Sessions 탭에서는 연산 세션을 조회하는 것뿐만 아니라 새로운 세션을 시작하거나 실행 중인 세션을 사용 및 관리할 수도 있습니다.

#### 6.1 새로운 세션 시작하기

새로운 연산 세션을 시작하려면 START 버튼을 누르세요. 다음과 같은 설정 창이 뜨는데, 사용하려는 언어 환경(Environments, Version)과 자원 설정(원하는 CPU, RAM, GPU 등 할당량)을 지정하고 LAUNCH 버튼을 누릅니다.

**Start new session** [X]

Environments\*  
TensorFlow

Version\*  
2.2-py36-cuda10.1

Resource Group\*  
nvidia-GPU

Session name (optional)

Folder to mount

**Resource allocation**

SMALL  
2 CPU  
4GB RAM

**LARGE**  
4 CPU  
8GB RAM  
1GB SHRAM

MEDIUM  
2 CPU  
4GB RAM  
1GB SHRAM

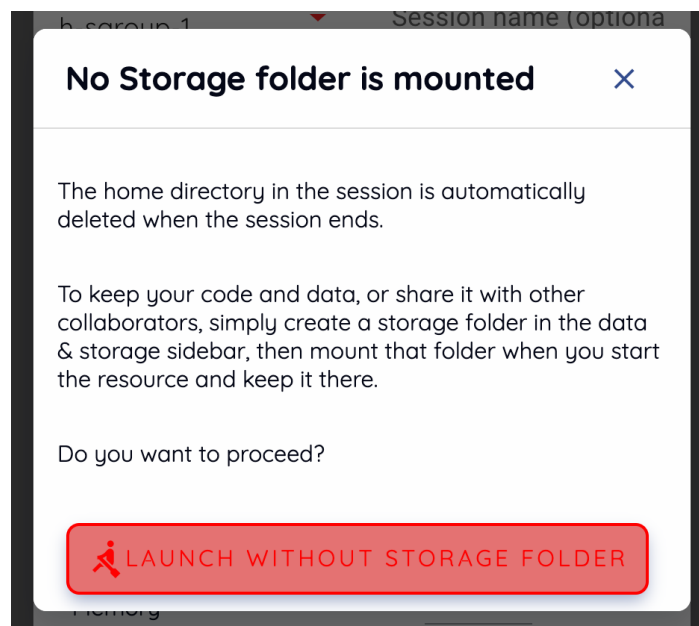
Advanced Custom allocation

**LAUNCH**

**주석:** 만약 GPU 자원이 FGPU로 표시되어 있다면, 이는 서버가 GPU 자원을 가상화된 형태로 제공하고 있다는 것을 의미합니다. Backend.AI는 하나의 물리 GPU를 여러 개로 분할해서 여러 사용자가 나누어 사용할 수 있는 가상화 기술을 지원하고 있습니다. 따라서, GPU 연산 소요가 크지 않은 작업을 수행하고자 할 경우에는 GPU의 일부만 할당하여 연산 세션을 생성할 수 있습니다. 1 FGPU가 실제로 할당하는 GPU 자원의 양은 관리자 설정에 따라 시스템 별로 다양할 수 있습니다. 예를 들어, 관리자가 하나의 GPU를 다섯 조각으로 분할 설정한 경우, 5 FGPU가 1 물리 GPU, 또는 1 FGPU가 0.2 물리 GPU를 뜻합니다. 이 때 1 FGPU를 설정하여 연산 세션을 생성하면, 그 세션에서는 0.2 물리 GPU에 해당하는

SM(streaming multiprocessor)과 GPU 메모리를 활용할 수 있습니다.

이 때, “Folder to mount”에 아무 마운트 폴더도 지정하지 않은 경우 다음과 같은 알람 다이얼로그가 뜰 수 있습니다. 연산 세션을 삭제하면 기본적으로 세션 내의 모든 데이터도 같이 삭제되므로 저장 폴더를 하나 이상 마운트 하는 것을 권장한다는 내용입니다. 마운트 폴더를 지정한 후 그 폴더에 데이터를 보관하면 연산 세션이 삭제되더라도 데이터를 계속 보존할 수 있습니다. 저장 폴더에 보존된 데이터는 다른 세션을 생성할 때 다시 마운트 하여 데이터를 재사용할 수도 있습니다. 알람을 무시하고 연산 세션을 생성해도 상관없습니다. 하지만, 세션 내에서 데이터를 보관해야 할 필요가 있는 작업을 하는 경우라면 꼭 폴더를 마운트 하는 것이 좋습니다. 폴더를 마운트 하여 연산 세션을 실행하는 법에 관해서는 [관련 내용](#) 을 참고 하십시오.

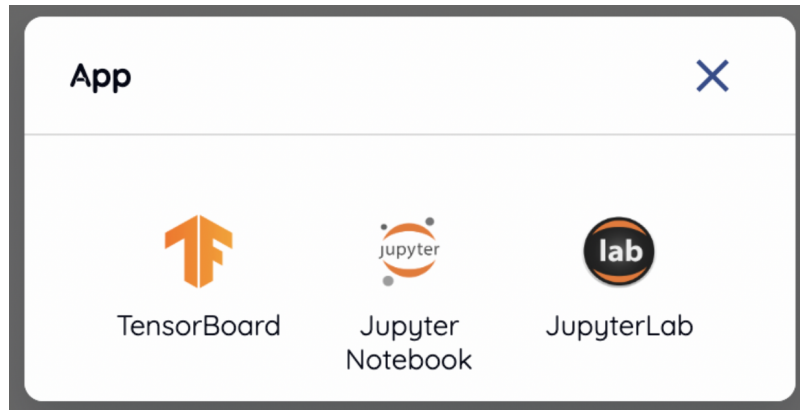


Running 탭에서 새로운 연산 세션이 생성되는 것을 확인합니다.

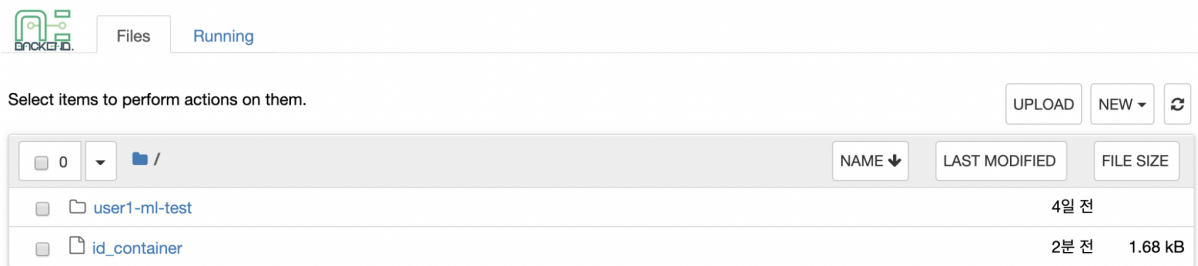
Running		Finished		Others				+ START
	#	Session Info	Status	Control	Configuration	Usage	Starts	
<input type="radio"/>	1	dbYxwVPa-console TensorFlow 19.11 Nvidia GPU Cloud TF2 PY3	RUNNING		h-sgroup-1SG 2core 4.00GB 1.00GPU	No data OMB OMB READ WRITE	2020. 2. 4. 오후 2	
<input type="radio"/>	2	2M9V9lq3-console TensorFlow 19.11 Nvidia GPU Cloud TF2 PY3	RUNNING		h-sgroup-1SG 1core 2.00GB 0.30GPU	No data OMB OMB READ WRITE	2020. 2. 4. 오후 2	

## 6.2 이미 실행 중인 세션을 사용하고 관리하기

이번에는 이미 실행 중인 연산 세션을 사용하고 관리하는 법에 대해 살펴보겠습니다. 세션 리스트의 Control 열을 보면 몇 가지 아이콘이 있습니다. 플레이 아이콘을 누르면 다음 그림과 같이 해당 세션이 지원하는 몇 가지 앱 서비스가 뜨게 됩니다.



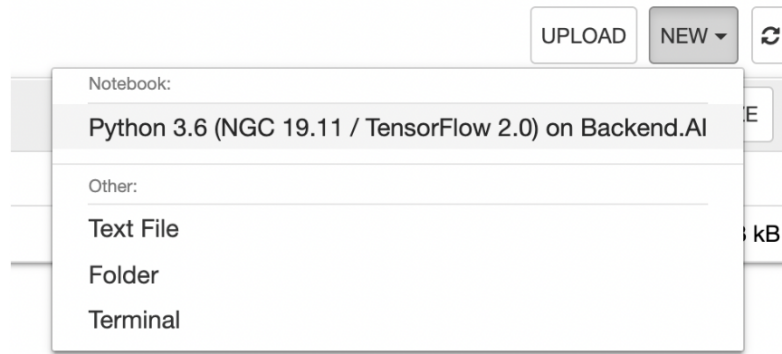
시험 삼아 Jupyter Notebook을 눌러봅시다.



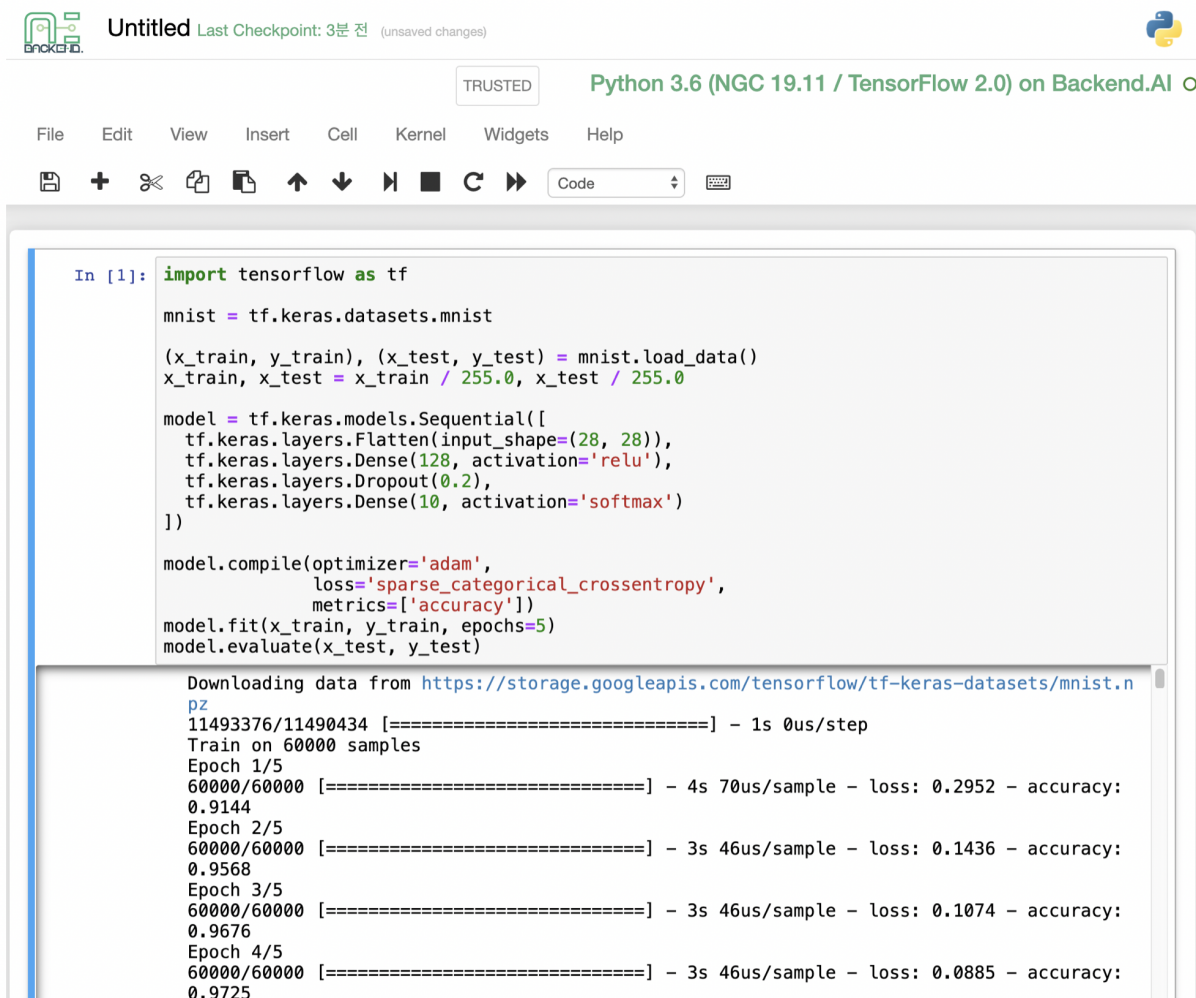
새로운 창이 뜨면서 Jupyter Notebook이 실행되는 것을 확인할 수 있습니다. 이 Notebook은 실행 중인 연산 세션 내부에서 생성된 것으로, 별다른 설정 없이 버튼 클릭만으로 손쉽게 사용할 수 있습니다. 또한, 연산 세션이 기본적으로 제공하는 언어 환경 및 라이브러리를 그대로 활용할 수 있어 별도의 패키지 설치 과정이 필요 없습니다. 자세한 Jupyter Notebook 사용법은 공식 문서 등을 참고하시기 바랍니다.

Notebook의 파일 탐색기에서 `id_container` 파일은 private SSH key를 담고 있습니다. 필요할 경우 다운로드 하여 컨테이너로의 SSH/SFTP 접속에 이용할 수 있습니다.

우측 상단의 NEW 버튼을 클릭한 후 Backend.AI 용 Notebook을 선택하면 새로운 코드를 입력할 수 있는 ipynb 창이 뜹니다.



이 창에서 세션 환경에 맞는 코드를 입력하고 실행해볼 수 있습니다. 코드는 Backend.AI 서버를 구성하는 노드 중 연산 세션이 실제로 생성된 노드에서 실행이 되며, 로컬 머신에는 별도 환경을 구성할 필요가 없습니다.



창을 닫으면 Notebook 파일 탐색기에 방금 작업한 Untitled.ipynb 파일이 생성되어 있는 것을 확인할 수 있습니다. 여기 생성된 파일은 세션을 삭제할 경우 같이 삭제되는 것에 주의하십시오. 생성된 파일을 세션이 사라지더라도 보존하는 방법은 폴더 섹션에서 설명합니다.





다시 세션 리스트 페이지로 돌아옵니다. 이번에는 터미널을 띄워보도록 하겠습니다. 플레이 버튼 우측의 터미널 아이콘을 누르면 해당 컨테이너의 터미널을 사용할 수 있습니다. 터미널 역시 새 창에서 뜨게 되며, 다음 그림처럼 연산 세션 내부에 접속해서 커맨드 명령을 내릴 수 있습니다. 커맨드 사용에 익숙한 분이라면 손쉽게 여러 가지 리눅스 명령을 내릴 수가 있습니다.

```
work@b0483c80121:~$ pwd
/home/work
work@b0483c80121:~$ ipython
Python 3.6.8 (default, Oct 7 2019, 12:59:55)
Type 'copyright', 'credits' or 'license' for more information
IPython 7.9.0 -- An enhanced Interactive Python. Type '?' for help.

In [1]: import tensorflow as tf
2020-02-04 06:00:54.770009: I tensorflow/stream_executor/platform/d
udart.so.10.2
t
In [2]: tf.__version__
Out[2]: '2.0.0'

In [3]: exit
lwork@b0483c80121:~$ ll
total 308
drwxr-xr-x 10 work work    209 Feb  4 05:44 .
drwxr-xr-x  1 root root    32 Feb  4 05:33 ..
-rw-r--r--  1 work work    98 Dec 25 09:29 .bashrc
drwxr-xr-x  2 work work    39 Feb  4 05:40 .ipynb_checkpoints
drwxr-xr-x  5 work work    67 Feb  4 05:33 .ipython
drwxr-xr-x  3 work work    36 Feb  4 05:35 .jupyter
drwxr-xr-x  3 work work    40 Feb  4 05:42 .keras
drwxr-xr-x  4 work work    30 Feb  4 05:35 .local
drwxrwxr-x  3 work work    18 Nov 13 02:21 .logs
drwx----- 2 work work    29 Feb  4 05:33 .ssh
-rw-r--r--  1 work work   501 Dec 25 09:29 .vimrc
-rw-r--r--  1 work work 302666 Feb  4 05:44 Untitled.ipynb
-rw-----  1 work work  1675 Feb  4 05:33 id_container
drwxrwxr-x  4 work work    55 Jan 31 04:26 user1-ml-test
work@b0483c80121:~$
```

만약 여기서 파일을 만들면 앞서 띄운 Jupyter Notebook에서 즉시 그 파일을 확인할 수 있습니다. 반대로, Jupyter Notebook에서 편집한 파일의 변경 사항도 터미널에서 바로 확인할 수 있습니다. 같은 연산 세션을 사용하고 있기 때문입니다.

이 외에도 연산 세션이 제공하는 서비스의 종류에 따라 TensorBoard, Jupyter Lab 등과 같은 웹 기반 서비스를 이용할 수 있습니다.

특정 세션을 삭제하기 위해서는 빨간색 휴지통 아이콘을 누르면 됩니다. 연산 세션 내부 폴더의 데이터는 연산 세션이 종료되는 순간 같이 삭제되므로, 계속 유지해야 하는 데이터라면 마운트 한 폴더에 옮기거나 처음부터 폴더에 업로드하는 것을 권합니다.

## 6.3 웹 터미널 고급 사용법

위에서 사용 해보았던 웹 기반 터미널은 내부적으로 **tmux** 라는 유틸리티를 사용하고 있습니다. **tmux**는 하나의 셸 내에서 여러 개의 창을 띄워 다중 작업 할 수 있도록 지원하는 **terminal multiplexer**로, 셸이 닫히더라도 작업하던 내용을 보존할 수 있는 등 다양한 장점을 가지고 있습니다. 보다 강력한 터미널 기능을 활용하고 싶다면 **tmux** 공식 문서 및 기타 인터넷 상의 다양한 사용 예제를 참고하십시오.

여기서는 몇 가지 간단하지만 유용한 기능을 소개하겠습니다.

### 터미널 내용 복사하기

**tmux**는 다양한 장점을 가지고 있지만 처음 접하는 사용자라면 다소 혼란스러운 부분도 있습니다. 특히, **tmux**는 자체 클립보드 버퍼를 가지고 있어 터미널의 내용을 복사할 경우 기본적으로 **tmux** 내에서만 붙여넣을 수 있다는 부분에서 어려움을 겪을 수 있습니다. 여기에 웹 기반 터미널에서는 사용자 시스템의 클립보드를 **tmux**에 노출시키기 어렵다는 한계가 더해져서, **tmux** 셸을 사용하고 있는 상태에서는 마우스 드래그를 통해 터미널 내용을 복사한 후 사용자 컴퓨터의 다른 프로그램에 붙여넣을 수가 없습니다. 소위 말하는 Ctrl-C / Ctrl-V 가 작동하지 않는 것입니다.

만약 터미널 내용을 사용자 시스템의 클립보드로 복사하여 붙여넣기를 할 필요가 있을 때는 잠시 **tmux**의 마우스 지원 설정을 끌 수 있습니다. 먼저 Ctrl-B 키를 눌러 **tmux**의 제어 모드로 진입합니다. 그 후 `:set -g mouse off` 를 입력하고 엔터키를 누릅니다 (처음 콜론까지 입력해야 합니다). 입력하는 내용은 **tmux** 하단의 상태바에서 확인할 수 있습니다. 그 후 터미널에서 마우스로 원하는 텍스트를 드래그 하고 Ctrl-C 또는 Cmd-C 키를 누르면 사용자

컴퓨터의 클립보드에 해당 내용이 복사 됩니다.

마우스 지원을 끈 상태에서는 마우스 휠로 스크롤하여 터미널의 이전 페이지 내용을 확인할 수 없습니다. 이 때는 다시 마우스 지원을 켜면 됩니다. Ctrl-B 를 누른 후 이번에는 :set -g mouse on 을 입력해봅시다. 이제 마우스 휠을 스크롤하여 이전 페이지의 내용을 볼 수 있게 되었습니다.

이와 같이 Ctrl-B 후 :set -g mouse off 또는 :set -g mouse on 을 기억하면 조금 더 편리하게 웹 터미널을 활용할 수 있습니다.

---

**주석:** Ctrl-B 키는 tmux의 기본 제어 모드 키입니다. 만약 홈 디렉토리의 .tmux.conf 를 수정하여 다른 제어 키를 설정한 경우에는, Ctrl-B 대신 설정된 키 조합을 눌러야 합니다.

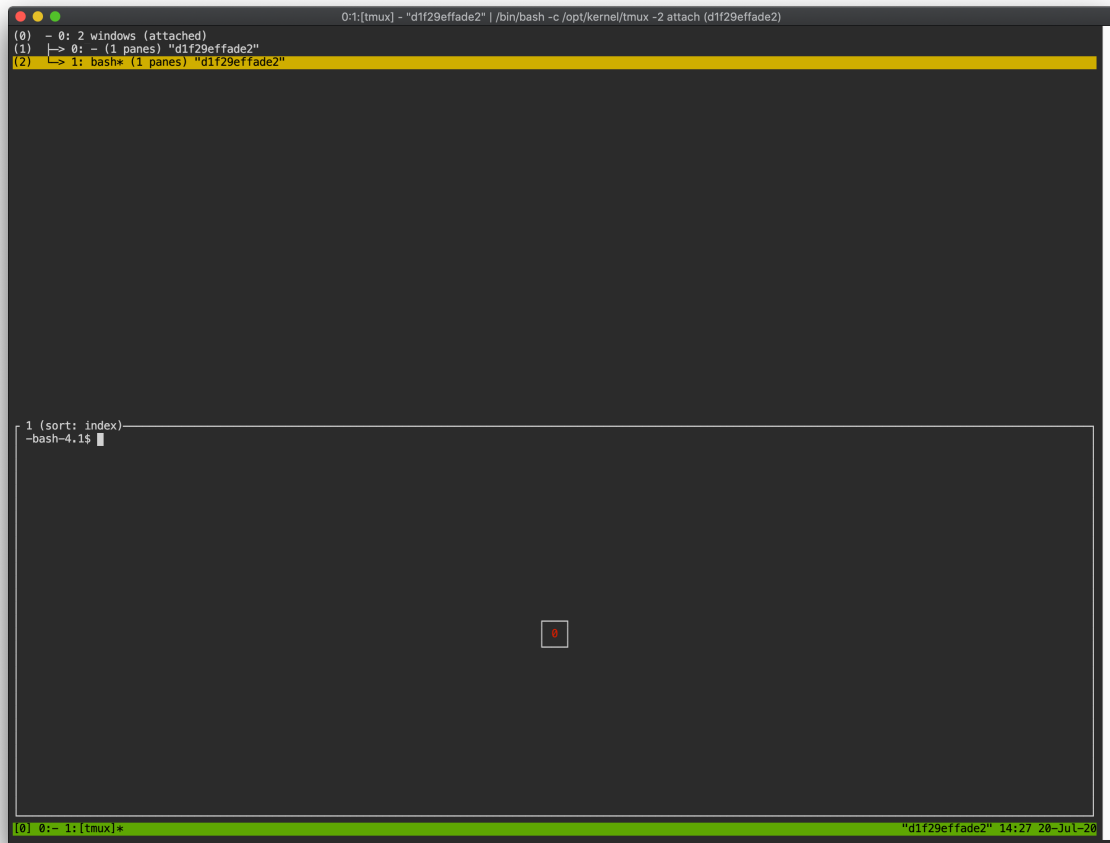
---

### 키보드를 이용해 터미널 이전 내용 확인하기

터미널 내용 복사도 하면서 터미널의 이전 내용도 확인하고 싶은 경우에도 방법이 있습니다. 바로 키보드를 이용해서 이전 내용을 확인하는 것입니다. 이번에도 Ctrl-B 를 먼저 클릭 해준 뒤 Page Up 과 Page Down 키를 눌러 봅시다. 키보드 만으로 터미널의 이전 내용을 탐색할 수 있다는 것을 확인할 수 있습니다. 탐색 모드에서 빠져 나오려면 q 키를 눌러주면 됩니다. 이 방법을 이용하면 마우스 지원을 끈 상태에서도 터미널 이전 내용 확인이 가능합니다.

### 여러 개의 셸 띄우기

tmux의 가장 큰 장점은 하나의 터미널 창에서 여러 개의 셸을 띄우고 사용할 수 있다는 점입니다. 백문이 불여일견이니 Ctrl-B 키를 누른 후 c 키를 이어서 눌러봅시다. 기존 창의 내용이 사라지고 새로운 셸 환경이 뜬 것을 확인할 수 있습니다. 그러면 기존 창은 사라진 것일까요? 그렇지 않습니다. Ctrl-B 누른 후 w 키를 눌러 봅시다. 다음과 같이 현재 tmux 상에서 열려 있는 셸 리스트가 조회되는 것을 볼 수 있습니다. 여기서 0: 으로 시작하는 셸이 처음 보던 셸 환경이고, 1: 로 시작하는 셸은 방금 새로 생성한 셸입니다. 위/아래 방향 키를 이용해서 셸 사이를 이동할 수 있습니다. 0: 번 셸에 커서를 가져다 두고 엔터 키를 눌러 선택 해보겠습니다.



처음 보았던 셸 환경이 나타나는 것을 볼 수 있습니다. 이와 같은 방식으로 웹 터미널 내에서 여러 개의 셸 환경을 띄워 두고 사용할 수 있습니다. 현재 셸을 종료하려면 `exit` 명령을 입력하거나 `Ctrl-B x` 키를 누른 후 `y` 를 입력하면 됩니다.

정리하면 다음과 같습니다:

- `Ctrl-B c`: 새로운 `tmux` 셸 생성
- `Ctrl-B w`: `tmux` 셸 조회 및 이동/선택
- `exit` 또는 `Ctrl-B x`: 현재 `tmux` 셸 종료

위 명령을 조합하여 여러 개의 셸에서 동시에 다양한 작업을 수행할 수 있습니다.

## CHAPTER 7

### 폴더 다루기

Backend.AI는 사용자의 파일을 보존할 수 있도록 별도의 저장소를 지원합니다. 연산 세션의 기본 폴더는 세션이 삭제되면 같이 삭제되므로, 보존해야 하는 파일이 있을 경우 Storage 폴더에 저장하는 것이 좋습니다. Storage 폴더는 좌측 메뉴의 Storage 탭을 선택하면 조회할 수 있습니다.

Data & Storage							
<div>☰</div> <div>Summary</div> <div>Sessions</div> <div>Data &amp; Storage</div> <div>Statistics</div> <div>Settings</div> <div>Terms of Service · Privacy Policy · About</div> <div>Lablup Inc. 2006.12.00613</div>	Folders	Automount Folders		Shared Data		Mod	+ NEW FOLDER
	#	Name	id	Location	Ty...	Permissi...	Control
	1	jpark-test	19d7feda9dfb4fb18514	L3	👤	R W D	📁 🔗 👤 ✎ 🗑
	2	devops	9ae646f6075246f4a2b7	L3	👤	R W D	📁 🔗 👤 ✎ 🗑
	3	ml	a43d6cec7a714c30b19c	L3	👤	R W D	📁 🔗 👤 ✎ 🗑
	4	new-data	d0bc3dc2076b4748808f	L3	👤	R W D	📁 🔗 👤 ✎ 🗑
	5	lablup-document	70437e5694f54829805c	L3	👤	R	📁 🔗 👤 ✎ 🗑
	6	19_covid	02e1514521cd47ebbd8f	L3	👤	R W	📁 🔗 👤 ✎ 🗑

Storage 폴더에는 두 가지 형태가 있습니다. 사용자 형태(User)의 경우 일반 사용자가 자신

의 폴더를 생성한 것으로, Type 컬럼에서 한 명의 사용자가 있는 것으로 확인할 수 있습니다. 그룹 폴더(Group)의 경우 Type 컬럼에 여러 명의 사용자가 있는 아이콘으로 알아볼 수 있으며, 도메인 관리자에 의해서만 그룹별로 생성되고 일반 사용자는 생성된 그룹폴더를 사용만 할 수 있습니다.

## 7.1 폴더 생성

NEW FOLDER 버튼을 눌러서 폴더를 원하는 이름으로 생성할 수 있습니다. Folder name에 생성할 폴더의 이름을 입력하고, Type은 User/Group 중 하나를 선택합니다. (서버 설정에 따라 User, Group 중 하나만 선택 가능할 수 있습니다.) 그룹 폴더를 생성할 경우 Group 필드를 설정해 주어야 합니다. 그룹 폴더는 Group 필드에 지정된 그룹 전용이 되며, 해당 그룹에 속해있는 사용자만 그 그룹 폴더를 마운트해서 사용할 수 있습니다. 원하는 대로 값을 설정한 후 CREATE 버튼을 클릭하여 폴더를 생성할 수 있습니다.

Create a new storage folder

Folder name \*  
user1-ml-test

Host local Type User

Group h-group-A-b

Folders starting with a .(dot) are auto session is started.

CREATE

## 7.2 폴더 내용 조회하기

Control 열의 폴더 아이콘을 클릭하여 해당 폴더의 내용을 조회할 수 있는 파일 탐색기를 띄울 수 있습니다.

test-folder	8ad730f8f421433ea35ff	local								
-------------	-----------------------	-------	--	--	--	--	--	--	--	--

내부에 폴더 및 파일이 존재하고 있는 것을 확인할 수 있습니다. Name 열에 있는 폴더 이름을 클릭하면 해당 폴더로 이동할 수 있습니다. Actions 열의 다운로드 버튼과 삭제 버튼을 클릭하여 파일을 받거나 폴더 상에서 완전히 삭제할 수 있습니다. 파일과 디렉토리의 이름을 변경하는 것도 가능합니다. 보다 섬세한 파일 작업이 필요하다면 이 폴더를 연산 세션 생성 시 마운트 한 뒤 터미널이나 Jupyter Notebook 등과 같은 서비스를 활용해서 수행할 수 있습니다.

Folders

Automount Folders

Shared Data

Mod

+ NEW FOLDER

#

Name

id

Location

Tu...

Permissi...

Control

jpark-test

X

>

📁

📁 UPLOAD FILES...

📁 NEW FOLDER

#

Name

Created

Size

Actions

1

📁

tensorflow

Fri, 10 Apr 2020 17:07:16 GMT

1728016531

☁

✎

🗑

2

📁

mlperf

Fri, 10 Apr 2020 17:07:21 GMT

93231821

☁

✎

🗑

3

📁

pytorch

Fri, 10 Apr 2020 17:07:27 GMT

1757016770

☁

✎

🗑

4

📁

models

Fri, 10 Apr 2020 17:07:38 GMT

1053531967

☁

✎

🗑

5

📄

ml\_examples.zip

Fri, 10 Apr 2020 17:07:38 GMT

1274250233

☁

✎

🗑

NEW FOLDER 버튼으로 현재 경로 상에 새로운 폴더를 생성할 수 있으며, UPLOAD FILES 버튼으로 현재 경로에 로컬 파일을 업로드 할 수도 있습니다. 이런 파일 작업은 앞서 설명한 연산 세션 마운트 방식으로도 모두 수행 가능합니다.

파일 탐색기를 닫기 위해서는 우측 상단의 X 버튼을 누르면 됩니다.

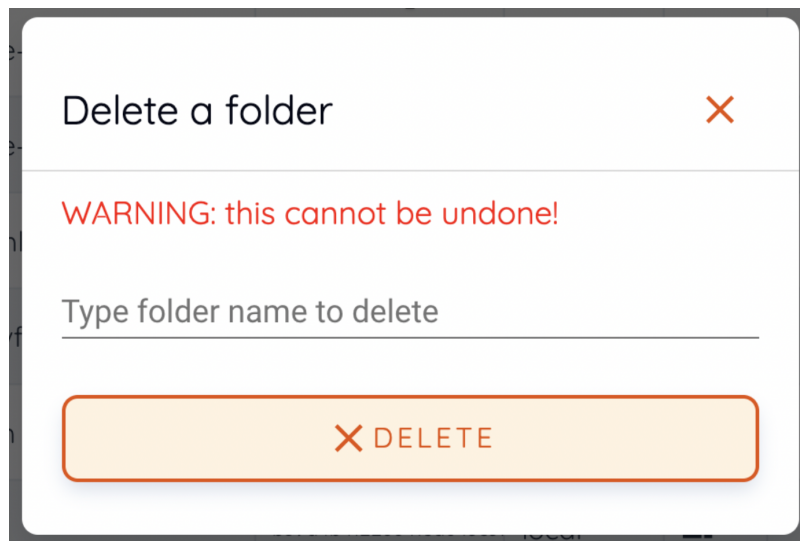


## 7.3 폴더 이름 변경

Storage 폴더의 이름을 변경할 수 있는 권한을 가지고 있는 경우, Control 열의 편집 아이콘을 클릭해서 이름 변경 작업을 수행할 수 있습니다. 편집 아이콘을 클릭하면 이름 변경 다이얼로그가 나타나게 됩니다. 새로운 폴더 이름을 기록한 후 RENAME 버튼을 클릭하십시오.

## 7.4 폴더 삭제하기

Storage 폴더의 삭제 권한을 가지고 있는 경우, Control 열의 휴지통 아이콘을 눌러서 폴더 삭제할 수 있습니다. 삭제 버튼을 클릭하면 확인 다이얼로그가 나오는데, 실수로 인한 삭제를 방지하기 위해 삭제할 폴더의 이름을 입력한 후에 삭제 가능하게 만들어져 있습니다.



여기서 생성한 폴더는 연산 세션을 만들 때 마운트 할 수 있습니다. 폴더는 사용자의 기본 작업 폴더인 /home/work/ 폴더 아래에 마운트 되며, 이 폴더에 저장한 파일은 연산 세션이 종료되어도 사라지지 않습니다(단, 폴더 자체를 삭제하면 사라지게 됩니다).



## 7.5 자동 마운트 폴더

데이터 & 폴더 페이지에는 Automount Folders 탭이 있습니다. 이 탭을 클릭하면 이름이 점(.)으로 시작하는 폴더 리스트를 볼 수 있습니다. 폴더를 생성할 때 점(.)으로 시작하는 이름을 지정하면 Folders 탭이 아닌 Automount Folders 탭에 추가됩니다. Automount Folders는 연산 세션 생성 시 별도로 마운트 하지 않더라도 자동으로 홈 디렉토리에 마운트 되는 특수한 폴더입니다. 이를 이용하여, .local, .linuxbrew, .pyenv 등과 같은 Storage 폴더를 생성해서 사용하면 연산 세션에 관계없이 늘 일정한 사용자 패키지 또는 환경을 구성할 수 있습니다.

Automount Folders 사용에 관한 보다 자세한 내용은 [자동 마운트 폴더 사용 예제](#) 섹션을 참고 하십시오.

Data & Storage

Folders

Automount Folders

Shared Data

Mod

+ NEW FOLDER

Folders starting with a .(dot) are automatically mounted when a new session is started.

#	Name	id	Location	Ty...	Permissi...	Control
1	.local	9abddbcec55b46fd9e05b83	local		<div><div>R</div><div>W</div><div>D</div></div>	<div><div></div><div></div><div></div><div></div><div></div><div></div></div>
2	.logs	a028e539662a4cf0be67ef5	local		<div><div>R</div><div>W</div><div>D</div></div>	<div><div></div><div></div><div></div><div></div><div></div><div></div></div>
3	.linuxbrew	20f8a7c266e44f07893fcc4	local		<div><div>R</div><div>W</div><div>D</div></div>	<div><div></div><div></div><div></div><div></div><div></div><div></div></div>

## CHAPTER 8

---

### 폴더를 마운트 하여 연산 세션 생성

---

연산 세션을 시작하면 사용자는 `/home/work/` 폴더에 접근 권한을 가지게 되는데, `/home/work/` 및 그 하위에 생성된 일반 폴더와 파일은 연산 세션이 삭제되면 함께 사라지게 됩니다. 이는 연산 세션이 컨테이너에 기반하여 동적으로 생성/삭제 되기 때문으로, 컨테이너 내부의 데이터를 컨테이너의 수명 주기와 상관없이 보존하려면 별도의 호스트 폴더를 컨테이너에 마운트 한 뒤 마운트 된 폴더 내에 파일을 생성해야 합니다. Backend.AI에서는 Storage 폴더 연산 세션 생성 시 마운트 할 수 있는 기능을 제공하고 있습니다.

Sessions 페이지로 이동해서 **START** 버튼을 눌러 새로운 세션을 생성해 보겠습니다. 세션 생성 다이얼로그에서 **Folders to mount**를 클릭하면 현재 사용자가 마운트 할 수 있는 폴더의 리스트를 볼 수 있습니다. 이 중에서 마운트 하고자 하는 폴더를 클릭하여 추가할 수 있습니다. 여러 개를 클릭해서 여러 폴더를 동시에 마운트 할 수도 있습니다. 이 예제에서는 `user1-ml-test` 와 `user2-vfolder` 두 개의 폴더를 마운트한 후 세션을 생성 하겠습니다.

**Start new session** [X]

Environments: TensorFlow (NGC) | Version: 19.11-tf2-py3

Resource Group: h-sgroup-1 | Session name (option...)

Folders to mount: user1-ml-test, user2-vfolder

**Resource allocation** [^]

MALL	SMALL-GPU	MEDIUM
1 CPU 2GB RAM 0 GPU	1 CPU 2GB RAM 0.3 FGPU	2 CPU 4GB RAM NO GPU

Advanced | Custom allocation [v]

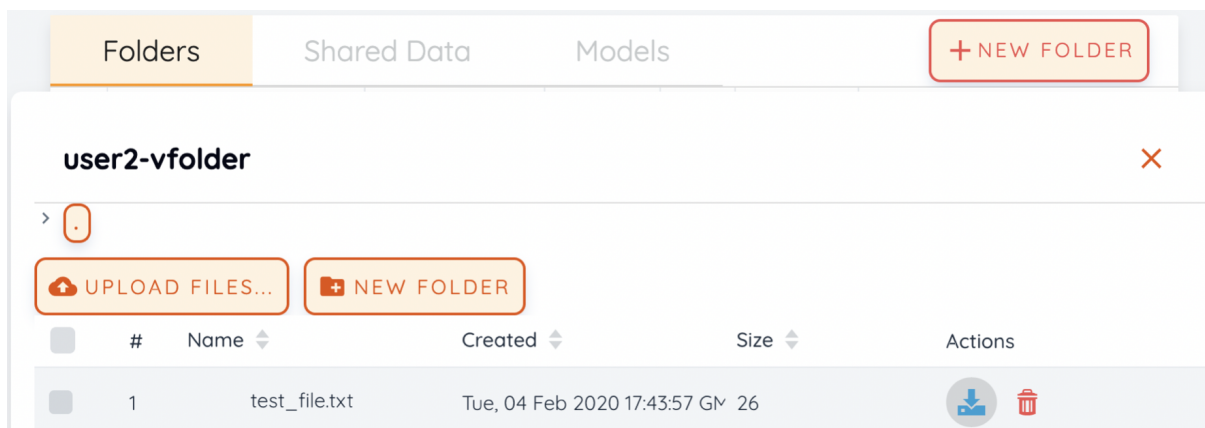
**LAUNCH**

생성된 세션에서 터미널 아이콘을 클릭해 터미널 창을 띄워 봅시다. 터미널에서 `ls` 명령을 내려보면 `user1-ml-test` 폴더와 `user2-vfolder` 폴더가 홈 디렉토리 하위에 마운트 되어 보이는 것을 알 수 있습니다. 연산 세션이 삭제된 후에도 파일을 보존할 수 있는지 확인하기 위해 `user2-vfolder` 하위에 `test_file` 파일을 생성해 보겠습니다. 이 파일의 내용은 "file inside user2-vfolder"로 채우겠습니다.

```
work@7d51e0e00bbf:~$ ls
id_container  user1-ml-test  user2-vfolder
work@7d51e0e00bbf:~$ ls user2-vfolder/
basic_classification.ipynb  tc-perft01.txt
work@7d51e0e00bbf:~$ echo "file inside user2-vfolder" > user2-vfolder/test_file
work@7d51e0e00bbf:~$ ls user2-vfolder/
basic_classification.ipynb  tc-perft01.txt  test_file
work@7d51e0e00bbf:~$ cat user2-vfolder/test_file
file inside user2-vfolder
work@7d51e0e00bbf:~$
```

user2-vfolder 를 대상으로 ls 명령을 실행해보면 파일이 잘 생성된 것을 확인할 수 있습니다. cat 명령으로 파일 내용도 확인해 보았습니다.

이제 세션을 삭제한 후 Data & Storage 페이지로 이동합니다. user2-vfolder 폴더를 찾아 파일 탐색기를 열고 test\_file 파일이 존재하는 것을 확인합니다. Actions의 파일 다운로드 버튼을 클릭하여 파일을 로컬 머신에 받아 내용을 확인해서 파일의 내용이 "file inside user2-vfolder" 임을 확인합니다.



이처럼 연산 세션을 생성할 때 폴더 마운트를 하고, 마운트 된 폴더에서 파일 작업을 수행하면 연산 세션이 종료된 이후에도 데이터를 보존할 수 있습니다.

## 8.1 자동 마운트 폴더를 이용하여 연산 세션 환경 구성하기

연산 세션에 사전 설치되어 있지 않은 새 프로그램이나 라이브러리를 필요로 할 때가 있습니다. 이런 경우, 연산 세션의 수명 주기와 관계없는 Storage 폴더 특성과 **자동 마운트 폴더**를 이용해서 패키지를 설치하고, 연산 세션의 종류에 상관없이 일정한 환경을 구성할 수 있습니다.

## pip을 통한 파이썬 패키지 설치

.local 이라는 이름의 폴더를 생성하면 파이썬 사용자 패키지를 해당 폴더에 설치하고 공유할 수 있습니다. pip 에 -user 옵션을 붙여 패키지를 설치하면 사용자 홈폴더 하위에 있는 .local 폴더에 패키지를 설치하기 때문입니다 (자동 마운트 폴더는 사용자 홈폴더 하위에 마운트 됩니다). 따라서, 만약 tqdm 이라는 파이썬 패키지를 연산 환경과 상관없이 항상 설치되어 있는 상태를 유지하려면 터미널에서 다음과 같은 명령을 내릴 수 있습니다:

```
pip install --user tqdm
```

이후, 연산 세션을 새로 생성하게 되면 tqdm 패키지가 설치된 .local 폴더가 자동으로 마운트 되므로, tqdm 패키지를 다시 설치하지 않고 사용할 수 있습니다.

**경고:** 여러 파이썬 버전이나 다른 파이썬 버전을 사용하는 세션을 사용할 경우 패키지들의 호환성 문제가 있을 수 있습니다. 이는 .bashrc 편집기능을 통해 PYTHONPATH 를 분기하는 방식으로 적용할 수 있습니다. PYTHONPATH 환경 변수에 지정된 경로에 사용자 pip 패키지가 설치되기 때문입니다.

## Homebrew를 통해 패키지 설치

Ubuntu의 apt 나 CentOS의 yum 같은 패키지 매니저는 일반적으로 sudo 권한을 필요로 합니다. Backend.AI의 연산 세션은 보안을 위해 sudo 권한 및 root 접근이 기본적으로 막혀 있기 때문에 (설정에 따라 접근을 허용할 수도 있습니다), 패키지 설치를 위해서는 [Homebrew on Linux](#) 사용을 권장하고 있습니다. Homebrew는 다음과 같은 방식으로 설치 및 사용할 수 있습니다:

- Data & Storage 페이지에서 .linuxbrew 폴더 생성
- 연산 세션을 생성 (.linuxbrew 폴더가 자동으로 마운트)
- Homebrew on Linux 설치

```
sh -c "$(curl -fsSL https://raw.githubusercontent.com/Linuxbrew/install/master/
↪install.sh)"
export PATH=/home/work/.linuxbrew/bin:$PATH
brew
```

- 패키지 설치

```
brew install fortune
fortune
```

위와 같은 방식으로 자동 마운트 폴더를 이용하여 다양한 설정 관리가 가능합니다. 보다 자세한 사항은 [Backend.AI 위키](#) 에서 확인할 수 있습니다.

## CHAPTER 9

---

### 연산 세션에 SSH/SFTP 접속하기

---

Backend.AI는 생성된 연산 세션(컨테이너)에 SSH/SFTP 접속을 지원하고 있습니다. 이번 절에서는 그 방법에 관해 알아봅니다.

---

**주석:** SSH/SFTP 지원은 데스크탑 앱 콘솔에서만 지원하며, 웹 기반 콘솔에서는 아직 지원하지 않습니다.

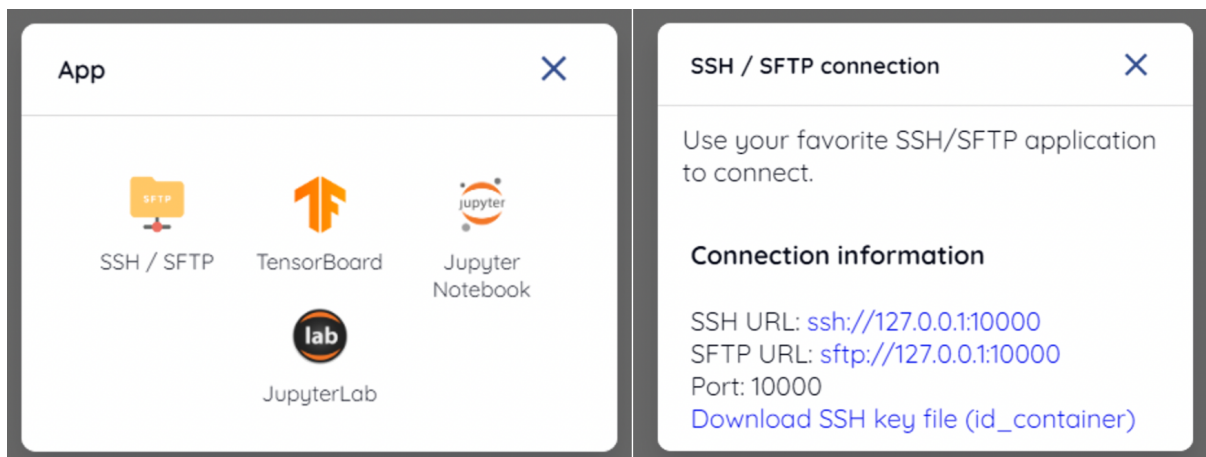
---

#### 9.1 연산 세션에 SSH/SFTP 접속하기 (Linux/Mac)

먼저 연산 세션을 하나 생성한 후 Control의 앱 아이콘(첫 번째 버튼)을 클릭하고 SSH / SFTP 아이콘을 클릭합니다. 그러면 해당 컨테이너 내부에서 SSH/SFTP 접속을 할 수 있도록 해주는 데몬(demon)이 뜨게 되고, 로컬 proxy를 통해 콘솔 앱과 컨테이너 내부의 데몬이 연결됩니다.

**경고:** 이 아이콘을 클릭하기 전에는 해당 세션에 SSH/SFTP 접속을 할 수가 없습니다. 만약, 콘솔 앱을 껐다가 다시 켜게 되면 로컬 proxy와 콘솔 앱 사이의 연결이 초기화 되므로 SSH/SFTP 아이콘을 다시 한 번 클릭해주어야 합니다.

이어 SSH/SFTP connection 정보를 담고 있는 다이얼로그가 뜨는데, SSH URL에 적힌 접속 주소(특히 할당 받은 포트)를 기억한 뒤 id\_container 파일 다운로드 링크를 클릭하여 로컬 머신에 받아 둡니다. 이 파일은 자동으로 생성된 SSH private key입니다. 링크를 통해 다운로드하는 대신 터미널 또는 Jupyter Notebook으로 /home/work/ 하위에 위치하고 있는 id\_container 파일을 다운로드 할 수도 있습니다. 자동 생성된 SSH key는 세션이 새로 생성될 때마다 변할 수도 있는데, 이럴 경우 다시 다운로드 받아야 합니다.



다운로드 받은 SSH private key로 연산 세션에 SSH 접속을 하려면 셸 환경에서 다음과 같은 명령을 내릴 수 있습니다. 이 때 -i 옵션에 다운로드 받은 id\_container 파일의 위치를, -p 옵션에 연산 세션이 할당 받은 포트 번호를 적어주어야 합니다. 연산 세션 내의 사용자는 보통 work 로 설정되는데, 만약 work 가 아닌 다른 계정을 사용하는 경우라면 work@localhost 에서 work 부분을 실제 접속되는 계정으로 바꾸어 주어야 합니다. 정상적으로 명령을 내렸다면 연산 세션에 SSH 접속이 이루어지고 컨테이너 내부의 셸 환경에 접속되는 것을 확인할 수 있습니다.

```
$ ssh -o StrictHostKeyChecking=no \
> -o UserKnownHostsFile=/dev/null \
```

(continues on next page)



(이전 페이지에서 계속)

```
> -i ~/.ssh/id_container \
> work@localhost -p 10022
Warning: Permanently added '[127.0.0.1]:9922' (RSA) to the list of known hosts.
f310e8dbce83:~$
```

SFTP 접속을 하는 방법도 거의 같습니다. SFTP 클라이언트를 실행하고 public key 기반의 접속 설정을 한 후 id\_container 를 SSH private key로 지정하면 됩니다. FTP 클라이언트 별로 사용 방법은 상이할 수 있으므로, 자세한 방법은 각 FTP 클라이언트 매뉴얼을 참고하십시오.

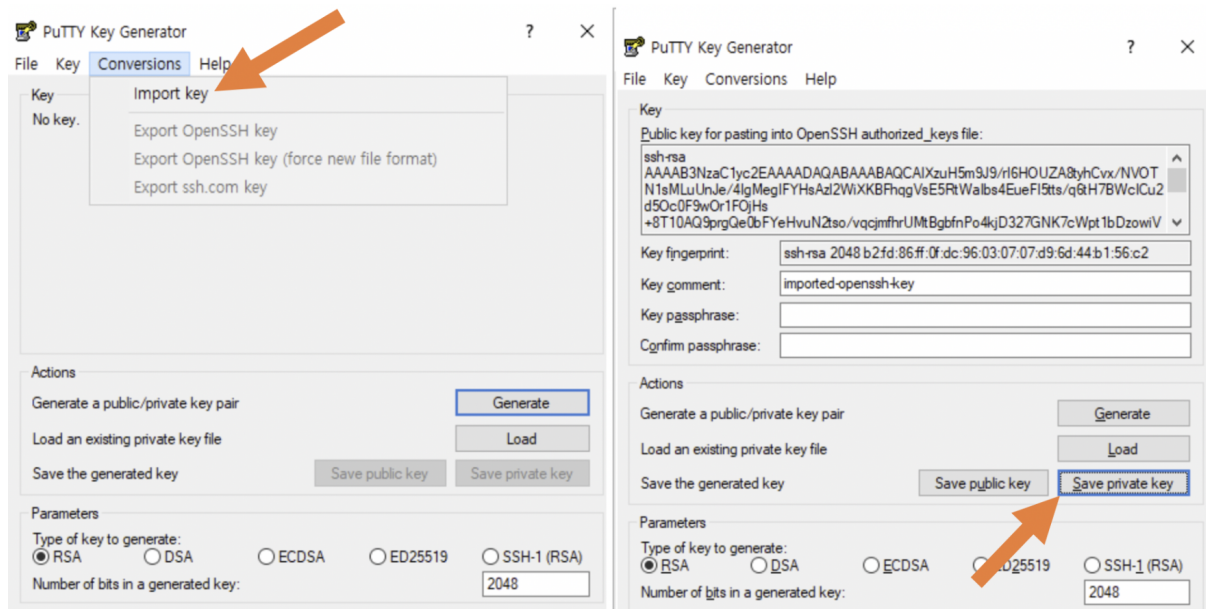
**주석:** SSH/SFTP 접속 포트 번호는 연산 세션을 생성할 때마다 임의의 값이 할당됩니다. 만약, SSH/SFTP 접속 포트를 특정한 값으로 고정하고 싶다면, 사용자 설정 메뉴에서 “Preferred SSH Port” 항목에 포트 번호를 기입할 수 있습니다. 세션 내의 다른 서비스와의 충돌을 가능한 방지하기 위해 포트 번호는 10000-65000 사이의 값을 지정하는 것을 권장합니다. 다만, 두 개 이상의 연산 세션에서 동시에 SSH/SFTP 연결을 맺게되는 경우, 두 번째 SSH/SFTP 연결은 지정된 포트를 사용할 수 없으므로 (첫 번째 SSH/SFTP 연결이 이미 점유) 임의의 포트 번호를 할당받게 됩니다.

**주석:** 만약 id\_container 대신 사용자 자신의 SSH keypair를 사용하고 싶다면, .ssh 라는 이름의 사용자 폴더를 하나 만듭니다. 그 폴더 안에 authorized\_keys 파일을 만든 후 사용할 SSH public key를 파일 내용으로 추가하면, 연산 세션 생성 후 id\_container 파일을 다운로드 할 필요없이 자신의 SSH private key를 통해 SSH/SFTP 접속할 수 있습니다.

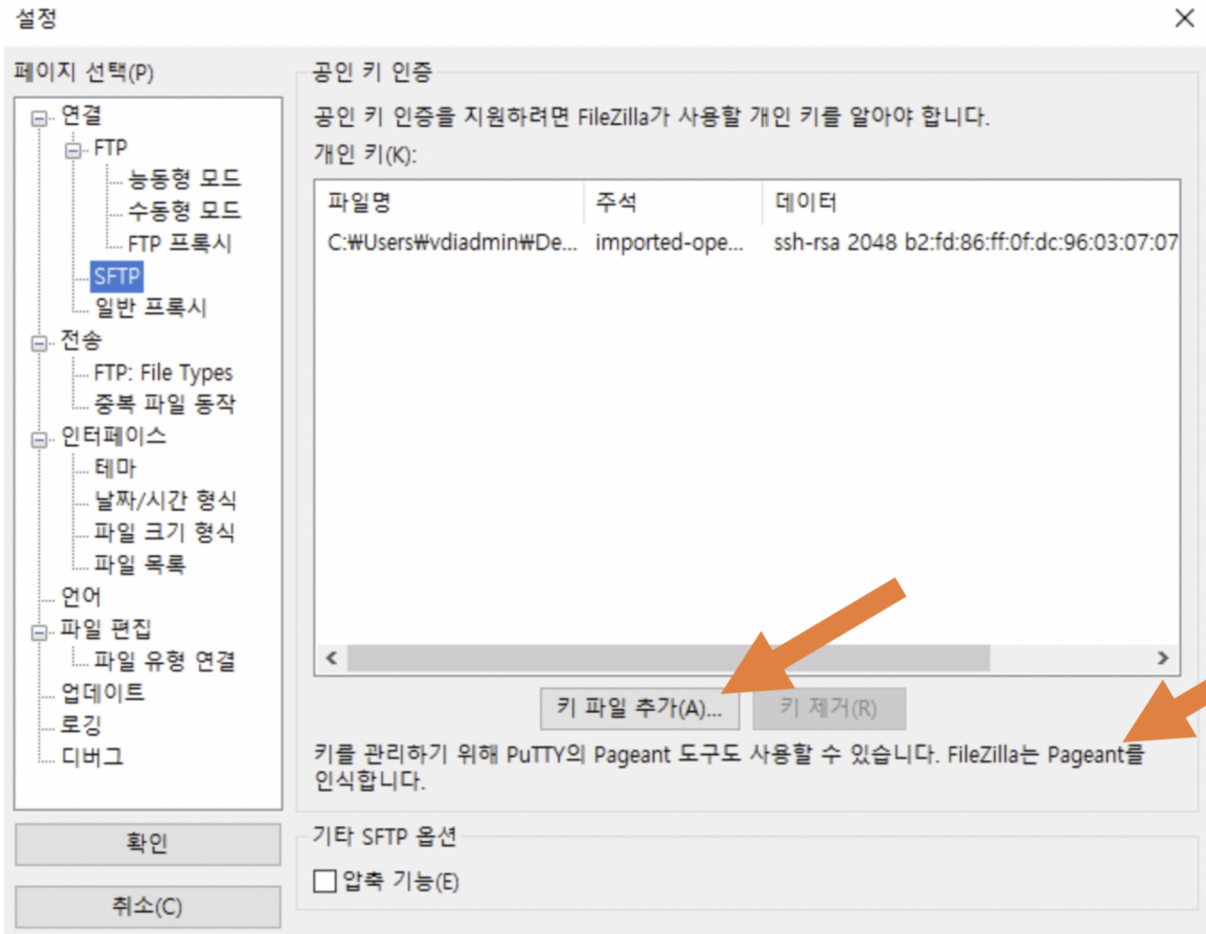
## 9.2 연산 세션에 SSH/SFTP 접속하기 (Windows / FileZilla)

앱에서 지원하는 SSH/SFTP 접속 방법은 OpenSSH 기반 public key 접속(RSA2048)입니다. 윈도우즈의 PuTTY 등 클라이언트를 통해 접속하려면 PuTTYgen 등과 같은 프로그램으로 SSH private key를 ppk 파일로 변환하는 작업이 추가로 합니다. 변환 방법은 다음 링크를 참조 할 수 있습니다: <https://wiki.filezilla-project.org/Howto>. 보다 쉬운 설명을 위해 이번 절에서는 Windows에서 FileZilla 클라이언트를 통해 SFTP 접속하는 방법에 관해 알아보겠습니다.

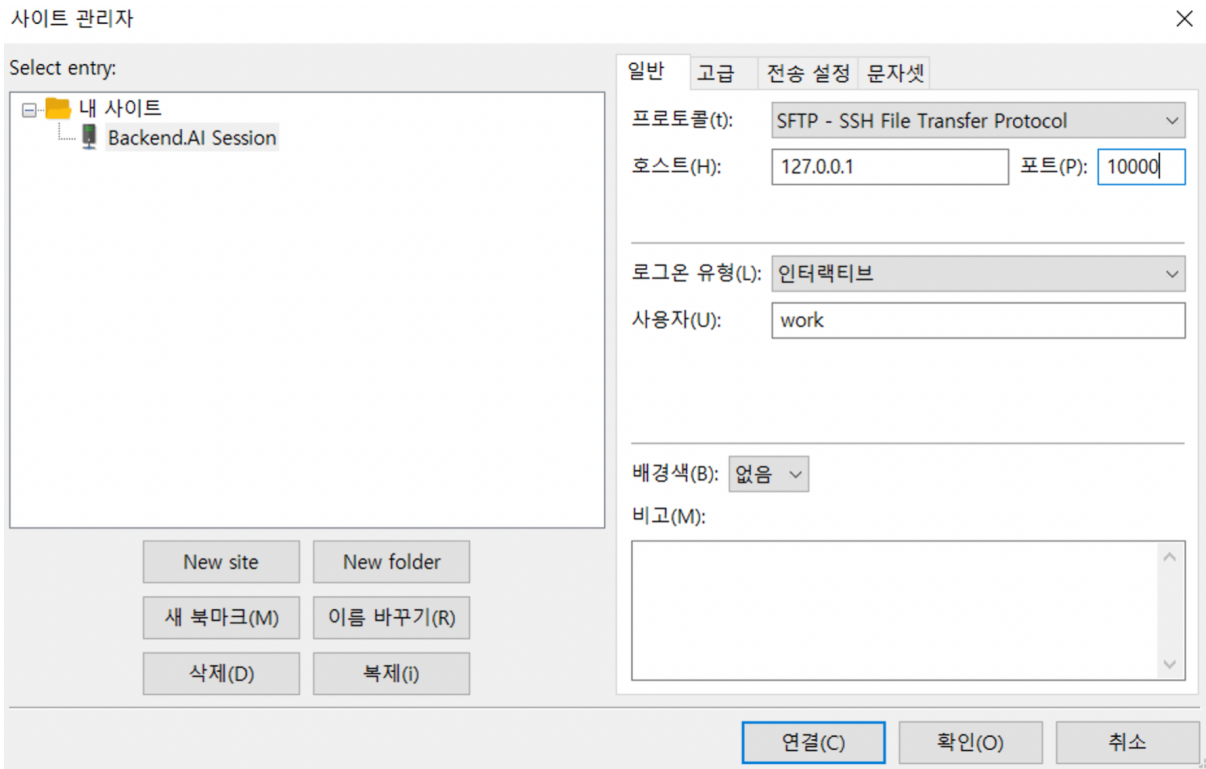
Linux/Mac에서의 접속 방법을 참고하여 연산 세션을 생성하고 접속 포트 확인과 id\_container 다운로드를 수행합니다. id\_container 는 OpenSSH 기반의 키이므로, 윈도우 또는 ppk 형태의 키만 지원하는 클라이언트를 사용할 경우 변환을 해야 합니다. 여기서는 PuTTY와 함께 설치되는 PuTTYgen 프로그램을 통해 변환을 해보겠습니다. PuTTYgen 프로그램을 실행한 후 Conversions 메뉴의 import key를 클릭합니다. 이어 뜨는 파일 선택 다이얼로그에서 다운로드 한 id\_container 파일을 선택합니다. PuTTYGen의 Save private key 버튼을 클릭하여 id\_container.ppk 라는 이름으로 파일을 저장합니다.



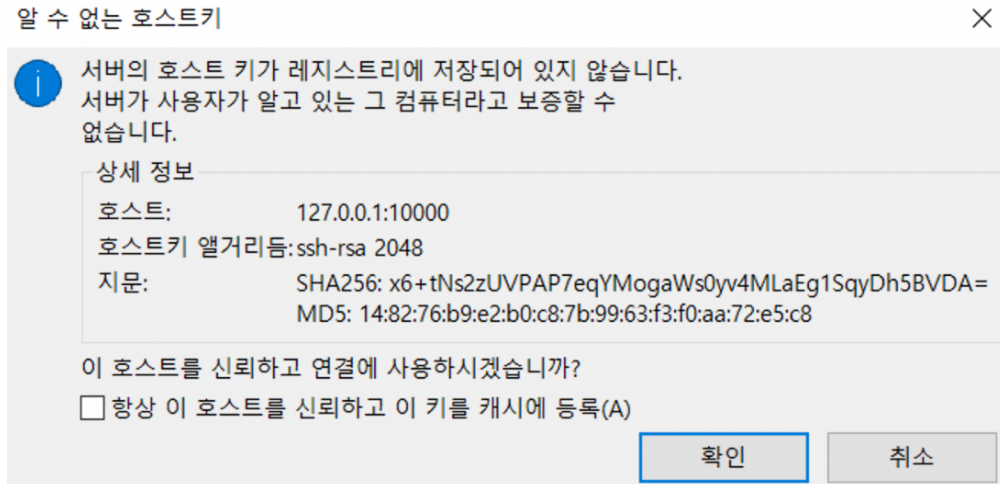
이제 FileZilla 클라이언트 앱을 실행한 후 설정-연결-SFTP 페이지로 들어가 생성한 id\_container.ppk (OpenSSH를 지원하는 클라이언트의 경우 id\_container) 키 파일을 등록해 줍니다.



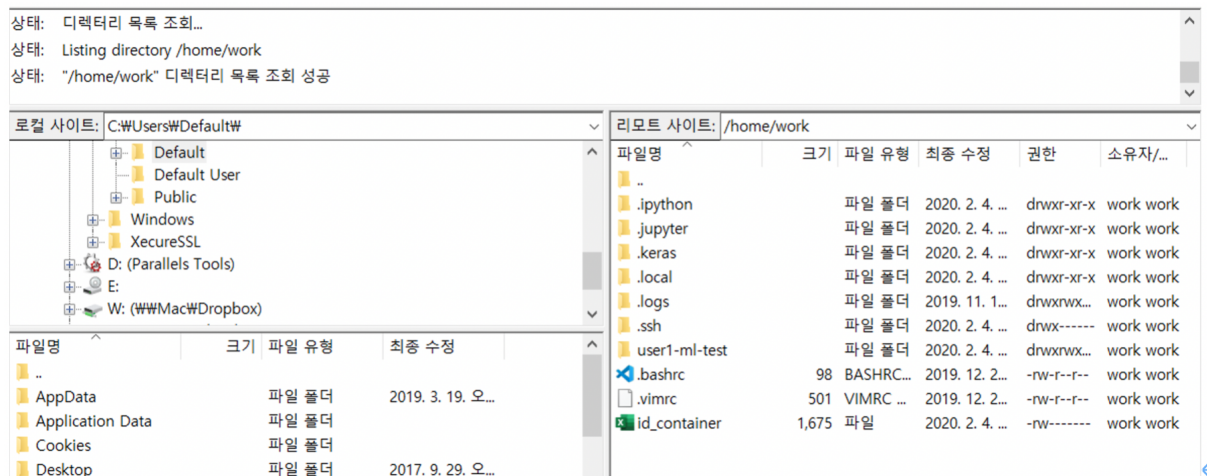
사이트 관리자를 열고 새 사이트를 하나 만든 후 다음과 같이 접속 정보를 적어 줍니다.



해당 컨테이너에 처음 접속하는 경우 다음과 같은 확인 창이 뜰 수 있습니다. 확인 버튼을 눌러 호스트 키를 저장합니다.



잠시 기다린 후 다음과 같이 접속이 되는 것을 확인할 수 있습니다. SFTP 접속을 통해 /home/work/ 및 마운트 된 Storage 폴더에 대용량 파일을 전송하는 등의 작업을 수행할 수 있습니다.



## CHAPTER 10

### 관리자 기능

관리자 계정으로 로그인 하면 왼쪽 사이드바에 Administration 메뉴가 추가로 보입니다. Backend.AI에 등록된 사용자 정보는 Users 탭에서 볼 수 있습니다. Domain admin의 경우 도메인에 속한 사용자 정보만 확인할 수 있고, super admin은 전체 사용자 정보를 조회할 수 있습니다. 사용자의 생성 및 비활성화는 super admin만 할 수 있습니다.

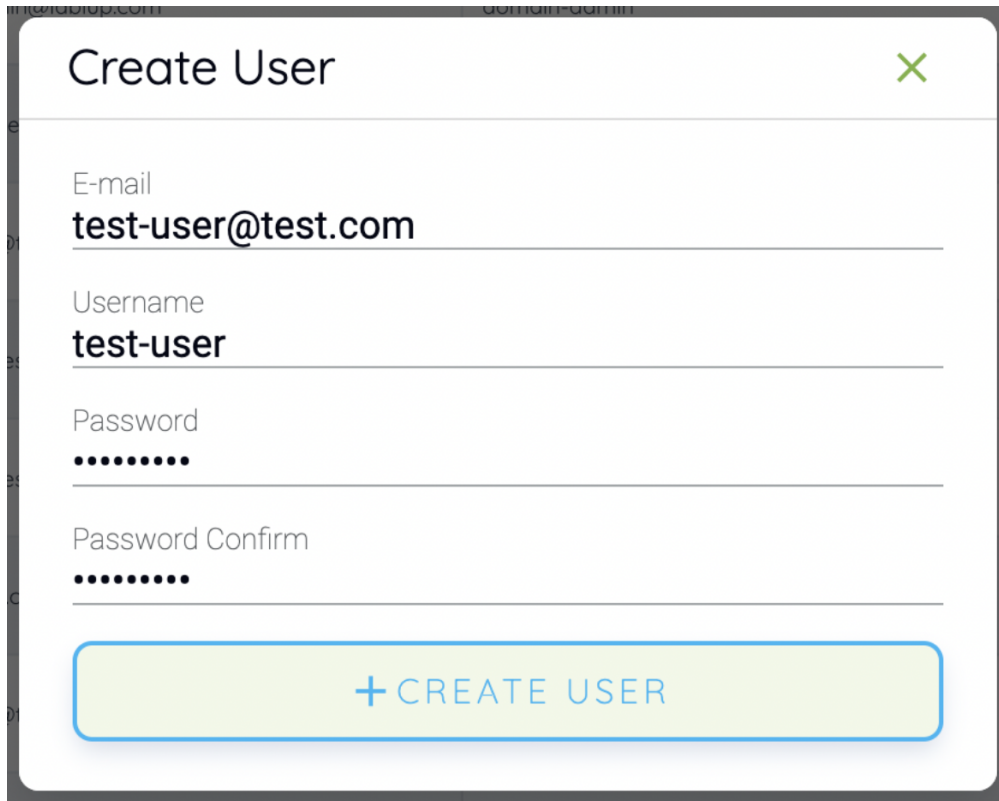
The screenshot displays the Backend.AI Administration interface. On the left is a dark sidebar with a menu. The 'Administration' section is expanded, showing 'Users' as the selected option. The main content area has a green header 'User Credentials & Policies' and three tabs: 'Users' (active), 'Credentials', and 'Resource Policies'. There are two green buttons: '+ ADD CREDENTIAL' and '+ CREATE USER'. Below the tabs is a table of users.

#	User ID	Name	Control
1	monitor@labup.com	monitor	[Menu] [Settings] [Delete]
2	domain-admin@labup.com	domain-admin	[Menu] [Settings] [Delete]
3	user2-A-a@test.com	user2-A-a	[Menu] [Settings] [Delete]
4	admin1-B-a@test.com	admin1-B-a@test.com	[Menu] [Settings] [Delete]
5	user1-B-a@test.com	user1-B-a@test.com	[Menu] [Settings] [Delete]
6	user1-B-b@test.com	user1-B-b@test.com	[Menu] [Settings] [Delete]
7	user@labup.com	user	[Menu] [Settings] [Delete]






## 10.1 새로운 사용자 생성 및 정보 갱신

사용자는 CREATE USER 버튼을 클릭하여 생성할 수 있습니다. 이 때 비밀번호는 8자 이상, 알파벳/특수문자/숫자를 1개 이상 포함해야 합니다.



The image shows a 'Create User' dialog box with a close button (X) in the top right corner. It contains four input fields: 'E-mail' with the value 'test-user@test.com', 'Username' with the value 'test-user', 'Password' with masked characters '.....', and 'Password Confirm' with masked characters '.....'. At the bottom is a large green button with a blue outline and the text '+ CREATE USER'.

사용자가 생성된 것을 확인합니다.

13	test-user@test.com	test-user	  
----	--------------------	-----------	---

Controls 열에서 초록색 정보 버튼을 클릭하면 보다 자세한 사용자 정보를 확인할 수 있습니다. 사용자가 소속된 Domain과 Group 정보 또한 확인할 수 있습니다.

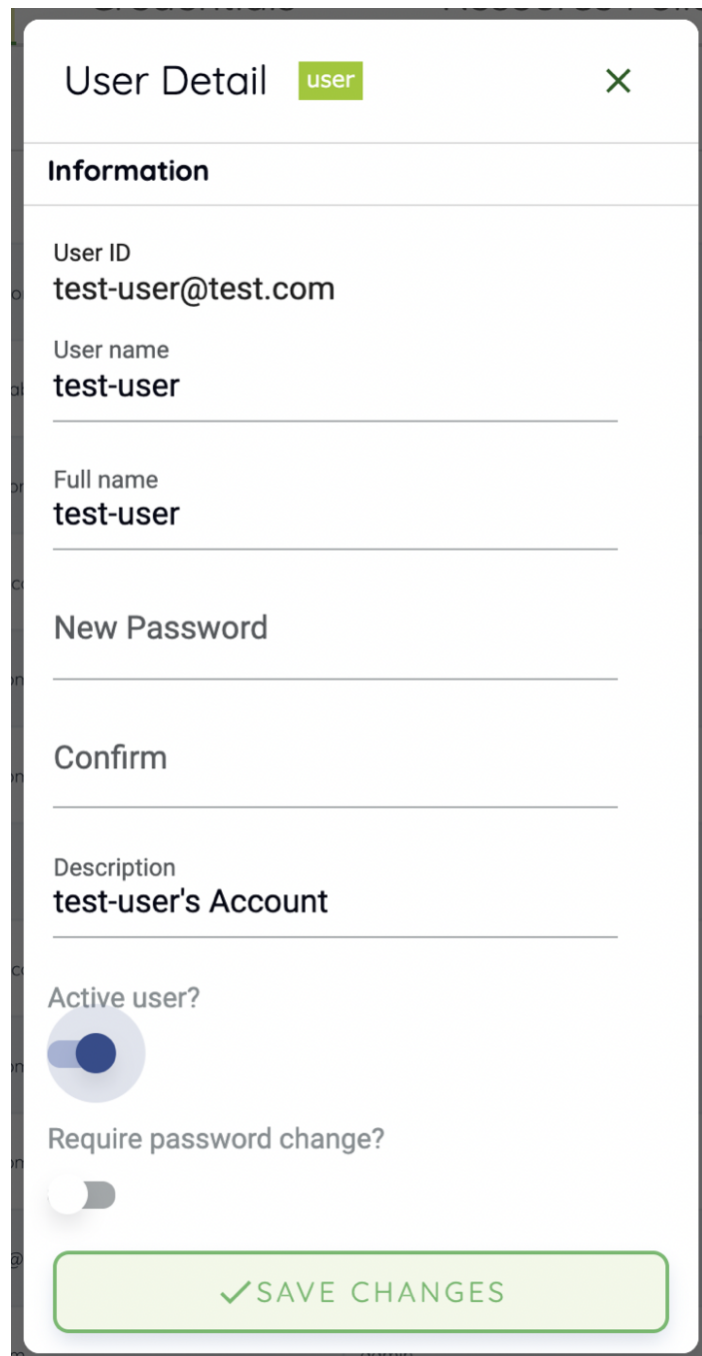
User Detail

user

X

Information	Association
User ID test-user@test.com	Domain default
User name test-user	Role user
Full name test-user	Groups • default
Description test-user's Account	
Active user? Yes	
Require password change? No	

Controls 열의 톱니바퀴로 된 설정 아이콘을 클릭하면 이미 존재하는 사용자의 정보를 업데이트 할 수 있습니다. 사용자의 이름, 비밀번호, 활성/비활성 여부 등을 변경하여 저장할 수 있습니다.



**User Detail** user ✕

---

**Information**

User ID  
**test-user@test.com**

User name  
**test-user**

---

Full name  
**test-user**

---

New Password  
\_\_\_\_\_

Confirm  
\_\_\_\_\_

Description  
**test-user's Account**

---

Active user?  
☒

Require password change?  
☐

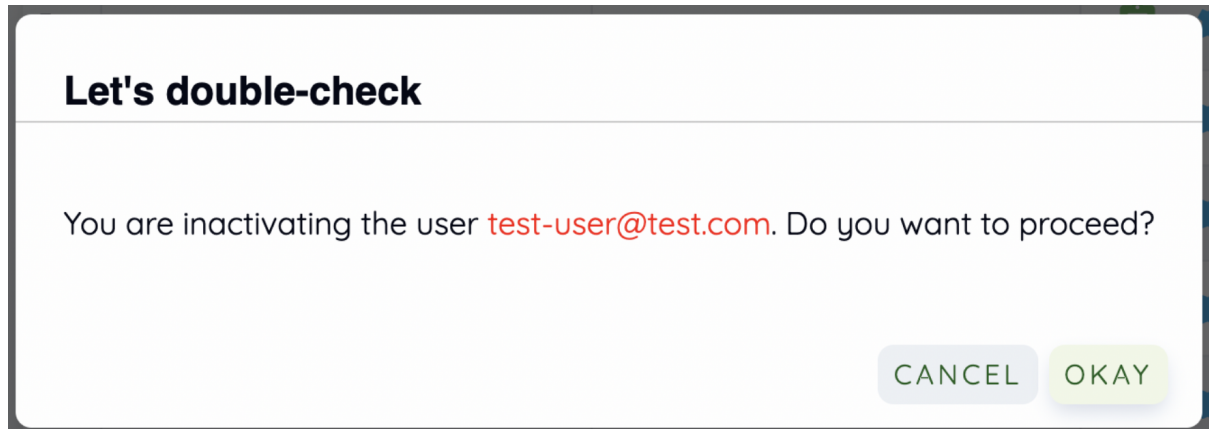
✓ **SAVE CHANGES**

## 10.2 사용자 계정 비활성화

사용자 계정을 삭제하는 기능은 사용량 통계 처리 및 메트릭 보존, 실수로 인한 계정 유실을 막기 위하여 만약을 위해 관리자라 할지라도 막혀 있습니다. 대신 사용자 계정을 비활성화 해서 사용자가 해당 계정을 통해 로그인 하는 것을 막을 수 있습니다. Controls 열의 휴지통 아이콘을 클릭합니다. 확인을 위한 다이얼로그가 뜨는데, OKAY 버튼을 클릭하면 사용자를



비활성화 할 수 있습니다.



비활성 상태로 바뀐 사용자 계정은 Users 탭에서 보이지 않습니다.

## 10.3 사용자의 키페어 관리

사용자 계정에는 보통 하나 이상의 키페어가 할당되어 있습니다. 키페어는 사용자 로그인 후 Backend.AI 서버로 요청을 보낼 때 인증을 위해 사용 됩니다. 로그인을 위해서는 사용자 이메일 및 비밀번호를 통한 인증이 필요하지만, 사용자가 서버로 보내는 때 요청은 키페어에 기반하여 인증하게 됩니다.

한 사용자가 여러 개의 키페어를 가질 수 있지만, 사용자의 키페어 관리 부담을 줄이기 위해 현재는 사용자의 키페어 중 하나만 사용하여 요청을 보내도록 하고 있습니다. 또한, 새 사용자를 생성하면 자동으로 키페어가 하나 만들어지므로, 사용자 생성 시 별도로 키페어를 생성하여 할당할 필요는 없습니다.

키페어는 Users 페이지의 Credentials 탭에서 조회할 수 있습니다. 현재 활성화된 키페어가 바로 출력이 되고, 비활성 키페어를 조회하려면 하단의 Inactive 패널을 클릭하면 됩니다.

Users

Credentials

Resource Policies

+ ADD CREDENTIAL

Active

User ID

Access Key

Permissi...

Key age

Resource Policy

Allocation

Control

er1-B-b@test.com

AKIAJ7Q26VOKERDW

user

156 Days

(Mon, 02 Sep 2019 02:4

∞cores

16GB

2fGPU

0GB

10Folders

0 / 0

1000 req./15r

Sess. 162 queries

r@lablup.com

AKIANABBDUSEREXA

user

218 Days

(Tue, 02 Jul 2019 01:54

∞cores

16GB

2fGPU

0GB

10Folders

0 / 0

30000 req./1

Sess. 4729 queries

r2-A-a@test.com

AKIARZM3X7LLPXE2

user

153 Days

(Wed, 04 Sep 2019 14:4

∞cores

16GB

2fGPU

0GB

10Folders

0 / 0

1000 req./15r

Sess. 86 queries

nin@lablup.com

AKIAIOSFODNN7EXAN

admin user

218 Days

(Tue, 02 Jul 2019 01:54

∞cores

16GB

2fGPU

0GB

10Folders

0 / 0

30000 req./1

Sess. 143191 queries

main-admin@lablup

AKIAHUKCHDEZGEXA

admin user

218 Days

(Tue, 02 Jul 2019 01:54

∞cores

16GB

2fGPU

0GB

10Folders

0 / 0

30000 req./1

Sess. 874 queries

er1-A-b@test.com

AKIA4PNNTDY6FHC4C

user

156 Days

(Mon, 02 Sep 2019 02:4

∞cores

16GB

2fGPU

0GB

10Folders

0 / 0

1000 req./15r

Sess. 7 queries

Inactive

Users 탭과 마찬가지로 Controls 열의 버튼을 이용해서 키페어의 상세 정보를 확인하거나 업데이트 할 수 있습니다. 파란색 휴지통 버튼을 클릭하면 해당 키페어를 비활성화 할 수 있고, 빨간색 휴지통 버튼을 클릭하면 키페어를 완전히 삭제할 수 있습니다. 단, 키페어를 사용해서 연산 세션을 생성한 적이 있으면 삭제할 수 없습니다. 만약, 실수로 키페어를 삭제한 경우 우측 상단의 ADD CREDENTIAL 버튼을 클릭하여 해당 사용자의 키페어를 다시 생성할 수 있습니다. 필요한 경우 Advanced 패널을 클릭해서 access key와 secret key를 직접 명시적으로 적어줄 수도 있습니다.




Rate Limit 필드는 15분 동안 Backend.AI 서버로 보내는 요청의 최대 수를 지정하는 곳임

니다. 예를 들어 1000으로 설정한 경우, 해당 키페어로는 15분 동안 1000개 이상의 API 요청을 보내면 서버에서 에러를 발생하고 요청을 받아들이지 않습니다. 기본값을 사용하다가 사용자의 패턴에 따라 API 요청 빈도가 많을 경우 이 값을 증가시키는 것을 권장합니다.

## 10.4 리소스 정책 관리

리소스 정책은 Users 페이지의 Resource Policies에서 조회 및 수정할 수 있습니다. 리소스 정책을 통해 키페어 별로 자원 사용 및 기타 세션 사용 관련 정책을 설정할 수 있습니다. 필요에 따라 사용자용/연구용 등 목적으로 여러 가지 리소스 정책을 설정하여 키페어에 분리 적용할 수 있습니다.

**주석:** 도메인과 그룹 수준에서 자원 정책을 설정하려면 super admin 전용 UI인 Manager Hub를 사용해야 합니다. 앱/웹 콘솔에서는 현재 키페어 기반의 자원 정책만 설정할 수 있습니다. Manager Hub는 엔터프라이즈 버전의 일부입니다.

Users		Credentials		Resource Policies		<div>+ ADD CREDENTIAL</div>	
Policy groups						<div>+ CREATE POLICY</div>	
#	Name ▾	Resources		Concurren... ▾	Cluster size ▾	Storage Nodes	Control
1	default	<div><div> ∞</div>cores</div> <div><div> ∞</div>GPU</div> <div><div> ∞</div>GB</div> <div><div> 16</div>GB</div> <div><div> 2</div>fGPU</div> <div><div> 10</div>Folders</div>	30	1	local		

위 예제 이미지에서는 default 라는 이름의 리소스 정책 하나가 있습니다. Control 열의 설정 아이콘을 클릭하여 자원 정책을 변경할 수 있습니다. 설정값을 변경한 후 UPDATE를 클릭하면 저장할 수 있습니다.

Update resource policy

Policy Name

default

Resource Policy

CPU

RAM(GB)

GPU

fGPU

16

2

☒ Unlimited

☐ Unlimited

☒ Unlimited

☐ Unlimited

Sessions

Container per session

Idle timeout (sec.)

Concurrent Jobs

1

0

30

☐ Unlimited

☒ Unlimited

☐ Unlimited

Folders

Allowed hosts

Capacity(GB)

Max.#

local

0

10

☒ Unlimited

+ UPDATE

각 설정값의 의미는 다음과 같습니다.

- CPU: 키페어가 사용할 수 있는 최대 CPU core 수. 예를 들어, 4로 설정한다면 해당 리소스 정책을 사용하는 키페어는 4개 이상의 코어를 컨테이너에 할당할 수 없습니다. 이 때, 코어 수는 키페어가 생성한 모든 컨테이너의 합을 기준으로 제한을 걸게 됩니다. 만약 3개의 코어로 컨테이너를 하나 할당한 상황이라면 새로운 컨테이너는 1개의 코어만 할당할 수 있습니다. 또한, Unlimited에 체크하면 별도 리소스 정책을 정하지 않고 서버 자원이 허용하는 대로 자원 할당이 가능합니다. 이는 다른 자원 설정에도 마찬가지로 적용됩니다.
- RAM: 최대 메모리
- GPU: 최대 할당 가능한 물리 GPU의 수. Backend.AI 서버의 GPU 제공 모드가 "device"로 설정된 경우에만 사용됩니다.
- fGPU: 최대 할당 가능한 가상화된 GPU의 수. Backend.AI 서버의 GPU 제공 모드가 "shares"로 설정된 경우에만 사용됩니다. fGPU의 단위는 GPU 장치 수와 상관없으며, 서버에 설정된 SM(streaming multiprocessor)과 GPU 메모리 단위에 따라 정해집니다.
- Container per session: 한 세션이 가질 수 있는 최대 컨테이너의 수를 지정. 여러 개의 컨테이너를 묶어 하나의 세션으로 사용할 수 있는 기능을 위한 설정이며, 여러 컨테이너를 묶는 기능은 개발 중에 있어 현재는 사용되지 않는 값입니다.
- Idle timeout: 작동 중인 세션이 idle timeout에 지정된 시간 동안 사용되지 않으면 자동으로 삭제하게 되는데, 그 시간 간격을 설정할 수 있습니다. 예를 들어, 600으로 설정할 경우 10분 동안 아무런 사용 내역이 없는 세션은 자동 삭제 됩니다. 0으로 설정하거나 Unlimited에 체크할 경우 자동 삭제를 하지 않습니다.
- Concurrent Jobs: 한 번에 한 사용자가 생성할 수 있는 최대 세션의 수입니다. 5로 설정할 경우 해당 정책을 사용하는 키페어는 5개 이상의 연산 세션을 생성할 수 없습니다.
- Allowed hosts: Storage 폴더 사용을 위한 스토리지 및 NFS가 여러 개 마운트 되어 있을 경우, 접근할 수 있는 스토리지 및 NFS 호스트를 제어할 때 사용합니다. 마운트 되어 있더라도 여기서 지정되지 않으면 사용자는 해당 저장소를 사용할 수 없습니다.

다만, 사용자가 속한 도메인/그룹 단에서 `allowed hosts`를 설정한 경우에는 사용할 수 있습니다. 도메인/그룹 단의 설정은 Manager Hub를 통해 할 수 있습니다.

- **Capacity:** 최대 사용 가능한 저장소의 크기를 설정하는 곳입니다. 디스크 크기 제한은 특정한 상황에서만 가능하여 현재는 일반적으로 지원하지 않고 있지 않으며, 일반적인 지원을 위해 개발 중입니다.
- **Max.#:** 생성할 수 있는 최대 Storage 폴더 개수입니다.

CREATE POLICY 버튼을 클릭하여 새로운 리소스 정책을 생성할 수도 있습니다. 각 설정값의 위미는 상기 설명과 동일합니다.

리소스 정책을 생성한 후 키페어에 연결하기 위해서는 Users 페이지의 Credentials 탭으로 가서 원하는 키페어의 Controls 열에 위치한 톱니바퀴 설정 버튼을 누른 후 Select Policy 필드를 클릭하여 선택하면 됩니다.

## 10.5 이미지 관리

세션 생성 시 사용할 이미지 관리는 Environments 페이지의 Images 탭에서 할 수 있습니다. 탭에 들어가면 현재 Backend.AI 서버에서 가지고 있는 모든 이미지의 메타 정보가 출력됩니다. 이미지 별로 속한 레지스트리, 네임스페이스, 이미지 이름, 이미지의 기반 OS, Digest, 요구되는 최소 자원 등의 정보를 확인할 수 있습니다. 관리되고 있는 agent 노드 중 하나 이상에 다운로드 되어 있는 이미지의 경우 좌측에 체크 표시가 됩니다. 체크가 없는 이미지는 어느 agent에도 설치되지 않았음을 의미합니다.

---

**주석:** 특정 agent를 선택하여 이미지를 설치하는 기능은 현재 개발 중에 있습니다.

---

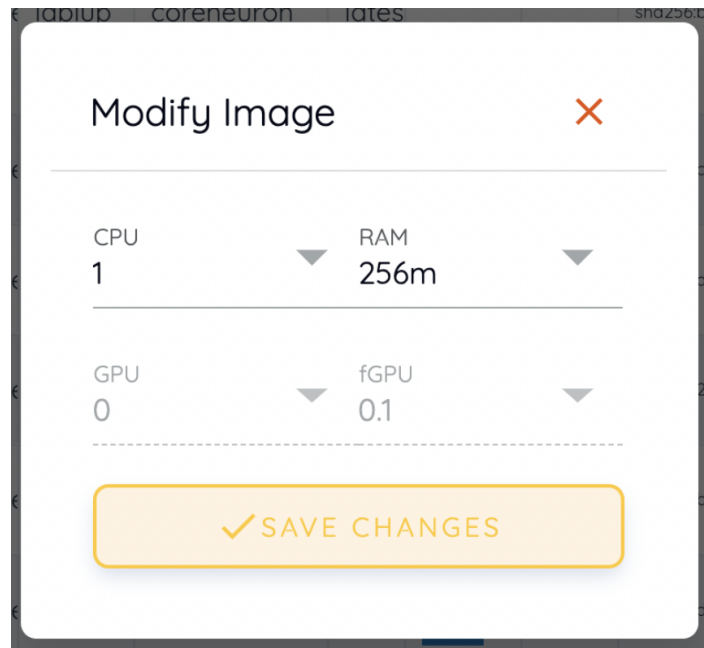


## Environments &amp; Presets

Images		Resource Presets				Registries			
	Regis...	N...	Langua...	\	Base	Co...	Digest	Resource Limit	Cont
<input type="radio"/>	index.docke	lablup	afni	ubun			sha256:d8ca287ff4660520bb7a04	1-∞core 256MB-∞	⚙
<input type="radio"/>	index.docke	lablup	coreneuron	lates			sha256:b845d0f41d506b3099cd84	1-∞core 1GB-∞ 1-∞GPU 0.1-∞fGPU	⚙
<input checked="" type="radio"/>	index.docke	lablup	C++	gpp6	Alpine Li		sha256:c30be1dd8425705ceaed0	1-∞core 256MB-∞	⚙
<input type="radio"/>	index.docke	lablup	Tester	gpp6	Alpine Li C++		sha256:dae98278792b4bd31b53bc	1-∞core 256MB-∞	⚙
<input type="radio"/>	index.docke	lablup	caf	lates	fortran		sha256:2941cdd7275b75e6a816f1d	1-∞core 256MB-∞	⚙
<input type="radio"/>	index.docke	lablup	GCC	gcc6	Alpine Li		sha256:bb624bb509bdde4556ef77	1-∞core 256MB-∞	⚙
<input type="radio"/>	index.docke	lablup	Tester	gcc6	Alpine Li GCC		sha256:d9721cf651cf7aa1b5bf2273	1-∞core 256MB-∞	⚙
<input type="radio"/>	index.docke	lablup	Go	1.10	Alpine Li		sha256:7753e64b4cfe20489c581f9	1-∞core 256MB-∞	⚙

Controls 열의 설정 아이콘을 클릭하여 이미지의 최소 자원 요구량을 변경할 수 있습니다. 이미지마다 최소 동작을 위해 필요한 연산 자원 양 및 하드웨어가 있습니다. (예를 들어, GPU 전용 이미지의 경우 최소 할당 GPU가 있어야 합니다.) 최소 자원량의 기본값은 이미지의 메타데이터에 포함된 채로 제공됩니다. 각 이미지마다 지정된 자원의 양보다 작은 자원으로 세션을 생성하려고 할 경우, 해당 요청은 자동으로 이미지 최소 자원 요구량으로 조정된 후 생성이 진행됩니다.

**경고:** 미리 지정된 값보다 작은 양으로 최소 자원 요구량을 변경하지 마세요! 이미지 메타데이터에 포함된 최소 자원 요구량은 테스트를 거쳐 결정된 값입니다. 변경하려는 최소 자원량에 대하여 정말 잘 알고 있지 않은 경우 기본값으로 남겨두시기 바랍니다.



## 10.6 도커 레지스트리 관리

Environments의 Registries 탭을 클릭하여 현재 연결되어 있는 도커 레지스트리의 정보를 확인할 수 있습니다. `index.docker.io` 는 도커에서 서비스 하는 레지스트리로 기본적으로 등록되어 있습니다.

**주석:** 오프라인 환경일 경우 기본 도커 레지스트리에 접근이 불가능하므로, 우측 휴지통 아이콘을 클릭하여 삭제하면 됩니다.





Controls에 있는 리프레시 아이콘을 클릭하여 해당 레지스트리에 저장된 Backend.AI 용 이미지 정보를 Backend.AI에 받아올 수 있습니다. 레지스트리에 저장된 이미지 중 Backend.AI 용으로 레이블 되지 않은 이미지 정보는 따로 받아오지 않습니다.



ImagesResource PresetsRegistries

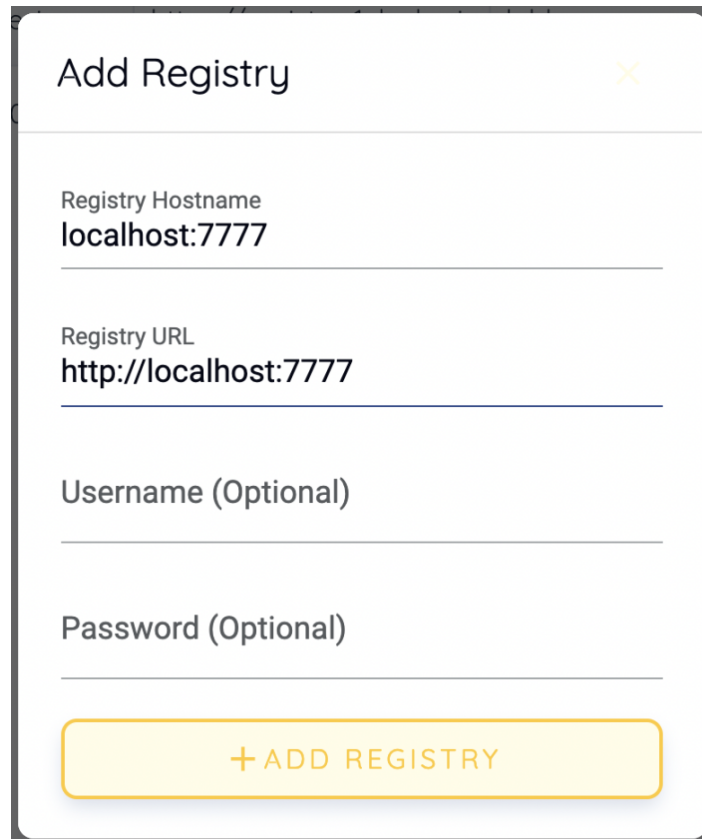
Registries

+ ADD REGISTRY

#	Hostname	Registry URL	Username	Password	Controls
1	index.docker.io	https://registry-1.docker.io	lablup		 
2	localhost:5000	http://localhost:5000			 

ADD REGISTRY 버튼을 클릭하여 운영하고 있는 사설 도커 레지스트리를 추가할 수 있습니다. 이 때, Registry Hostname과 Registry URL 주소는 동일하게 설정하여야 하고, Registry URL의 경우 http:// 또는 https:// 와 같은 scheme을 명시적으로 붙여 주어야 합니다. 또한, 해당 레지스트리에 저장되는 이미지는 반드시 Registry Hostname을 접두어로 한 이름을 가져야 합니다. Username과 Password는 선택 사항으로, 레지스트리에서 별도 인증 설정을 한 경우에는 채워주시면 됩니다.

**주석:** index.docker.io 의 경우 Hostname과 Registry URL가 다른데, 이는 도커에서 기본 레지스트리의 경우 별도 예외 처리를 해주기 때문입니다. 기본 레지스트리가 아닌 경우 Hostname과 Registry URL을 맞추어 주어야 연결할 수 있습니다.

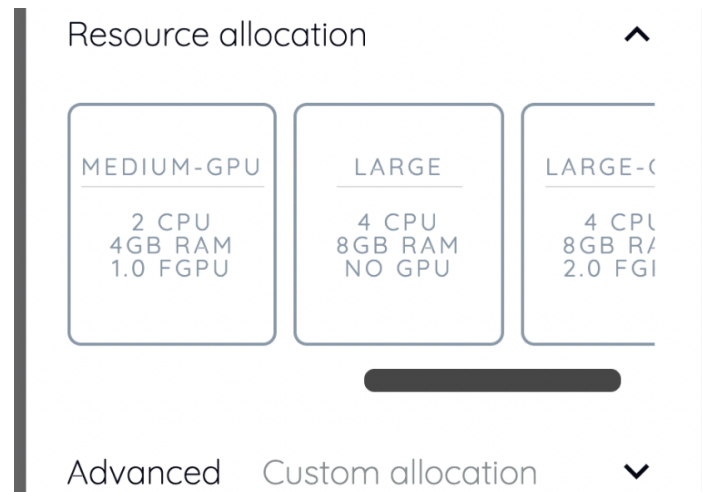


A dialog box titled "Add Registry" with a close button (X) in the top right corner. It contains four input fields: "Registry Hostname" with the value "localhost:7777", "Registry URL" with the value "http://localhost:7777", "Username (Optional)", and "Password (Optional)". At the bottom is a yellow button with the text "+ ADD REGISTRY".

레지스트리를 생성하고 메타 정보를 업데이트 했다고 하더라도 사용자가 바로 해당 레지스트리에 있는 이미지를 사용할 수는 없습니다. Storage 폴더 사용을 위해 allowed hosts를 등록해야 했던 것처럼, 레지스트리 등록 후 도메인 또는 그룹 수준에서 allowed docker registries 필드에 해당 레지스트리를 등록해야 도메인 또는 그룹 소속 사용자가 레지스트리 이미지에 접근할 수 있습니다. Allowed docker registries 등록은 도메인과 그룹 관리 기능이 있는 Manager Hub를 이용해서 할 수 있습니다. 키페어의 리소스 정책에서 allowed docker registries를 설정하는 기능은 아직 제공하지 않고 있습니다.

## 10.7 리소스 프리셋 설정

연산 세션을 생성할 때 Resource allocation 패널에서 다음과 같은 사전 정의된 리소스 프리셋이 출력 됩니다. Super admin에게는 이 리소스 프리셋을 설정할 수 있는 기능을 제공합니다.



Environment 페이지의 Resource Presets 탭으로 이동합니다. 현재 정의되어 있는 리소스 프리셋의 리스트를 확인할 수 있습니다.

Images   Resource Presets   Registries			
Resource Presets			<a href="#">+ CREATE PRESET</a>
#	Name	Resources	Control
1	small	1 cores 2 GB	
2	small-gpu	1 cores 2 GB 0.3 GPU	
3	medium	2 cores 4 GB	
4	medium-gpu	2 cores 4 GB 1.0 GPU	
5	large	4 cores 8 GB	
6	large-gpu	4 cores 8 GB 2.0 GPU	

Controls 열의 설정 아이콘(톱니바퀴 모양)을 클릭하여 리소스 프리셋이 제공할 CPU, RAM, fGPU 등의 자원을 설정할 수 있습니다. 아래 예제의 경우 Backend.AI 서버의 GPU 제공 모드가 shares이므로 GPU 필드는 비활성화 되어 있습니다. 원하는 값으로 자원을 설정한 후 저장하고 연산 세션 생성 시 해당 프리셋이 출력되는지 확인해 봅니다. 프리셋에 정의된 자원량보다 적은 자원만 할당 가능한 경우에는 해당 프리셋이 출력되지 않습니다.

Resources

Create Resource Preset ✕

Preset Name \*  
medium

Resource Preset

CPU 2	RAM (GB) 4
GPU 0	fGPU 0
Shared Memory (GB) 1	

+ ADD

## 10.8 Agent 노드 조회

Super admin의 경우 Resources 페이지의 Connected 탭에서는 현재 Backend.AI에 연결된 agent 워커 노드를 조회할 수 있습니다. 노드의 IP와 연결된 시간, 현재 사용중인 실제 리소스 등을 조회할 수 있습니다. 콘솔 앱에서는 별도로 agent 노드를 조작하는 기능은 제공하지 않습니다.

Connected		Terminated	Maintaining	Resource Group		
#	Endpoint	Region	Starts	Resources	Status	Control
1	i-manager tcp://127.0.0.1:6001	local	2019. 7. 2. 오전 11:05:19	7cores 12.99% 30GB 6.84GB 2fGPU	ALIVE	

Terminated 탭으로 이동하면 한 번 연결되었다가 종료되거나 연결이 끊긴 에이전트의 정보를 확인할 수 있습니다. 노드 관리에 참고 자료로 활용할 수 있습니다.

Connected		Terminated	Maintaining	Resource Group		
#	Endpoint	Region	Starts	Resources	Status	Control
1	i-adrysnui-MacBook tcp://172.30.1.41:6001	local	2020. 2. 24. 오후 5:58:34	10cores 0.00% 32GB 0.00GB	TERMINATED	
2	i-manager-test tcp://127.0.0.1:9999	local	2019. 12. 20. 오후 5:08:03	10cores 0.00% 32GB 0.00GB	TERMINATED	

## 10.9 리소스 그룹 관리

Agent는 리소스 그룹이라는 단위로 묶일 수 있습니다. 예를 들어, V100 GPU를 탑재한 agent가 3대, P100 GPU를 탑재한 agent가 2대 있는데, 사용자에게 두 GPU 자원을 별도로 노출하고 싶을 경우 V100 agent 3대를 하나의 리소스 그룹으로 묶고, 나머지 P100 agent 2대를 다른 리소스 그룹으로 묶어서 관리할 수 있습니다.

리소스 그룹에 특정 agent를 추가하는 작업은 현재 UI 상에서 처리되지 않으며, agent 설치 폴더의 설정 파일 옵션을 수정한 뒤 agent를 재시작 하는 방식으로 가능합니다. 리소스 그룹은 Resources 페이지의 Scaling Group 탭에서 조회할 수 있습니다.

Connected		Terminated	Maintaining	Resource Group				
Resource Groups								<a href="#">+ ADD</a>
#	Name	Description	Active Status	Driver	Driver Optio...	Scheduler	Scheduler O...	Control
1	default	The default a	active	static	{}	fifo	{}	
2	h-sgroup-1	ti-test	active	static	{}	drf	{}	
3	h-sgroup-2		active	static	{}	fifo	{}	

Control 열에서 설정 아이콘을 클릭하여 리소스 그룹을 편집할 수 있습니다. Select scheduler 필드에서 연산 세션 생성 스케줄링 방식을 선택할 수 있는데, 현재 지원하는 방식은 FIFO, LIFO, DRF 세 가지 입니다. FIFO와 LIFO는 가장 처음 또는 가장 마지막에 작업 큐에 들어온 연산 세션을 먼저 생성하는 방식으로 스케줄링 방식이고, DRF는 Dominant Resource Fairness의 약자로 사용자 별로 최대한 공평하게 자원 할당이 가능하도록 조절하여 스케줄링 하는 방식입니다. Active Status를 꺼서 해당 리소스 정책을 비활성화 할 수 있습

니다.

CREATE 버튼을 클릭하여 새로운 리소스 정책을 생성할 수 있습니다.

## 10.10 시스템 설정 조회

System Settings 페이지에서 Backend.AI에 설정된 주요 설정값을 조회할 수 있습니다. 현재는 몇가지 변경 기능 및 설정 조회 기능을 제공하고 있습니다.

---

**주석:** 향후 CLI에서 지원하는 다양한 설정 변경 기능을 GUI에도 계속 추가할 예정입니다.

---

## 10.11 서버 관리 메뉴

Maintenance 페이지로 이동하면 서버를 관리할 수 있는 몇 가지 버튼을 볼 수 있습니다.

- **RECALCULATE USAGE:** 간혹 네트워크 접속이나 도커 데몬의 컨테이너 관리 문제로 컨테이너가 실제로 사용하고 있는 자원과 Backend.AI에서 출력되는 자원 점유량이

일치하지 않는 경우가 있을 수 있습니다. 그 때 RECALCULATE USAGE 버튼을 클릭하면 자원 점유량을 수동 보정할 수 있습니다.

- **RESCAN IMAGES:** 등록된 모든 도커 레지스트리에서 이미지 메타 정보를 받고 업데이트 합니다. Backend.AI에서 사용 가능한 이미지를 레지스트리에 새로 등록한 경우 사용할 수 있습니다.

Maintenance

### General

Fix

**Match usage database with current state**

Recalculate usage database with current state.  
Useful if docker fails by accident, or some sessions have been forcibly terminated.

**RECALCULATE USAGE**

Images / Environment

**Rescan image list from repository**

Rescan image list from registered repositories.  
It may take a long time, so please wait after execution.

**RESCAN IMAGES**

**Clean up old images**

Clean up old images from docker image list.

**CLEAN UP IMAGES**

**주석:** 사용하지 않는 이미지를 제거하거나, 주기적 관리 일정 등록 등 기타 관리에 필요한 설정이 계속 추가될 예정입니다.

## CHAPTER 11

---

### GUI 콘솔 사용시 문제 해결

---

GUI 콘솔을 오래 사용하다 보면 세션 리스트 갱신이나 Jupyter Notebook, 터미널 등의 앱에 연결 상태가 좋지 않은 문제가 발생할 수 있습니다. 이 때 콘솔 페이지를 갱신하면 문제가 사라지는 경우가 많습니다. 콘솔 페이지를 갱신하려면 다음과 같은 명령을 내립니다.

- 웹 기반 콘솔: 브라우저 페이지를 갱신합니다(Ctrl-R 등 브라우저 별 제공하는 페이지 갱신 단축키 사용). 브라우저의 캐시가 쌓여 오동작 하는 경우도 있으므로 캐시를 사용하지 않고 페이지 갱신을 해보는 것도 좋습니다(Shift-Ctrl-R 등 브라우저 별 상이).
- 앱 기반 설치형 콘솔: Ctrl-R 단축키를 클릭하여 앱 페이지를 갱신할 수 있습니다.

#### 11.1 SFTP 연결이 끊기는 경우

콘솔 앱을 통한 SFTP 연결은 콘솔 앱이 내장하고 있는 로컬 proxy 서버를 사용합니다. SFTP 연결 후 파일 전송하는 과정에서 콘솔 앱을 종료하면 로컬 proxy 서버로 같이 종료되므로 파일 전송이 중간에 실패하게 됩니다. 따라서, 세션을 사용하지 않는다고 해도 SFTP 사용



중에는 콘솔 앱을 종료하면 안 됩니다. 만약 페이지 갱신이 필요한 상황이면 Ctrl-R 단축키를 이용하는 것을 권합니다.

또한, 콘솔 앱을 종료한 후 다시 시작한 경우 기존에 존재하던 컨테이너에서 SFTP 서비스를 자동으로 시작하지 않습니다. 명시적으로 원하는 컨테이너에서 SSH / SFTP 서비스를 시작해줘야 SFTP 연결을 맺을 수 있습니다.

## 11.2 세션이 실제 사용중인 자원과 표시되는 자원이 불일치하는 경우

**주석:** 이 기능은 super admin만 사용할 수 있습니다.

간혹 네트워크 접속 및 도커 데몬 자체의 컨테이너 관리 문제로 세션이 실제로 사용 중인 자원과 콘솔 앱에서 표시되는 자원 점유량이 어긋나는 경우가 있을 수 있습니다. 만약 이런 문제가 발생한 경우 Maintenance 페이지로 이동하여 RECALCULATE USAGE 버튼을 클릭해서 수동 보정할 수 있습니다.

## 11.3 도커 레지스트리에 이미지 등록 후 세션 생성 환경에 이미지가 보이지 않을 때

**주석:** 이 기능은 super admin만 사용할 수 있습니다.

사실 도커 레지스트리에 이미지가 새로 등록된 경우 Backend.AI에서 레지스트리 별 이미지 메타 데이터를 업데이트 해야 세션 생성할 때 이용할 수 있습니다. 메타 데이터 업데이트는 Maintenance 페이지의 RESCAN IMAGES 버튼을 클릭하여 수행할 수 있습니다. 만약

연결된 도커 레지스트리가 여러 개일 경우, RESCAN IMAGES 버튼을 클릭하면 모든 레지스트리에서 메타 정보를 받아 옵니다.

특정한 도커 레지스트리의 메타 정보만 업데이트 하고자 할 경우 Environments 페이지의 Registries 탭에서 원하는 레지스트리의 정보만 갱신할 수 있습니다. 원하는 레지스트리의 Controls 열에서 리프레시 버튼을 클릭하면 됩니다. 휴지통 아이콘을 클릭하여 레지스트리를 삭제하지 않도록 주의해야 합니다.

## CHAPTER 12

### 사용권 계약 조건

#### 12.1 Backend.AI 라이선스 (소프트웨어)

이 문서는 Backend.AI 소프트웨어에 관한 사용권 계약 조건을 정의합니다. 래블업에서 제공하는 Backend.AI Cloud 서비스의 사용료 및 지원 플랜은 이 정책과 별도입니다.

Backend.AI의 서버 컴포넌트들(이하 “Backend.AI 서버”)은 GNU Lesser General Public License v3.0(이하 “LGPL”)으로 배포되며, Backend.AI 서버에 접속하기 위한 API Client 라이브러리들 및 보조 컴포넌트들(이하 “Backend.AI 클라이언트”)은 MIT 라이선스로 배포됩니다. LGPL을 준수하는 경우에도 Backend.AI 서버를 이용한 수익 활동을 하는 경우 조건에 따라 래블업 주식회사(“래블업”)와의 상용 계약이 필요합니다. Backend.AI 엔터프라이즈 솔루션을 대상으로 하는 추가적인 몇몇 플러그인 및 관리 허브는 오픈소스가 아니며, 상용 소프트웨어입니다.

##### 용어 정의

- 하드웨어 : 사용자가 자기 소유 또는 임대한 물리적 컴퓨터와 소프트웨어 실행 권한을

가지는 가상머신 및 컨테이너 환경을 포함합니다.

- 조직 : 개인, 법인, 단체, 기관 (비영리·영리 모두 포함; 단, 별도 법인인 자회사는 포함하지 않습니다)

사용자가 Backend.AI Server (Manager / Agent / Common) 를 사용 및 변경하거나 이를 이용하는 소프트웨어를 개발 및 배포하는 경우 LGPL을 따라야 합니다. 아래는 이때 LGPL에 따른 의무가 없는 경우에 대한 예시입니다:

1. Backend.AI 서버를 변경하지 않고 모듈로 불러와서(예: Python import) 사용한 소프트웨어를 배포하는 경우.
2. Backend.AI 서버를 하드웨어에 설치하여 일반 공중이 네트워크를 통해 사용하는 경우.

이 외의 모든 경우에 대한 정확한 해석은 LGPL 원문과 법원의 판단을 따릅니다.

LGPL 준수와 별개로, 다음과 같은 경우에는 래블업과 상용 계약을 맺어야 합니다:

1. Backend.AI 서버를 설치해야만 동작하는 소프트웨어를 조직 외부 고객에게 판매하는 경우.
2. Backend.AI 서버를 포함한 하드웨어를 조직 외부 고객에게 판매하는 경우.
3. Backend.AI 서버를 하드웨어에 설치하고 이를 사용하는 조직 외부 고객으로부터 사용료를 받는 경우.

그 외의 경우에는 Backend.AI 서버를 무료로 사용할 수 있습니다.

### 해석 예시

- Backend.AI 서버를 변경하여 조직 외부로 배포한다면 코드를 공개하고 반드시 LGPL을 똑같이 적용해야 합니다. 조직 내부적으로만 사용한다면 코드를 공개할 의무가 없습니다.
- Backend.AI 서버를 필수 라이브러리로 사용한 소프트웨어를
  - 무료 배포하는 경우: 해당 소프트웨어는 (L)GPL이 아니어도 되고, 래블업과의 별도 계약도 필요하지 않습니다.

- 유료 배포하는 경우: 해당 소프트웨어는 (L)GPL이 아니어도 되지만, 래블업과의 상용 계약이 필요합니다.
- Backend.AI 서버를 하드웨어에 설치하고 이를 네트워크를 통해 일반 공중이
  - 무료로 사용하게 하는 경우: 래블업과의 별도 계약이 필요하지 않습니다.
  - 유료로 사용하게 하는 경우: 래블업과의 상용 계약이 필요합니다.
- Backend.AI 서버를 설치한 하드웨어를
  - 무료로 사용하게 하는 경우: 래블업과의 별도 계약이 필요하지 않습니다.
  - 유료로 사용하게 하는 경우: 래블업과의 상용 계약이 필요합니다.

상용 계약에는 기본적으로 엔터프라이즈 버전의 월간/연간 구독 사용료가 포함되지만, 개별 계약에 따라 세부 내용은 다를 수 있습니다. 오픈소스 무료 버전 사용자들도 유지보수 및 지원 플랜을 별도로 구입할 수 있습니다.

## CHAPTER 13

---

### 참고 자료

---

- 온라인 API 매뉴얼 : <https://docs.backend.ai>
- 온라인 Client SDK 매뉴얼 : <https://docs.client-py.backend.ai>

이 문서의 최신 버전은 아래 사이트에서 볼 수 있습니다:

- <https://console.docs.backend.ai/en/stable> (영문)
- <https://console.docs.backend.ai/ko/stable> (한글)