



Backend.AI WebUI

User's Guide

(25.05)

Lablup Inc.



목차

0 빠른 시작	2
0.1 목표	2
1 면책조항	4
2 개요	6
2.1 주요 개념	7
2.2 Backend.AI 기능 상세	9
2.3 사용자 역할 별 접근 가능한 메뉴 안내	11
3 설치	13
4 가입 및 로그인	14
4.1 가입	14
4.2 로그인	17
4.3 비밀번호를 잊어버렸을 경우	18
4.4 사이드바의 메뉴	19
5 상단 바 기능	20
5.1 프로젝트 선택기	20
5.2 이벤트 알림	20
5.3 테마 모드	21
5.4 도움말	22
5.5 사용자 메뉴	22
6 시작 페이지	26

7 요약 페이지	27
7.1 자원 사용량	28
7.2 시스템 자원	28
7.3 초대	28
7.4 Backend.AI WebUI 앱 다운로드	28
8 데이터 페이지	30
8.1 데이터 & 폴더 활용하기	30
8.2 폴더 종류	45
9 연산 세션	47
9.1 새로운 세션 시작하기	47
9.2 세션 정보	60
9.3 Jupyter Notebook 사용하기	63
9.4 웹 터미널 활용	67
9.5 연산 세션 로그 조회	68
9.6 실행중인 세션 이름 변경하기	68
9.7 연산 세션 삭제하기	69
9.8 유휴 상태 검사	69
9.9 세션 생성하기 전에 환경 변수를 추가하는 방법	71
9.10 세션 생성하기 전에 사전 개방 포트를 추가하는 방법	72
9.11 세션 커밋 저장하기	74
9.12 진행 중인 세션의 변환된 이미지 활용	75
9.13 웹 터미널 고급 사용법	76
10 연산 세션에 폴더 마운트	79
10.1 자동 마운트 폴더를 이용하여 연산 세션 환경 구성하기	82
11 저장 폴더 공유 및 접근 제어	84
11.1 개인 스토리지 폴더를 다른 사용자와 공유하기	84
11.2 공유 폴더 권한 조정	89
12 모델 서빙	90
12.1 모델 서비스	90
12.2 23.03 및 이전 버전에서 모델 서비스를 구성하는 방법과 한계	91
12.3 모델 서비스를 사용하기 위한 단계 안내	91
13 채팅 페이지	112
13.1 모델 선택하기	113
13.2 채팅 카드 추가 및 삭제	114
13.3 채팅 기록 삭제	114

13.4 입력 연동	115
14 Jupyter notebook과 웹 기반 Git 저장소 불러오기 및 실행	116
14.1 Jupyter notebook 파일 가져오기 및 실행	116
14.2 Jupyter notebook을 바로 실행할 수 있는 버튼 생성하기	119
14.3 GitHub 저장소 불러오기	120
14.4 GitLab 저장소 불러오기	120
15 나의 실행 환경	122
16 자원 요약	124
17 Statistics 페이지	125
18 연산 세션에 SSH/SFTP 접속하기 (Linux/Mac/Visual Studio Code)	127
18.1 Linux / Mac 환경	128
18.2 Windows / FileZilla 사용 환경	131
18.3 For Visual Studio Code	133
18.4 Backend.AI Client 패키지를 통해 SSH 접속하는 방법	137
19 사용자 설정 페이지	143
19.1 GENERAL 탭	145
19.2 LOGS 탭	152
20 Backend.AI 클러스터 연산 세션	155
20.1 Backend.AI 클러스터 연산 세션 개괄	155
20.2 Backend.AI 클러스터 연산 세션의 사용	157
20.3 컨테이너 별 로그 확인하기	160
21 관리자 기능	162
21.1 새로운 사용자 생성 및 정보 갱신	163
21.2 사용자 계정 비활성화	167
21.3 사용자의 키페어 관리	168
21.4 프로젝트 Storage 폴더를 다른 사용자와 공유하기	170
21.5 모델 카드 관리	172
21.6 자원 정책 관리	176
21.7 이미지 관리	185
21.8 도커 레지스트리 관리	188
21.9 자원 프리셋 설정	190
21.10 Agent 노드 관리	192
21.11 자원 그룹 관리	194

21.12 저장소	198
21.13 세션 자원 다운로드	202
21.14 시스템 설정 조회	203
21.15 서버 관리 메뉴	206
21.16 상세 정보	206
22 FAQ 및 문제 해결	208
22.1 사용자 문제 해결 가이드	208
22.2 관리자 문제 해결 가이드	210
23 Appendix	212
23.1 GPU 가상화를 통한 컨테이너 별 GPU 분할 할당	212
23.2 GUI 를 통한 자원 모니터링 및 스케줄링 자동화	216
23.3 Multi-version 머신러닝 컨테이너 지원	218
23.4 실행 중인 연산 세션을 새로운 사용자 이미지로 변환하는 방법	223
23.5 Backend.AI 서버 설정 가이드	224
23.6 통합 예제	225
24 사용권 계약 조건	230
24.1 Backend.AI 라이선스 (소프트웨어)	230
25 참고 자료	232

User's guide for the Backend.AI WebUI.

Backend.AI WebUI is a web or app that provides easy-to-use GUI interface to work with the Backend.AI server.

이 문서의 최신 버전은 아래 사이트에서 볼 수 있습니다:

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

CHAPTER 0

빠른 시작

Backend.AI WebUI를 사용하는 빠른 시작 가이드입니다. 이 튜토리얼에서는 다른 지식기반이 없이도 Backend.AI 를 사용할 수 있도록 꼭 필요한 내용만을 다룹니다.

0.1 목표

0.1.1 파트 1. Backend.AI 기초 사용 가이드

- 가상 폴더 만드는 방법
- 세션 생성하는 방법
- 세션 사용하는 방법
- 세션 종료하는 방법

0.1.2 파트 2. Backend.AI 고급 사용 가이드

- tmux 를 사용한 terminal 사용법
- 자동마운트되는 가상폴더를 사용해서 pip 패키지를 설치하는 방법
- Backend.AI에서 SSH/sFTP 을 사용하는 방법
- APT 패키지 설치하는 방법

CHAPTER 1

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 사용함으로 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.

문서 버전: v25.05.250508

최종 갱신 일자 (년/월/일): 2025/05/08

CHAPTER 2

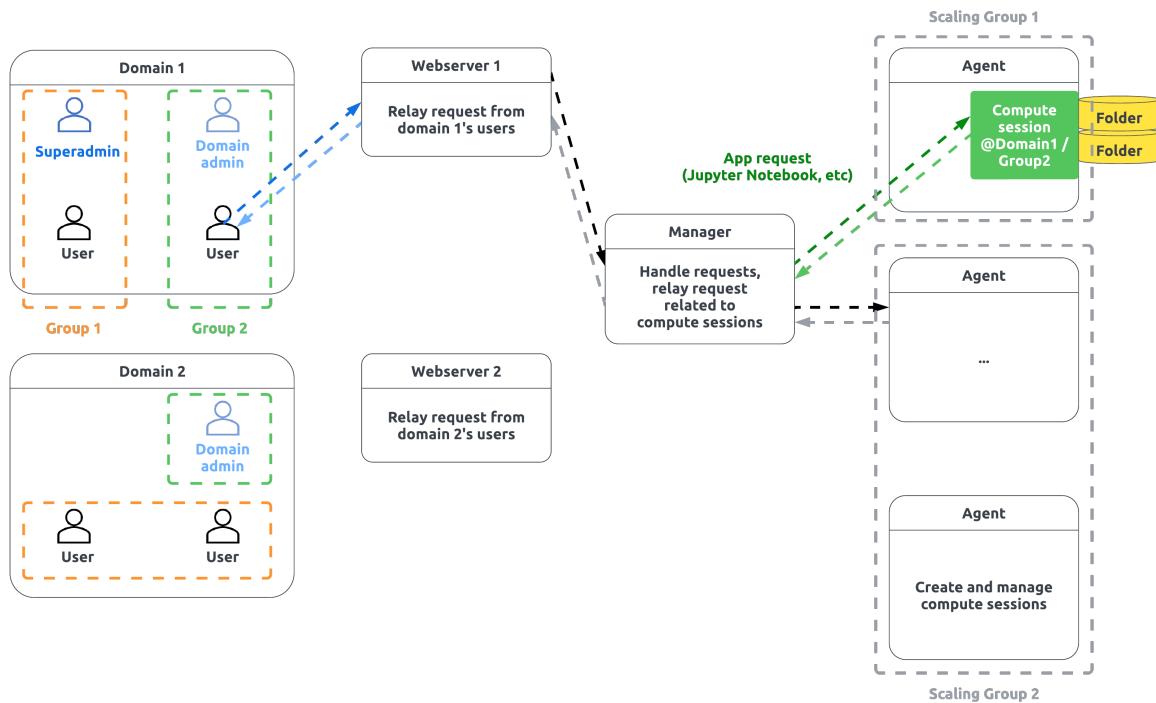
개요

Backend.AI 는 오픈소스 클라우드 자원 관리 플랫폼입니다. Backend.AI 는 클라우드 또는 온-프레미스 클러스터 환경에서 연산 자원을 효율적으로 관리하며 언제 어디서나 on-demand 로 계산에 필요한 가상화된 연산 환경을 제공합니다. 또한, Backend.AI 의 컨테이너 기반 GPU 분할 가상화 기술을 통해 하나의 물리 GPU 를 유연하게 나누어 여러 사용자가 동시에 사용할 수 있도록 지원하고 있어, GPU 의 효율적인 활용을 돋습니다.

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 Control-Panel 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 Web-UI는 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.

2.1 주요 개념



- 사용자: 사용자는 Backend.AI에 접속하여 작업을 수행하는 주체입니다. 사용자는 가지는 권한에 따라 일반 사용자, 도메인 어드민(domain admin), 수퍼 어드민(superadmin)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 프로젝트에 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 프로젝트 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.

- 프로젝트: 도메인 하위에 속하는 계층으로, 한 도메인에는 여러 개의 프로젝트가 존재할 수 있습니다. 프로젝트는 하나의 작업 프로젝트 단위라고 생각할 수도 있습니다. 사용자는 한 도메인 내에 존재하는 여러 개의 프로젝트에 동시에 속할 수 있습니다. 연산 세션은 반드시 하나의 프로젝트에 속해야 하며, 사용자는 자신이 속한 프로젝트 내에서만 세션을 생성할 수 있습니다. 도메인 관리자는 도메인 내에 속한 프로젝트의 정책을 설정하거나 세션을 관리할 수 있습니다. 예를 들어, 프로젝트 내에서 사용할 수 있는 총 자원량을 설정할 경우, 프로젝트 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 실행되는 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 Web-UI를 이용하여 원하는 세션의 데몬에 바로 접속할 수 있습니다.
- Web-UI: 웹 (web) 형태로 서비스 되는 GUI 클라이언트. Backend.AI 서버의 주소를 지정하고 사용자 계정 정보를 입력해 로그인 한 후 서비스를 사용할 수 있습니다.
- 로컬 wsproxy: Web-UI 앱에서 내장하고 있는 프록시 서버. 콘솔 앱에서 사용 가능한 Jupyter Notebook, Terminal 등의 서비스는 websocket을 통해 서버와 통신하는데, Web-UI 앱과 주고받는 일반적인 HTTP 요청을 websocket 형태로 변환하여 메시지를 전달해 주는 역할을 합니다. Web-UI 앱과 wsproxy 와의 연결이 끊기거나 wsproxy 서버가 죽은 경우 Notebook, Terminal 등의 서비스 접속이 불가능합니다.
- 웹 wsproxy: 웹 형태로 제공되는 Web-UI의 경우 브라우저의 특성상 별도 내장 서버를 운영할 수 없습니다. 이 경우 wsproxy 서버를 별도의 웹 서버로 독립 운영하여 Web-UI 앱에서 웹 wsproxy를 바라보게 만듦으로써 Jupyter Notebook, Terminal 등과 같은 서비스를 웹 환경에서도 사용할 수 있습니다.

2.2 Backend.AI 기능 상세

Category	Feature
NVIDIA support	Container-level multi GPU
	(Enterprise) Fractional GPU sharing & scaling
	Multiple CUDA library version support (8.0 to 12.2)
	GPUDirect storage support
	NGC (NVIDIA GPU Cloud) image integration
NPU Support	GPU-GPU Network autoconfiguration
	<ul style="list-style-type: none"> • Graphcore IPU • Rebellions ATOM • FuriosaAI Warboy • Hyperaccel LPU
	On-premise installation on both bare-metal / VM
	Hybrid cloud (on-premise + cloud)
	Polycloud (multi-cloud federation)
Scheduling	Unified scheduling & monitoring with GUI admin
	Per-user (keypair) resource policy
	(Enterprise) Per-project resource policy
	Availability-slot based scheduling
	(Enterprise) Utilization based resource management
Cluster partitioning	Resource groups by H/W spec and usage
	(Enterprise) Access control of users to resource group
	(Enterprise) Access control of project to resource group
	Sandboxing via hypervisor/container

다음 페이지에 계속

표 1 – 이전 페이지에서 계속

Category	Feature
UI / UX	Access logs for each user
	Per session (container) logs
	GUI web interface
	(Enterprise) Admin GUI web interface
Data management	(Enterprise) MLOps pipeline web interface
	EFS, NFS, SMB and distributed file system (CephFS, GlusterFS, HDFS, etc)
	(Enterprise) Storage solution integration:
	<ul style="list-style-type: none"> • PureStorage FlashBlade • NetApp ONTAP & NetApp XCP • Weka.io • Dell EMC • IBM SpectrumScale (GPFS) • Vast Data
Developer support	Fine-grained Access control to data by user/project
	Per user/project based storage quota management*
	Universal programming languages (Python, C/C++, etc)
	Interactive web apps (Terminal, Jupyter, VSCode, MLFlow, Microsoft NNI, R Studio, etc)
For data scientists	Offering various type of computing sessions tailored to specific purposes
	<ul style="list-style-type: none"> • Interactive • Batch • Inference
	Concurrent user of multiple versions of libraries: Tensorflow, PyTorch, etc
	Concurrent user of multiple versions of libraries

다음 페이지에 계속

표 1 – 이전 페이지에서 계속

Category	Feature
Customer support (Enterprise)	Periodic update of ML libraries
	On-site installation (bare-metal / VM)
	Configuration support (on-premise + cloud)
	Admin/user training
	Support for updating to latest version
	Priority development and escalation
	Customized container image / kernel or kernel repository

- ONLY Available on storage supports directory quota management

2.3 사용자 역할 별 접근 가능한 메뉴 안내

참고:

- * 표시가 붙은 페이지는 관리자용 메뉴입니다.
 - 관리자 전용 기능은 [어드민 전용 메뉴](#)에서 확인할 수 있습니다.
-

페이지 \ 역할	일반 사용자	어드민
요약 (Summary)	O	O
세션 (Sessions)	O	O
모델 서빙 (Serving)	O	O
가져오기 및 실행하기 (Import & Run)	O	O
데이터 & 폴더 (Data & Storage)	O	O
자원 요약 (Agent Summary)	O	O
통계 (Statistics)	O	O
사용자 (Users)*	X	O
실행 환경 (Environments)*	X	O
자원 (Resources)*	X	O
환경 설정 (Configurations)*	X	O
관리 (Maintenance)*	X	O
정보 (Information)*	X	O
사용자 설정 (UserSettings)	O	O

CHAPTER 3

설치

Backend.AI WebUI can be used in two different forms. It can be used as a web service by connecting to a separate web address provided by the admins, or as a form of app provided as a stand-alone executable file which does not require separate installation.

With web-based form of WebUI, users just need the latest versions of web browsers, no need to install platform-dependent desktop applications.

- Recommended browser: Latest version of Chrome (at least > 80)
 - Requirement: Any machine that runs web browser (2 cores, 4 GiB memory)
-

참고: We do not support Microsoft Internet Explorer since it is deprecated and does not follow web standard, and does not support up-to-date browser features.

The stand-alone WebUI app can be downloaded from following link: <https://github.com/lablup/backend.ai-webui/releases>

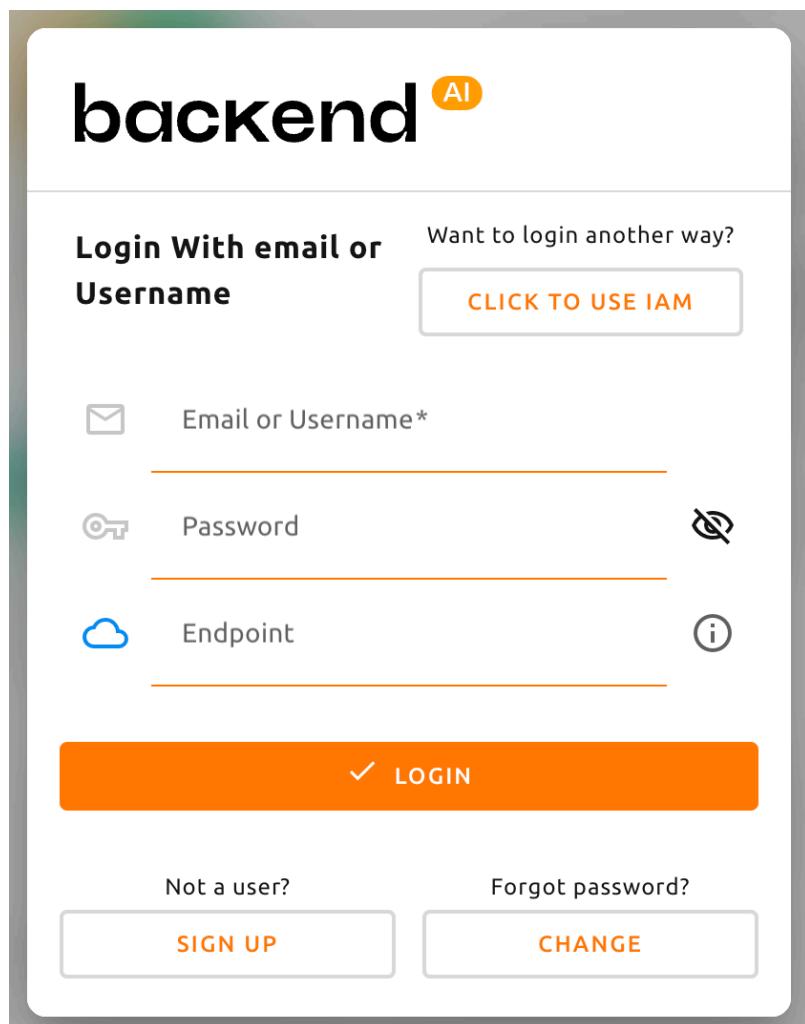
참고: Depending on the security settings of the Operating System, our installer may be recognized as an unsigned executable file. Permission check may be required.

CHAPTER 4

가입 및 로그인

4.1 가입

WebUI를 실행하면 다음과 같은 로그인 대화창이 뜹니다. 아직 가입을 하지 않은 경우에는 가입하기 버튼을 눌러주세요.



이메일과 사용자 이름, 비밀번호 등의 정보를 입력하고 약관과 개인정보보호정책을 읽고 동의한 뒤 가입하기 버튼을 클릭합니다. 시스템 설정에 따라 가입하기 위해 별도의 초대 토큰을 입력 해야 할 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.

The screenshot shows a modal window titled "Signup (Beta invitation only)". It contains fields for Email*, UserName, Invitation Token*, Password*, and Verify Password*. There is also a checkbox for accepting terms and conditions. An orange "SIGNUP" button at the bottom right is labeled with a checkmark.

Email*

Email address is required.

UserName

Invitation Token*

Password*

Verify Password*

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

✓ SIGNUP

참고: 서버 설정이나 플러그인 설정에 따라 사용자에 의한 가입이 막혀있을 수 있습니다. 이 경우에는 시스템 관리자에게 문의 하십시오.

참고: 악성 사용자가 다른 사용자의 비밀번호를 추측하기 어렵게 만들기 위해, 비밀번호는 8 자 이상, 알파벳/특수문자/숫자를 1 개 이상 포함해야 합니다.

4.2 로그인

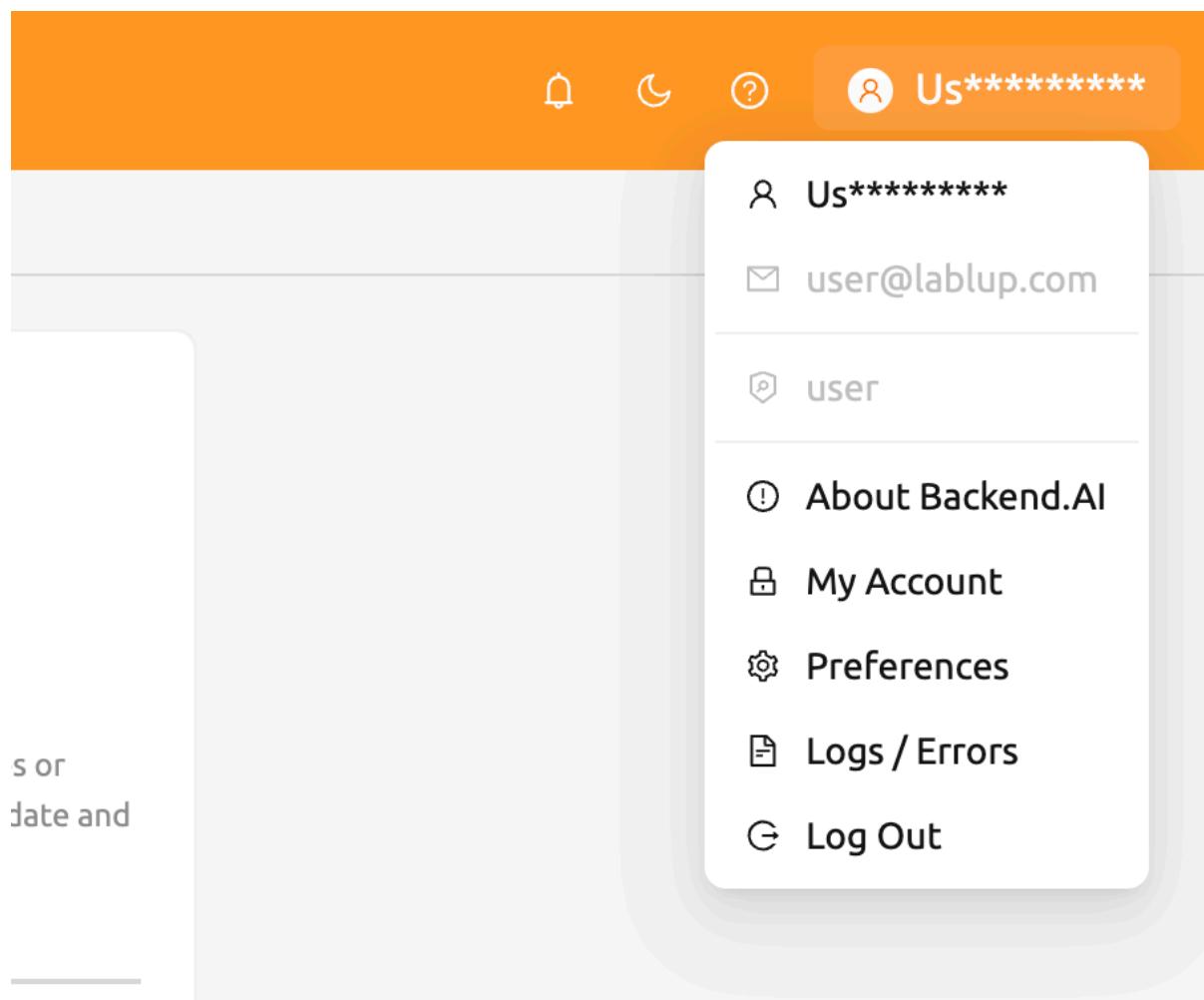
ID와 암호를 입력하고 로그인 버튼을 눌러주세요. 엔드포인트에는 Manager로의 연결을 중계하는 Backend.AI Webserver가 작동하고 있는 URL을 넣어주시면 됩니다.

참고: Webserver의 설치 및 설정 환경에 따라, 엔드포인트가 고정되어 있을 수 있습니다.

참고: Backend.AI는 사용자의 비밀번호를 단방향 해시를 통해 안전하게 보관하고 있습니다. BSD의 기본 암호 해시인 BCrypt를 사용하고 있어, 서버 관리자도 사용자의 비밀번호를 알 수 없습니다.

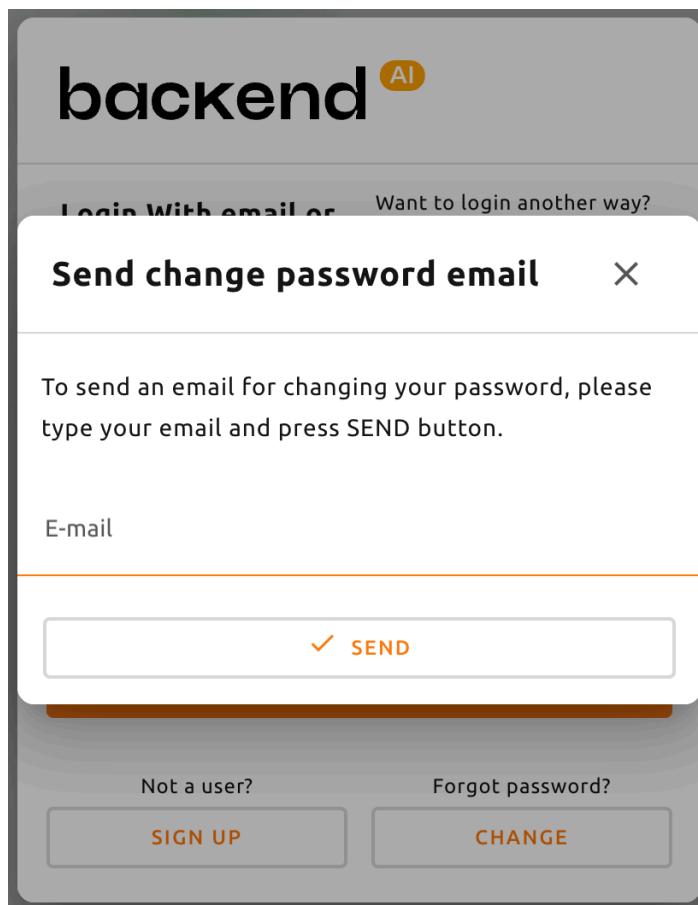
로그인이 완료되면 요약 탭에서 현재 사용하고 있는 자원량 등의 정보를 확인할 수 있습니다.

우측 상단의 아이콘을 클릭하면 하위 메뉴가 나옵니다. 로그아웃 버튼을 클릭해서 로그아웃 할 수 있습니다.



4.3 비밀번호를 잊어버렸을 경우

비밀번호를 잊어버렸을 경우, 로그인 패널의 비밀번호 재설정 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.

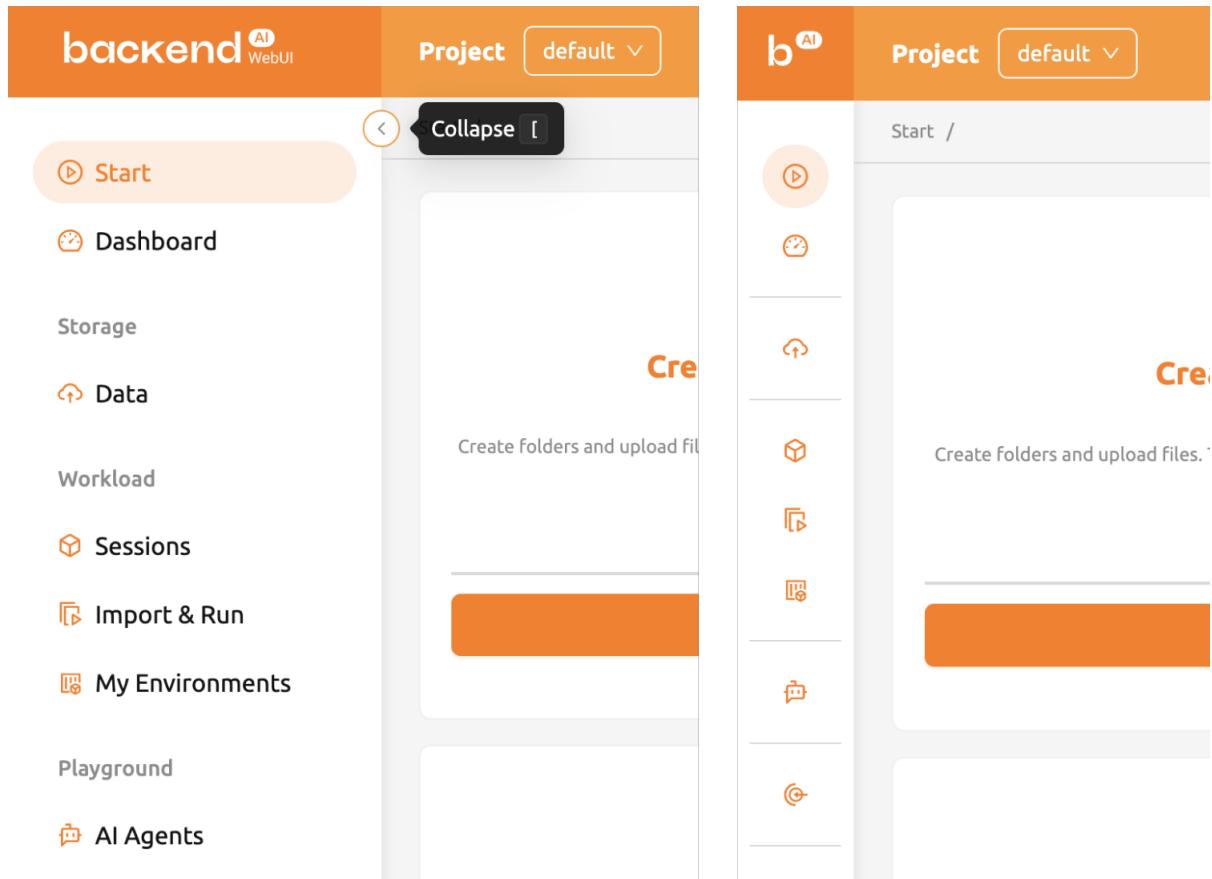


참고: 이 기능도 모듈화되어 있어 어떤 환경에서는 지원되지 않을 수 있습니다.

경고: 로그인 실패가 10 회 이상 발생하면 보안상의 이유로 로그인 시도가 20 분 간 제한됩니다. 만약 20 분 후에도 로그인 제한이 계속 유지되는 경우에는 시스템 관리자에게 문의 하십시오.

4.4 사이드바의 메뉴

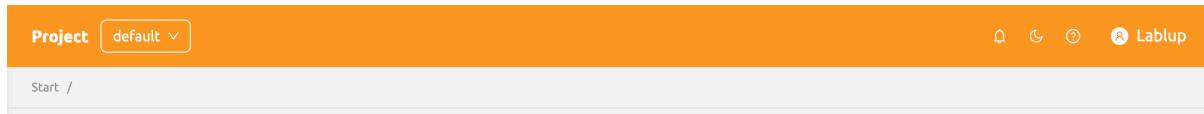
사이드바의 우측에 위치한 버튼을 통하여 사이드바의 크기를 변경합니다. 클릭하면 사이드바의 폭이 크게 줄어들면서 내용을 보다 넓게 볼 수 있습니다. 다시 한 번 클릭하면 사이드바의 폭이 다시 원래대로 돌아옵니다. 또한, 단축키 ([)를 사용하여 사이드바의 너비를 변경할 수도 있습니다.



CHAPTER 5

상단 바 기능

상단 바에는 WebUI 사용을 지원하는 다양한 기능이 포함되어 있습니다.

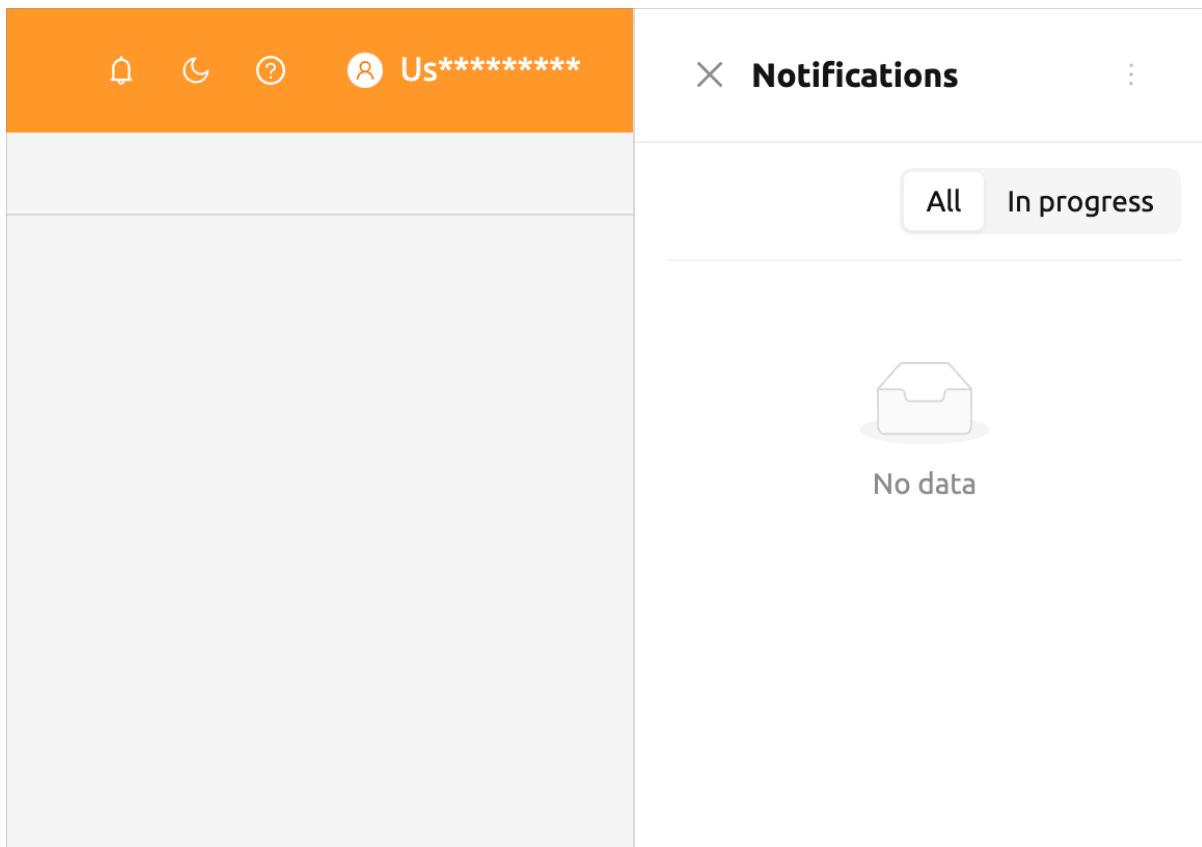


5.1 프로젝트 선택기

사용자는 상단 바의 프로젝트 선택기를 통하여 사용자의 현재 프로젝트를 선택할 수 있습니다. 각 프로젝트 별로 다른 자원 정책을 가질 수 있으므로, 프로젝트를 변경할 경우 가용 가능한 자원 정책이 변경될 수 있습니다.

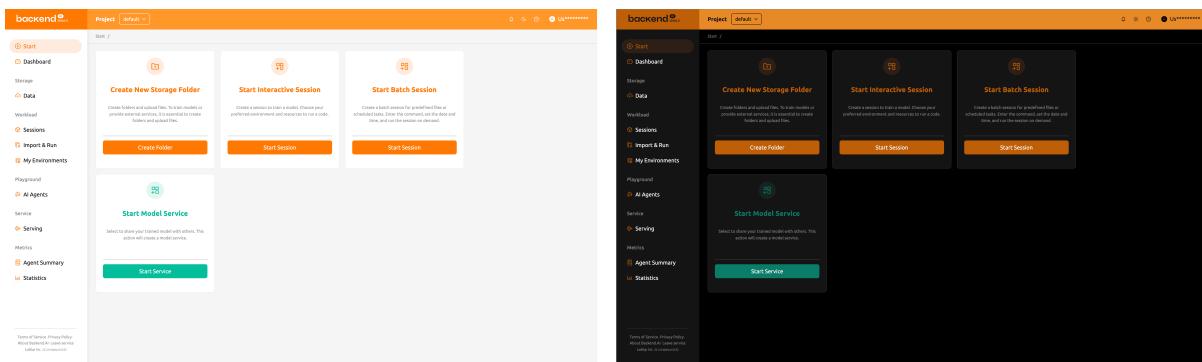
5.2 이벤트 알림

종 모양의 버튼은 이벤트 알림 버튼입니다. WebUI 동작 과정에서 기록이 필요한 이벤트 내역이 여기에 표시됩니다. 예를 들어, 연산 세션의 생성을 위한 백그라운드 작업이 진행되고 있는 경우, 이벤트 알림에서 해당 내용을 확인할 수 있습니다.



5.3 테마 모드

상단 바 우측에 있는 다크모드 버튼을 통하여 WebUI의 테마를 변경할 수 있습니다.

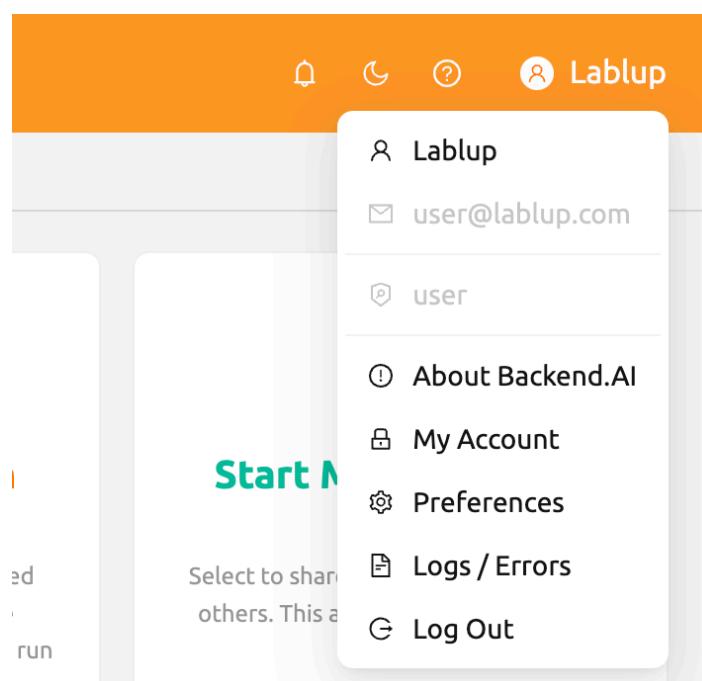


5.4 도움말

상단 바 우측의 물음표 버튼을 통하여, 본 가이드 문서의 웹 버전에 접속할 수 있습니다. 현재 사용자가 접근해 있는 페이지에 따라, 관련된 문서로 자동 연결됩니다.

5.5 사용자 메뉴

상단 바 우측의 사람 아이콘 버튼을 클릭하여, 사용자 메뉴를 확인할 수 있습니다. 각 항목은 다음과 같은 기능을 가집니다.



- Backend.AI에 대하여: Backend.AI Web-UI의 버전, 라이선스 종류 등과 같은 정보를 표시합니다.
- 사용자 정보 변경: 현재 로그인된 사용자 정보를 확인/변경합니다.
- 설정: 사용자 설정 페이지로 이동합니다.
- 로그 / 에러기록: 로그 페이지로 이동합니다. 클라이언트 측에 기록된 로그 및 오류 내역을 확인할 수 있습니다.

5.5.1 사용자 정보 변경

사용자 정보 변경을 클릭하면, 다음과 같은ダイアル로그가 나타납니다.

The dialog box is titled 'My Account Information'. It contains the following fields:

- Full Name: A text input field containing 'Lablup'.
- Original password: A password input field with a visibility icon (eye) to its right.
- New password: A password input field with a visibility icon (eye) to its right.
- New password (again): A password input field with a visibility icon (eye) to its right.
- 2FA Enabled: A toggle switch that is currently off.

At the bottom right of the dialog are two buttons: 'Cancel' and 'Update' (highlighted in orange).

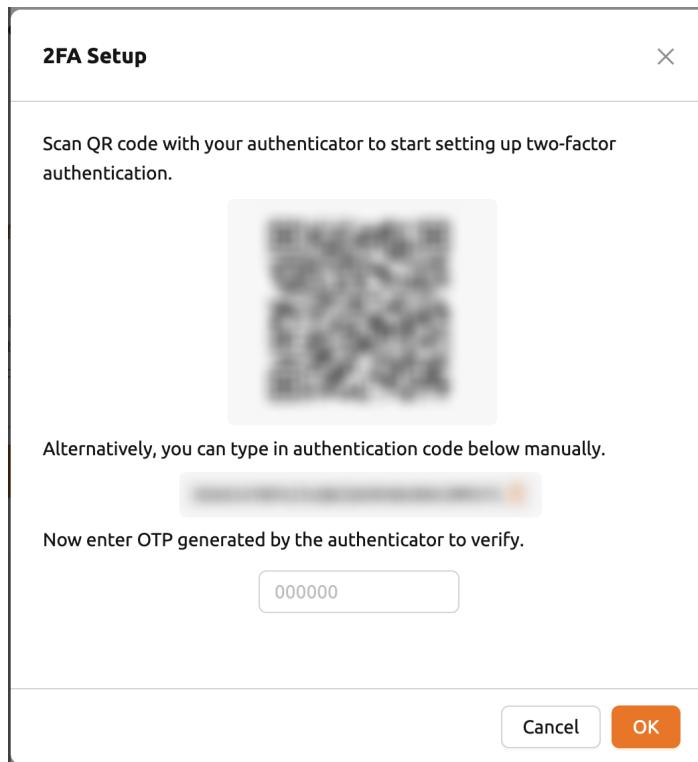
각 항목은 다음과 같은 의미를 가집니다.

- 사용자 이름: 사용자의 이름 (최대 64 자).
- 기존 비밀번호: 원래 비밀번호. 우측 보기 버튼을 클릭하면 입력 내용을 볼 수 있습니다.
- 새 비밀번호: 새로운 비밀번호 (영문자, 숫자, 기호가 1 개 이상 포함 된 8 글자 이상).
- 이중 인증 사용: 이중 인증(2FA) 사용 여부. 이중 인증이 활성화 되어 있으면 로그인 시 OTP 코드를 반드시 입력해야 합니다.

참고: 플러그인 설정에 따라 이중 인증 사용 항목은 제공되지 않을 수 있습니다. 사용을 원하시는 경우, 시스템 관리자에게 문의하세요.

5.5.2 이중 인증 설정

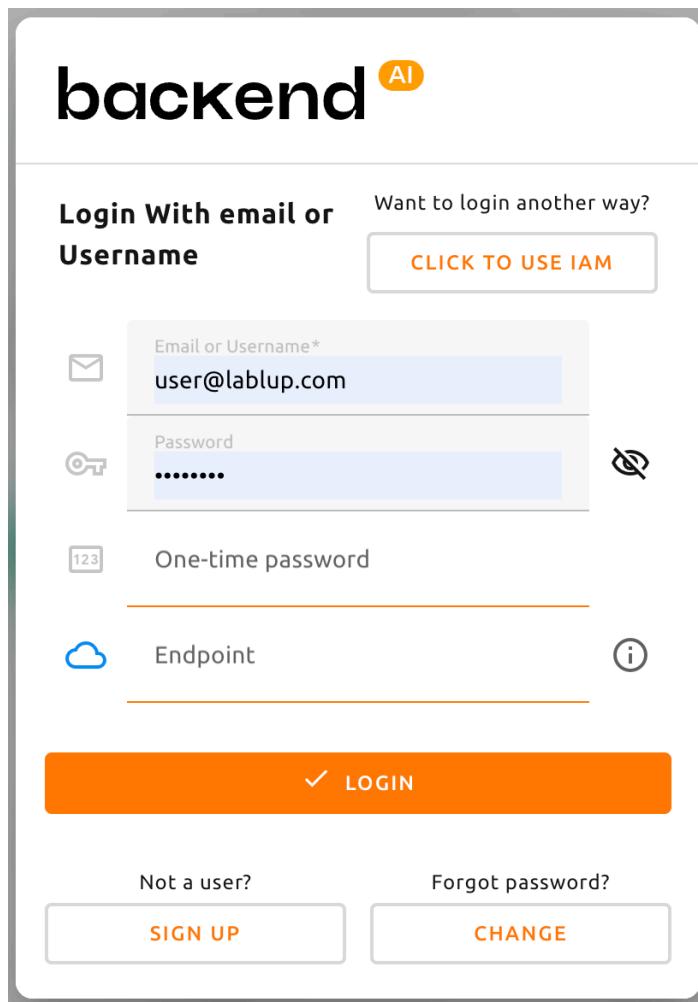
이중 인증 사용 스위치를 켜면 다음과 같은ダイ얼로그가 나옵니다.



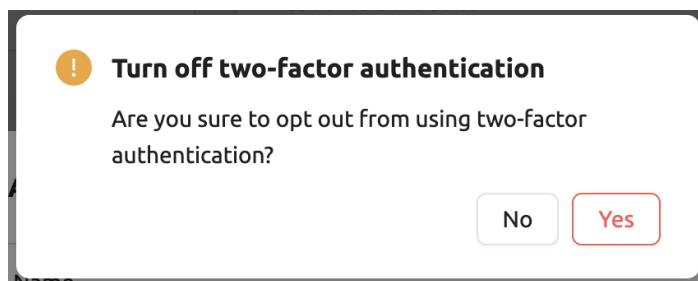
사용자가 사용하는 이중 인증 애플리케이션을 켜고 QR 코드를 스캔하거나 인증 코드를 직접 입력합니다. 이중 인증 지원 애플리케이션은 Google Authenticator, 2STP, 1Password, Bitwarden 등이 있습니다.

이중 인증 애플리케이션에 추가된 항목의 6자리 코드를 위 디아얼로그에 입력합니다. 확인 버튼을 누르면 이중 인증 활성화가 완료됩니다.

이후 해당 사용자의 로그인 과정에서 OTP 코드를 묻는 추가 필드가 나타납니다.



이중 인증 애플리케이션을 열고 One-time password 필드에 6자리 코드를 입력해야 로그인이 가능합니다.

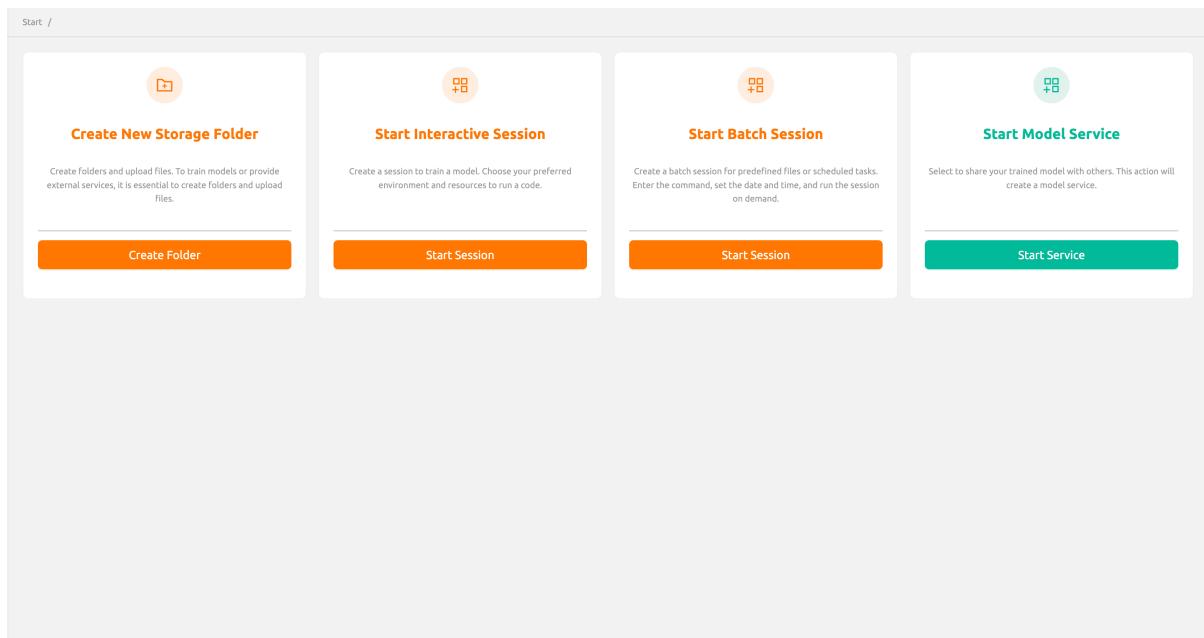


이중 인증을 비활성화하고 싶다면, 이중 인증 사용 스위치를 끄고 위와 같은ダイ얼로그에서 확인 버튼을 누르세요.

CHAPTER 6

시작 페이지

시작 페이지에서는 자주 사용되는 기능에 대한 바로가기를 제공합니다. 현재는 스토리지 폴더 생성, Interactive/Batch 세션 생성, 모델 서비스 생성 기능이 제공됩니다.



참고: 서버 설치 및 설정 환경에 따라, 모델 서비스 기능이 비활성화되어 있을 수 있습니다. 모델 서비스 사용을 원하시는 경우, 시스템 관리자에게 문의하십시오.

CHAPTER 7

요약 페이지

Summary 페이지에서는 사용자의 자원 사용 상태 및 연산 세션 현황 등의 정보를 확인할 수 있습니다.

The screenshot displays the 'Summary' page with three main sections:

- Resource Statistics**: Shows resource usage for the 'default' group. It includes metrics for CPU (0/0 Cores, 0/11 Cores), RAM (0/0 GiB, 0/31 GiB), and Sessions (0/5). A legend indicates that blue dots represent the 'Current Resource Group (default)' and green dots represent 'User Resource Limit'. The CPU section shows 0% usage for both cores and cores used.
- System Resources**: Displays 0 active sessions.
- Invitation**: States that there are no invitations to display.

Below the main sections, there is a 'Download Desktop App' section with a dropdown menu set to 'MacOS' and two download buttons for 'ARM64' and 'X64'.

7.1 자원 사용량

사용자가 할당 가능한 총 자원의 양과 현재 할당받아 점유 중인 자원의 양을 보여줍니다. 사용자의 CPU, 메모리, GPU 자원 점유량 및 할당량을 각각 확인할 수 있습니다. 또한, Sessions 슬라이더에서 사용자가 동시에 생성 가능한 연산 세션의 수 및 현재 작동 중인 연산 세션의 수를 확인할 수 있습니다.

상단의 Resource Group 필드를 클릭하여 자원 그룹을 변경할 수 있습니다. 자원 그룹은 다수의 Agent 노드가 있을 때, Agent의 일부를 묶어 하나의 워커 노드 그룹으로 설정할 수 있는 기능입니다. 다양한 Agent 노드를 보유하고 있는 경우, 각 자원 그룹 별로 특정 사용자 그룹에 독점 할당하는 등의 설정이 가능합니다. Agent 노드가 한 대인 경우에는 하나의 자원 그룹만 보이는 것이 보통입니다. 자원 그룹을 변경하면, 해당 자원의 자원 보유량에 따라 자원량이 변할 수 있습니다.

7.2 시스템 자원

Backend.AI 시스템에 연결된 Agent 워커 노드의 수와 현재 생성되어 있는 전체 연산 세션의 수를 보여줍니다. Agent 노드의 CPU, 메모리, GPU 실제 사용량 (utilization)도 확인할 수 있습니다. 일반 사용자로 로그인 한 경우에는 자신이 생성한 연산 세션의 숫자만 표시됩니다.

7.3 초대

다른 사용자가 저장 폴더를 공유한 경우 여기에 표시됩니다. 공유 요청을 수락하면 Data & Storage 폴더에서 공유 받은 폴더를 조회하고 접근할 수 있습니다. 접근 권한은 공유한 사용자가 지정한 설정에 따릅니다. 물론, 공유 요청을 거절할 수도 있습니다.

7.4 Backend.AI WebUI 앱 다운로드

Backend.AI WebUI는 데스크톱 앱도 지원합니다. 데스크톱 앱을 사용하면, 데스크톱 앱 전용 기능을 사용할 수 있는데, 그 예로 [연산 세션에 SSH/SFTP 접속하기](#) 가 있습니다. Backend.AI WebUI에서는 아래 운영체제에서 실행 가능한 데스크톱 앱을 지원합니다.

- Windows
- Linux
- Mac

참고: 여러분의 실행 환경 (e.g. 운영체제, 아키텍쳐)과 일치하는 버튼을 클릭하면, 현재 WebUI 버전과 일치하는 데스크톱 앱을 다운로드하게 됩니다. 만일 더 최신 또는 이전 버전의 WebUI 를 데스크톱 앱으로 받고자 할 경우 [여기](#) 를 방문해서 원하는 버전을 다운로드 받을 수 있습니다.

CHAPTER 8

데이터 페이지

8.1 데이터 & 폴더 활용하기

Backend.AI는 사용자의 파일을 보존할 수 있도록 별도의 저장소를 지원합니다. 연산 세션의 기본 폴더는 세션이 삭제되면 같이 삭제되므로, 보존해야 하는 파일이 있을 경우 스토리지 폴더에 저장하는 것이 좋습니다. 스토리지 폴더는 좌측 메뉴의 데이터 탭을 선택하면 조회할 수 있습니다. 데이터 페이지에서는 폴더 이름과 ID, 폴더가 위치한 NFS 호스트 이름, 폴더 접근 권한 등의 정보를 확인할 수 있습니다.

The screenshot shows the Backend.AI WebUI interface for managing data. At the top, there's a summary of storage status: 2/20 created folders, 1 project folder, and 0 invited folders. It also displays quota information for 'Project default' and 'User User Lablup' across storage volumes (local:myceph). Below this, a table lists three folders: 'project-folder', 'test-folder2', and 'test-folder'. Each entry includes a preview icon, controls (trash, edit), status (READY), location (local:volume1), type (Project or User), mount permission (Read & Write), and owner (User). A search bar and pagination controls (Total 3 items, 10/page) are also visible.

스토리지 폴더는 사용자 형태와 프로젝트 형태의 두 가지 종류가 있습니다.

사용자 형태(User)의 경우 일반 사용자가 자신의 폴더를 생성한 것으로, '종류' 열에서 한 명의 사용자가 있는 것으로 확인할 수 있습니다. 프로젝트 폴더(Project)의 경우 '종류' 열에 여러 명의 사용자가 있는 아이콘으로 알아볼 수 있으며, 도메인 관리자에 의해서만 프로젝트 별로 생성되고 일반 사용자는 생성된 프로젝트 폴더를 사용만 할 수 있습니다.

This screenshot is identical to the one above, showing the same storage status, quota information, and folder list. The table shows three entries: 'project-folder', 'test-folder2', and 'test-folder', all categorized under 'User' in the 'Type' column.

저장소 상태와 스토리지 볼륨별 가용량은 아래의 정보를 표시합니다.

• 저장소 상태

- 생성된 폴더: 사용자가 생성 한 폴더 개수.
 - * 생성 제한: 사용자가 현재 만들 수 있는 최대 폴더 수. 이 값은 사용자에게 적용된 자원 정책에 따라 다르며 자원 정책을 변경하지 않고 변경할 수 없음. 사용자가 생성하지 않은 폴더(예 : 공유하도록 초대된 폴더 또는 프로젝트 폴더)는 계산에 포함되지 않음.
- 프로젝트 폴더: 사용자가 생성한 프로젝트 폴더 수.

- 초대된 폴더: 다른 사용자가 초대를 통해 공유한 폴더 수.
- 저장소 볼륨별 가용량
 - 호스트: 스토리지 호스트의 이름.
 - 프로젝트: 현재 프로젝트의 폴더 사용량 / 현재 프로젝트의 폴더 가용량.
 - 사용자: 현재 사용자의 폴더 사용량 / 현재 사용자의 폴더 가용량.

참고: 쿼터(가용량)는 쿼터 설정을 제공하는 스토리지에서만 사용할 수 있습니다(XFS, CephFS, NetApp, Purestorage 등). 가용량 설정에 대해서는 [쿼터 설정 패널](#) 섹션을 참고하십시오.

8.1.1 스토리지 폴더 생성

'폴더 생성' 버튼을 눌러서 폴더를 원하는 이름으로 생성할 수 있습니다. 폴더 이름에 생성할 폴더의 이름을 입력하고, 종류 User/Project 중 하나를 선택합니다. (서버 설정에 따라 User, Project 중 하나만 선택 가능할 수 있습니다.) 프로젝트 폴더를 생성할 경우 프로젝트 선택 필드가 나타납니다. 프로젝트 폴더는 Project 필드에 지정된 프로젝트 전용이 되며, 해당 프로젝트에 속해있는 사용자만 그 프로젝트 폴더를 마운트해서 사용할 수 있습니다. 원하는 대로 값을 설정한 후 생성 버튼을 클릭하여 폴더를 생성할 수 있습니다.

The dialog box has the following fields and options:

- Folder name ***: A text input field with a maximum of 64 characters.
- Location**: A dropdown menu showing "local:volume1".
- Usage Mode**: Radio buttons for "General" (selected) and "Model".
- Type**: Radio buttons for "User" (selected).
- Permission**: Radio buttons for "Read & Write" (selected) and "Read Only".
- Buttons**: "Reset" (red outline), "Cancel", and "Create" (orange).

생성 대화 상자에서 선택할 수 있는 각 필드의 의미는 다음과 같습니다.

- 폴더 이름: 폴더명을 적는 칸으로, 최대 64 자까지 입력 할 수 있음.

- 위치: 폴더를 만들 NFS 호스트. NFS 호스트가 여러 개인 경우 하나를 선택할 수 있음. 선택한 호스트가 충분한 용량을 가지고 있는지 여부는 표시기를 통해 확인할 수 있음.
- 사용 방식: 폴더의 용도를 설정할 수 있음. 모드에는 일반, 데이터 및 모델의 세 가지 유형이 있는데, 향후 Data & Model Store 전용 기능 개발을 위한 분류이며 현재는 목적에 따른 UI 차이가 없음.
- 종류: 생성 할 폴더 유형을 결정함. 사용자(User) 또는 프로젝트(Project)로 설정할 수 있음. 사용자 폴더는 사용자가 단독으로 생성하여 사용할 수 있는 폴더이며 프로젝트 폴더는 관리자가 생성하고 프로젝트의 사용자가 공유하는 폴더를 의미함.
- 프로젝트: 프로젝트 태입을 선택한 경우 보여짐. 새 프로젝트 폴더를 만들 때 폴더가 속한 그룹을 지정할 수 있음. 프로젝트 폴더는 프로젝트에 속해야 하나, 사용자 폴더를 생성할 때 별도의 역할을 수행하지는 않음.
- 권한: 프로젝트 폴더의 권한을 설정함. "Read-Only"로 설정된 경우, 프로젝트 멤버가 이 폴더에 쓰기 작업을 수행할 수 없음.
- 복사 가능 여부: "Model" 사용 방식을 선택할 경우 보여짐. 생성될 vfolder의 복사 가능 여부를 설정할 수 있음.

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

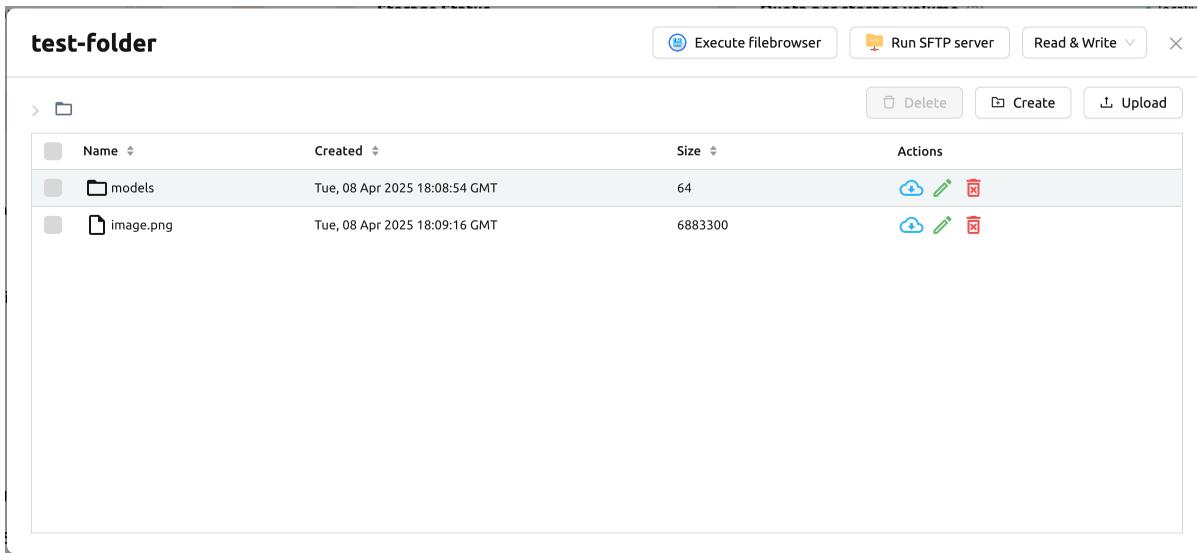
8.1.2 폴더 내용 조회하기

폴더 이름을 클릭하여 해당 폴더의 내용을 조회할 수 있는 파일 탐색기를 띠울 수 있습니다.

Name	Status	Location	Type	Mount Permission	Owner
project-folder	READY	local:volume1	Project	Read & Write	Project
test-folder2	READY	local:volume1	User	Read & Write	User
test-folder3	READY	local:volume1	User	Read & Write	User

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.

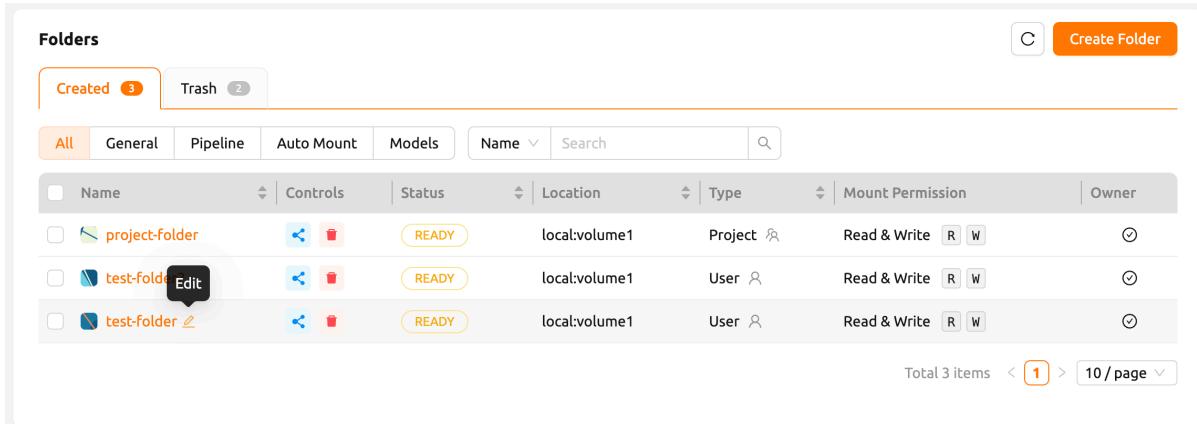


'만들기' 버튼으로 현재 경로 상에 새로운 폴더를 생성할 수 있으며, 'Upload' 버튼으로 현재 경로에 로컬 파일 혹은 폴더를 업로드 할 수도 있습니다. 이런 파일 작업은 앞서 설명한 연산 세션 마운트 방식으로도 모두 수행 가능합니다.

폴더 내 파일 또는 디렉토리의 최대 길이는 호스트 파일 시스템에 따라 달라질 수 있습니다. 그러나 일반적으로 255자를 초과할 수 없습니다.

8.1.3 폴더 이름 변경

스토리지 폴더의 이름을 변경할 수 있는 권한을 가지고 있는 경우, 폴더 이름 옆의 수정 버튼을 클릭해서 이름 변경 작업을 수행할 수 있습니다.



8.1.4 폴더 삭제하기

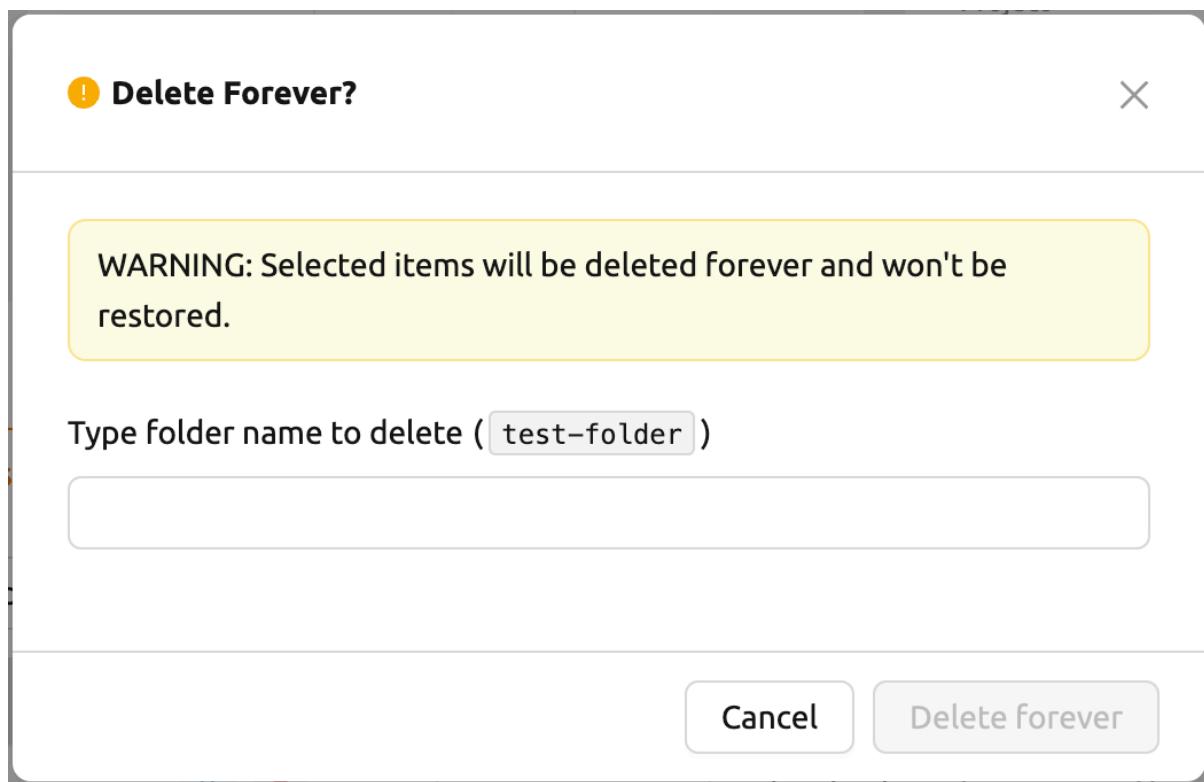
스토리지 폴더를 삭제할 수 있는 권한을 가지고 있는 경우, 제어 열의 '휴지통' 아이콘을 눌러 폴더를 휴지통으로 이동시킬 수 있습니다. 휴지통으로 이동된 폴더들은 DELETE-PENDING 상태로 표시됩니다.

Name	Controls	Status	Location	Type	Mount Permission	Owner
project-folder	Move to trash bin READY	local:volume1	Project	Read & Write R W		
test-folder2	Move READY	local:volume1	User	Read & Write R W		
test-folder	Move READY	local:volume1	User	Read & Write R W		

'DELETE-PENDING' 상태에서는 제어 열의 '복원' 버튼을 클릭해 폴더를 복원할 수 있습니다. 폴더를 완전히 삭제하려면, 같은 열에 존재하는 '휴지통' 버튼을 클릭합니다.

Name	Controls	Status	Location	Type	Mount Permission	Owner
test-folder	Trash READY	local:volume1	User	Read & Write R W		

삭제하려는 폴더 이름을 입력하는 필드를 포함한 모달이 나타나면, 입력 필드에 삭제하려는 폴더 이름을 정확하게 입력한 후, 빨간색 '영구 삭제' 버튼을 클릭하여 폴더를 완전히 삭제할 수 있습니다.



8.1.5 파일 브라우저 사용하기

Backend.AI에서는 20.09 버전부터 파일 브라우저를 지원합니다. 파일 브라우저는 웹 브라우저를 통해 원격 서버에 있는 파일 관리를 도와주는 프로그램입니다. 특히 사용자의 로컬 머신에 있는 디렉토리를 업로드 할 때에도 유용하게 사용할 수 있습니다.

현재 Backend.AI에서는 파일 브라우저를 연산 세션 내에서 실행되는 애플리케이션 형태로 제공합니다. 따라서, 다음과 같은 최소 조건이 필요합니다.

- 최소 1개 이상의 세션을 생성 할 수 있음.
- 최소 CPU 1 core, RAM 0.5 GB 이상의 여유 자원.
- 파일 브라우저를 지원하는 이미지가 설치되어 있어야 함.

파일 브라우저는 두 가지 방법으로 사용할 수 있습니다.

- 폴더 탐색기 창에서 실행.
- 세션 페이지에서 파일 브라우저 이미지로 연산 세션 직접 생성.

폴더 탐색기에서 파일 브라우저 실행

데이터 페이지로 이동 후 원하는 데이터 폴더 이름을 클릭하여 탐색기 창을 띄웁니다.

The screenshot shows the 'Folders' section of the Backend.AI WebUI. At the top, there are tabs for 'Created' (3 items) and 'Trash' (2 items). Below is a search bar and a filter menu with options like 'All', 'General', 'Pipeline', 'Auto Mount', and 'Models'. A 'Name' dropdown and a 'Search' input field are also present. The main table lists three folders:

Name	Controls	Status	Location	Type	Mount Permission	Owner
project-folder		READY	local:volume1	Project	Read & Write	
test-folder2		READY	local:volume1	User	Read & Write	
test-folder		READY	local:volume1	User	Read & Write	

Total 3 items < > 10 / page

탐색기 우측 상단의 '파일브라우저 실행' 버튼을 클릭해주세요.

The screenshot shows the 'test-folder' file browser window. At the top, there are buttons for 'Execute filebrowser' (blue), 'Run SFTP server' (orange), 'Read & Write' (green), and a close button. The main area displays a table of files:

Name	Created	Size	Actions
models	Tue, 08 Apr 2025 18:08:54 GMT	64	
image.png	Tue, 08 Apr 2025 18:09:16 GMT	6883300	

파일 브라우저가 새 창에서 열린 것을 확인할 수 있습니다. 탐색기를 열었던 저장 폴더가 파일 브라우저의 루트 디렉토리가 되는 것도 알 수 있습니다. 파일 브라우저 창에서 디렉토리와 파일을 자유롭게 업로드하고 수정하고 삭제할 수 있습니다.

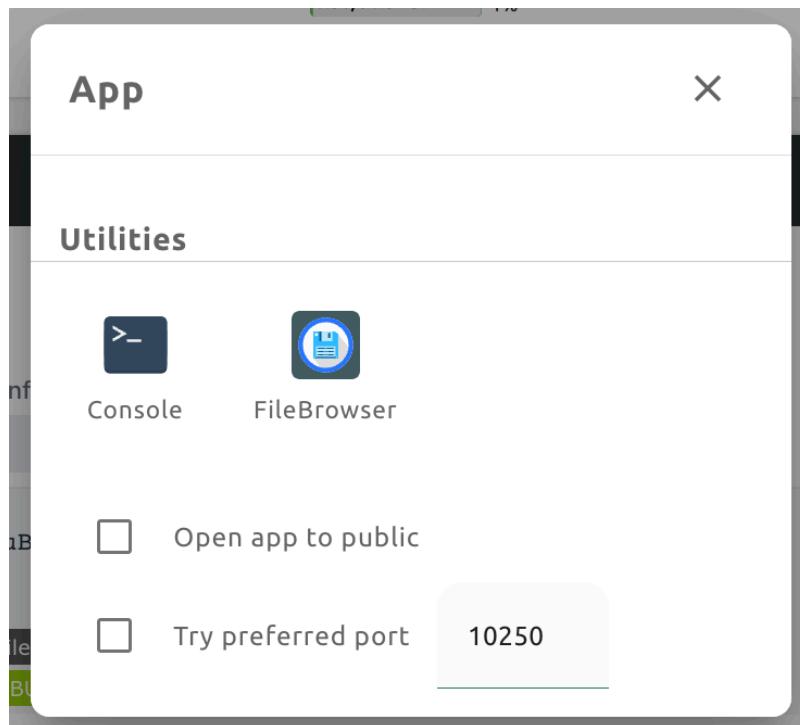
The screenshot shows a standard file browser interface with a sidebar and a main content area. The sidebar includes 'My files', 'New folder', 'New file', and 'Settings'. The main area shows a table with a single item:

Name	Size	Last modified
models	—	a minute ago

사용자가 '파일브라우저 실행' 버튼을 클릭하면, Backend.AI는 자동으로 파일 브라우저 전용 연산 세션을 하나 생성하게 됩니다. 따라서, 세션 페이지에서 파일 브라우저 연산 세션이 조회되는 것을 확인할 수 있습니다. 이 연산 세션을 삭제하는 것은 사용자의 몫입니다.

RUNNING	INTERACTIVE	BATCH	INFERENCE	UPLOAD SESSIONS	FINISHED	...	START
User ID	Session Info		Status	Control	Configuration	Usage	
demo-admin@lablup.c FileBrowser 23.10 AMD64	BQcc134z-jSDK	RUNNING		test-folder nvidia-H100RG 1core 0.50GiB (SHM: 0.06GiB) 0.00GPU	CPU 0.0 % RAM 0.09 / 0.5 GiB GPU(mem) 0 / 0 GiB I/O R: 0.0 MB / W: 0.5 M		

참고: 파일 브라우저 창을 실수로 종료하여 다시 열고자 한다면, 세션 페이지로 가서 해당 세션의 Control 열에 있는 애플리케이션 아이콘을 선택하고 파일 브라우저 버튼을 클릭하면 됩니다.



만약 탐색기에서 '파일브라우저 실행' 버튼을 다시 클릭하면 새로운 세션이 생성되어 총 두 개의 파일 브라우저 연산 세션이 나타나게 됩니다.

파일 브라우저 이미지로 연산 세션 생성하기

파일 브라우저를 지원하는 이미지를 선택해서 연산 세션을 직접 생성할 수도 있습니다. 연산 세션 생성 시 작업하기 원하는 데이터 폴더를 하나 이상 마운트 하십시오. 아무 데이터 폴더를 마운트 하지 않아도 파일 브라우저 사용에는 문제가 없지만, 연산 세션이 종료되면 연산 세션에서 작업한 모든 파일이 삭제됩니다.

참고: 세션 페이지에서 생성한 파일 브라우저의 root 디렉토리는 `/home/work`입니다. 따라서, 연산 세션이 마운트된 모든 데이터 폴더에 자유롭게 접근할 수 있습니다.

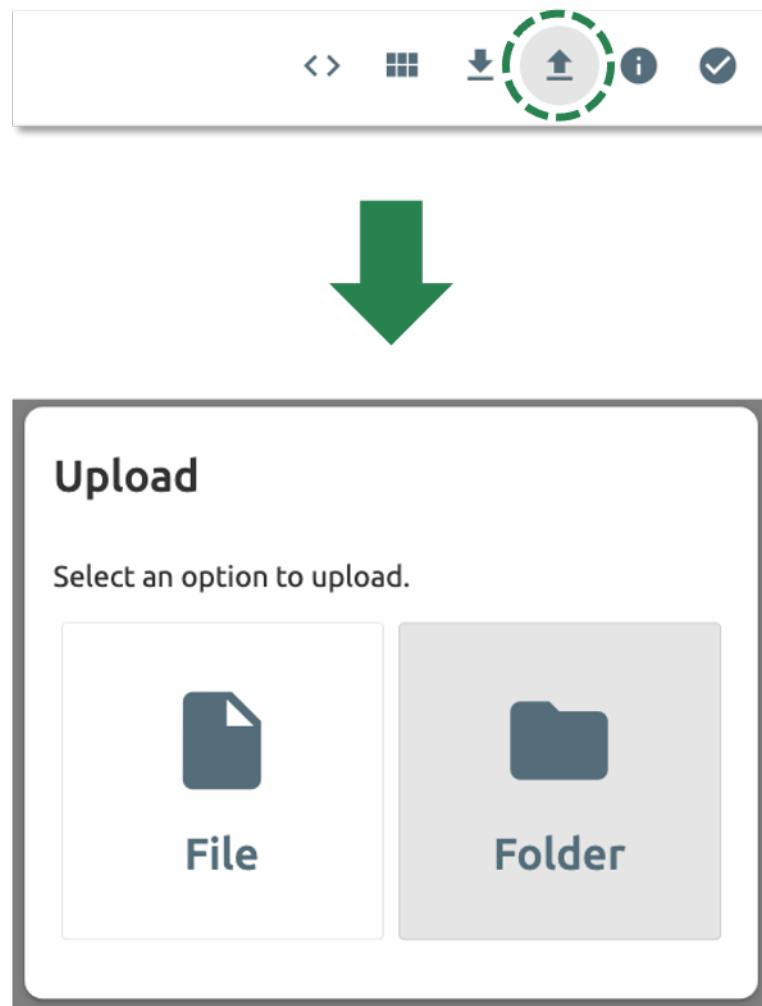
파일 브라우저 기본 사용법

여기서는 Backend.AI 상에서 파일 브라우저의 간단한 사용 예를 보여 드립니다. 대부분의 파일 브라우저 작업은 매우 직관적이지만, 보다 자세한 가이드가 필요하실 경우 [파일 브라우저 문서](#)를 참조하십시오.

파일 브라우저로 로컬 디렉토리 업로드 하기

파일 브라우저는 로컬 디렉토리 구조를 그대로 보존하여 업로드 할 수 있는 기능을 지원합니다. 파일 브라우저 윈도우 우측 상단의 업로드 버튼을 클릭한 후 Folder 버튼을 클릭하십시오. 로컬 파일 탐색창이 뜨는데, 여기서 업로드 하기 원하는 디렉토리를 선택하면 됩니다.

참고: 읽기 전용 폴더에 파일을 업로드 하는 경우, FileBrowser가 서버 에러를 띄우게 됩니다.



다음과 같은 구조를 가진 폴더를 업로드 해보겠습니다.

```
foo
+-- test
|   +-- test2.txt
+-- test.txt
```

foo 디렉토리를 선택한 후 폴더가 성공적으로 업로드 된 것을 확인할 수 있습니다.

Name ↑			Size	Last modified
test		—		a few seconds ago
test.txt		0 B		a few seconds ago

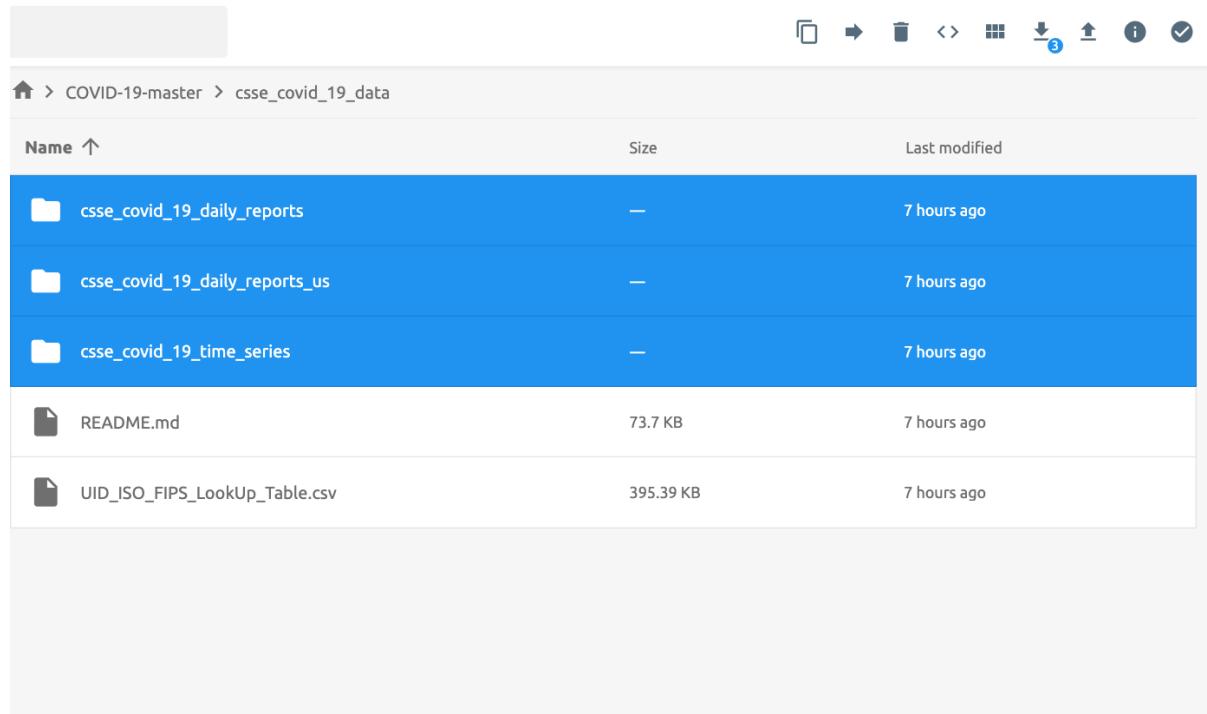
Name ↑			Size	Last modified
test2.txt		0 B		a few seconds ago

또한, 파일과 디렉토리를 드래그 앤 드롭 하여 업로드할 수도 있습니다.

파일 또는 디렉토리를 다른 디렉토리로 옮기기

데이터 폴더 내 파일 또는 디렉토리를 옮기는 것도 가능합니다. 파일 또는 디렉토리의 위치를 변경하는 방법은 다음과 같습니다.

1. 파일 브라우저에서 원하는 디렉토리나 파일을 선택하십시오.



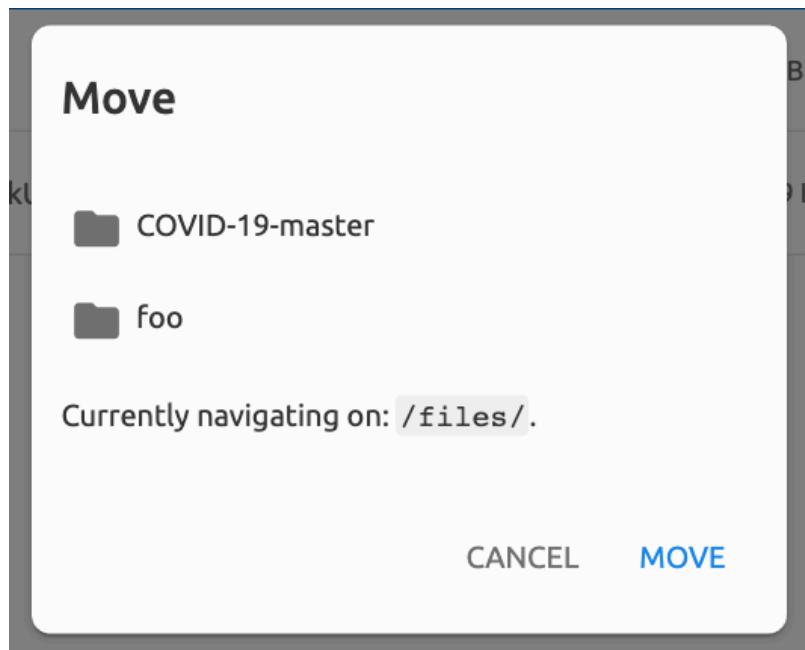
The screenshot shows a file browser interface with the following structure:

- Top Bar:** Includes icons for back, forward, search, refresh, download, upload, and other file operations.
- Breadcrumb:** Shows the path: [Home](#) > COVID-19-master > csse_covid_19_data
- Table Headers:** Name ↑, Size, Last modified
- Selected Item:** The 'csse_covid_19_time_series' directory is highlighted in blue.
- Other Items:**
 - csse_covid_19_daily_reports
 - csse_covid_19_daily_reports_us
 - README.md
 - UID_ISO_FIPS_LookUp_Table.csv

2. 파일 브라우저 상단의 '화살표' 버튼을 클릭합니다.



3. 이동할 경로를 선택합니다.



4. 'MOVE' 버튼을 클릭합니다.

옮기기 작업이 성공적으로 완료된 것을 확인할 수 있습니다.

Name ↑	Size	Last modified
COVID-19-master	—	7 hours ago
csse_covid_19_daily_reports	—	7 hours ago
csse_covid_19_daily_reports_us	—	7 hours ago
csse_covid_19_time_series	—	7 hours ago
foo	—	20 minutes ago

참고: 현재 파일 브라우저는 연산 세션 내 애플리케이션 형태로 제공되고 있습니다. 향후 세션을 생성하지 않고 파일 브라우저를 독립적으로 실행할 수 있는 기능을 추가할 예정입니다.

8.1.6 SFTP 서버 사용

Backend.AI 22.09 버전부터 데스크탑 앱 및 웹 기반 WebUI 환경에서 SSH / SFTP 프로토콜을 통해 파일을 업로드 할 수 있는 방법을 제공합니다. SFTP 서버를 사용하여 안정적으로 빠르게 파일을 업로드할 수 있습니다.

참고: 설정에 따라, SFTP 서버 실행하기가 허용되지 않을 수 있습니다.

Data 페이지의 폴더 탐색기에서 파일 브라우저 실행

Data 페이지로 이동 후 원하는 데이터 폴더의 탐색기 창을 띄웁니다. 폴더 아이콘 혹은 폴더 이름을 클릭하여 파일 탐색기를 띄웁니다.

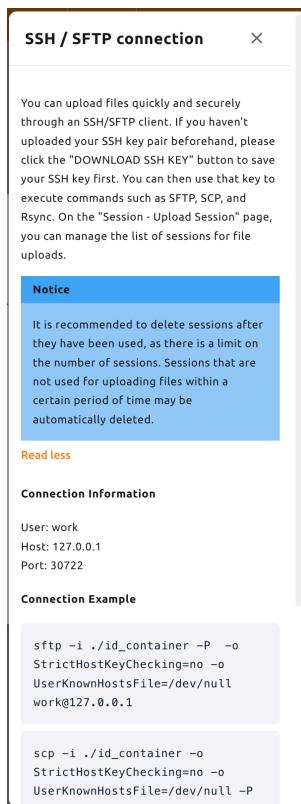
탐색기 우측 상단의 'SFTP 서버 사용' 버튼을 클릭해주세요.

Name	Created	Size	Actions
models	Tue, 08 Apr 2025 18:08:54 GMT	64	
image.png	Tue, 08 Apr 2025 18:09:16 GMT	6883300	

SSH / SFTP 연결ダイ얼로그가 뜨면서, 새로운 SFTP 세션이 자동으로 생성됩니다. (이 세션은 자원 점유 상태에 영향을 주지 않습니다.)



참고: 대용량 파일 업로드를 SSH/SFTP 연결을 통해 수행하는 방법을 상세히 제공하고 있습니다. 자세한 정보를 확인하시려면, '더 알아보기' 텍스트 링크를 클릭하여 실행에 대한 상세 정보를 확인해주시기 바랍니다.



연결을 위해 'SSH 키 다운로드' 버튼을 클릭하여 SSH 비밀키(id_container)를 다운로드 합니다. 접속 주소와 포트 번호도 기억해 두십시오. 디아일로그에 적힌 접속 예제 코드를 참고하거나, 다음 링크를 참고하여 세션으로 파일을 전송할 수 있습니다. [link](#) 파일을 보존하려면 데이터 폴더로 복사하십시오. 또한, 일정 시간 동안 파일 전송이 없는 경우에는 세션이 삭제될 수 있습니다.

참고: 미리 사용자 SSH 키페어를 업로드 해두면 id_container의 내용이 업로드한 SSH 사설 키로 채워집니다. 따라서, 연산 세션에 SSH 접속을 할 때마다 파일을 다운로드 할 필요가 없습니다. 사용자의 SSH 키페어 관리하기 부분을 참고하십시오.

8.2 폴더 종류

8.2.1 파이프라인 폴더

이 탭은 FastTrack에서 파이프라인을 실행할 때 자동으로 생성되는 폴더 목록을 보여줍니다. 파이프라인을 만들면 각 태스크 인스턴스(연산 세션)에 대응되는 컨테이너의 /pipeline 위치에 새 폴더가 생성되고 마운트됩니다.

8.2.2 자동 마운트 폴더

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

자동 마운트 사용에 관한 보다 자세한 내용은 [자동 마운트 폴더 사용 예제](#) 섹션을 참고하십시오.

The screenshot shows the 'Folders' interface with the 'Auto Mount' tab selected. The list view shows a single entry: '.local'. The entry has a status of 'READY' and is located at 'local:volume1'. It has 'Read & Write' permissions for the user. The bottom of the screen shows a pagination control with '10 / page'.

8.2.3 모델 폴더

'모델' 탭은 모델을 [서빙](#) 하는 데에 필요한 파일들을 저장해두는 곳입니다. 모델 학습이나 서비스를 위한 데이터를 저장해둘 수 있습니다.

The screenshot shows the 'Folders' interface with the 'Models' tab selected. The list view shows a single entry: 'model_folder'. The entry has a status of 'READY' and is located at 'local:volume1'. It has 'Read & Write' permissions for the user. The bottom of the screen shows a pagination control with '10 / page'.

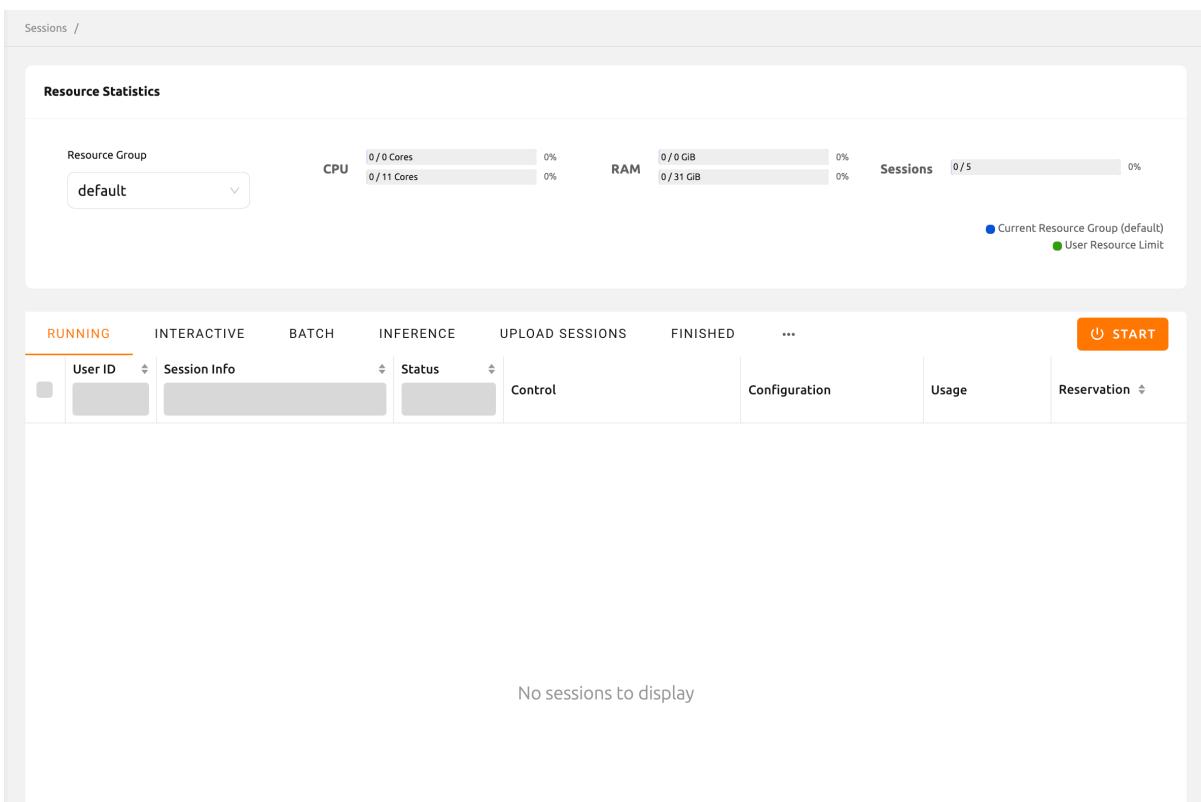
CHAPTER 9

연산 세션

Backend.AI WebUI 에서 가장 많이 방문하게 될 페이지는 세션 페이지와 데이터 페이지입니다. 세션 페이지에서는 컨테이너 기반의 연산 세션을 조회하거나 생성 및 사용할 수 있고, 데이터 페이지에서는 데이터를 보관하는 저장 폴더를 생성할 수 있습니다. 여기서는 세션 페이지에서 컨테이너 기반의 연산 세션을 생성하고 각종 웹 애플리케이션을 활용하는 방법을 알아봅니다.

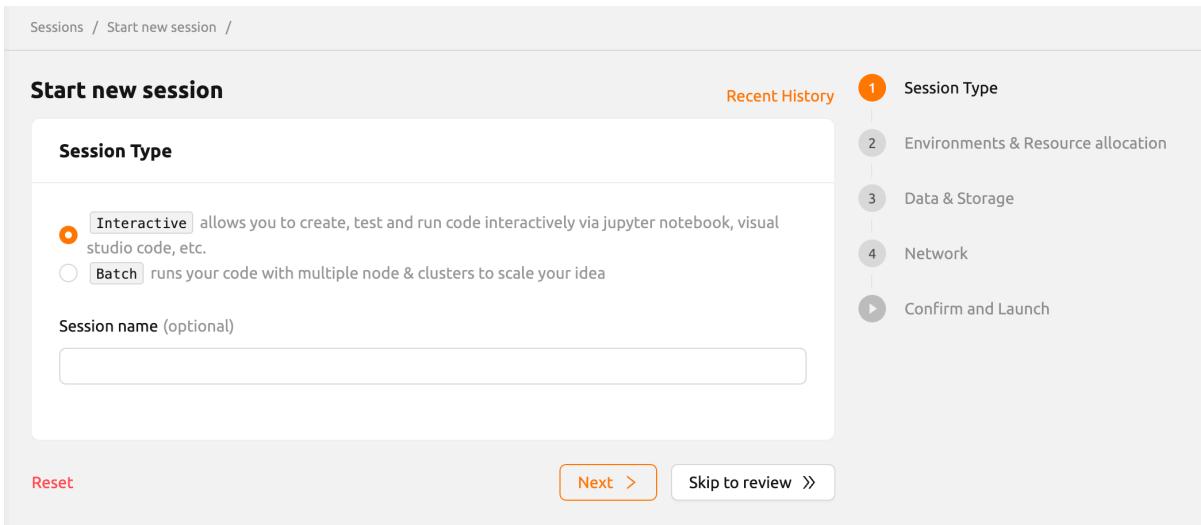
9.1 새로운 세션 시작하기

사용자 계정으로 로그인 후 좌측 메뉴의 '세션'을 클릭하여 세션 페이지로 이동합니다. '세션 페이지'는 새로운 연산 세션을 시작하거나 이미 실행 중인 연산 세션을 관리할 때 사용할 수 있습니다.



Backend.AI 24.09 버전부터는 세션 런처(NEO)의 개선된 버전을 기본 런처로 제공합니다. 이전의 대화식 세션 런처를 사용하고 싶다면, [사용자 설정 가이드](#)를 참고하세요. 이전의 대화식 세션 런처의 사용방법에 대한 자세한 내용은 다음 [링크](#)를 참고하세요.

새로운 연산 세션을 시작하려면 '시작' 버튼을 클릭하세요.



9.1.1 세션 타입

첫 번째 페이지에서는 세션의 형태인 interactive 또는 batch 를 선택해야 합니다. 그리고 세션 이름을 지정할 수 있습니다. (선택사항)

- 세션 타입: 세션의 형태를 결정합니다. 현재 가능한 세션 형태는 “Interactive” 와 “Batch” 두 가지가 있습니다. 두 형태의 주요한 차이점은 다음과 같습니다:

- Interactive 형태 연산 세션

- * Backend.AI 초기 버전부터 지원하던 형태입니다.
 - * 사용자가 별도의 실행 스크립트를 지정하지 않고 일단 세션을 생성한 후 상호 작용하는 방식으로 세션을 사용하게 됩니다.
 - * 사용자가 명시적으로 삭제하지 않는 한 세션은 자동 삭제되지 않습니다. 다만, 관리자가 별도의 세션 자동 수거 설정을 켜둔 경우에는, 그 조건에 따라 자동으로 삭제될 수도 있습니다.

- Batch 형태 연산 세션

- * Backend.AI 22.03 부터 GUI 를 통해 제공합니다. 다만, CLI 의 경우에는 그 이전부터 지원하고 있었습니다.
 - * 사용자가 연산 세션을 생성할 때 실행할 스크립트를 미리 지정합니다.
 - * 자원이 할당되는 즉시 해당 스크립트를 실행하고, 스크립트가 종료되는 그 즉시 연산 세션을 자동으로 삭제합니다. 따라서, 실행할 코드가 사전에 정의되어 있거나 작업을 파이프라인 하는 경우에는, 연산 서버 팜(server farm)의 자원을 보다 효율적으로 활용할 수 있는 장점이 있습니다.
 - * 연산 세션 시작 시점을 지정할 수 있습니다. 다만, 시작 시점에 반드시 연산 세션이 자원을 할당받을 수 있는 것은 아닙니다(자원 부족 등의 이유로 PENDING 상태에 계속 머물 수 있습니다). 시작 시점 이전에는 자원이 있어도 연산 세션을 스케줄링 하지 않는 개념으로 이해하는 것이 정확합니다.
 - * 배치 작업 최대 실행 시간: 배치 작업의 최대 실행 시간을 설정합니다. 지정된 시간이 초과되면 세션이 자동으로 종료됩니다.

Sessions / Start new session /

Start new session

Session Type

Interactive: allows you to create, test and run code interactively via jupyter notebook, visual studio code, etc.

Batch: runs your code with multiple node & clusters to scale your idea

Session name (optional)

Batch mode Configuration

Startup Command

```
sleep 10 && echo "hi, batch session!"
```

Session Starts (optional)

Enable 2025-04-09 14:06:16

in a minute

Batch Job Timeout Duration (optional)

Enable Sec

Recent History

- 1 Session Type
- 2 Environments & Resource allocation
- 3 Data & Storage
- 4 Network
- Confirm and Launch

Buttons: Reset, Next >, Skip to review >

- 세션 이름 (선택사항): 생성할 연산 세션의 이름을 지정할 수 있습니다. 지정하면 세션 정보에 이 이름이 나타나므로 연산 세션의 구분이 용이합니다. 지정하지 않으면 임의의 이름이 자동으로 지정됩니다. 세션 이름은 4-64자 사이의 알파벳 또는 숫자만 받아들이며, 공백은 허용되지 않습니다.

super admin 혹은 admin 계정으로 세션을 생성하는 경우, 추가적으로 세션 소유자를 할당할 수 있습니다. 토클을 클릭해 할당을 활성화하면 사용자 이메일 입력 필드가 나타납니다.

Sessions / Start new session /

Start new session

Session Type

Interactive: allows you to create, test and run code interactively via jupyter notebook, visual studio code, etc.

Batch: runs your code with multiple node & clusters to scale your idea

Session name (optional)

Session owner

Recent History

- 1 Session Type
- 2 Environments & Resource allocation
- 3 Data & Storage
- 4 Network
- Confirm and Launch

Buttons: Reset, Next >, Skip to review >

Sessions / Start new session /

Start new session

Session Type

Interactive: allows you to create, test and run code interactively via jupyter notebook, visual studio code, etc.

Batch: runs your code with multiple node & clusters to scale your idea

Session name (optional)

Session owner

Owner Email

Owner access key

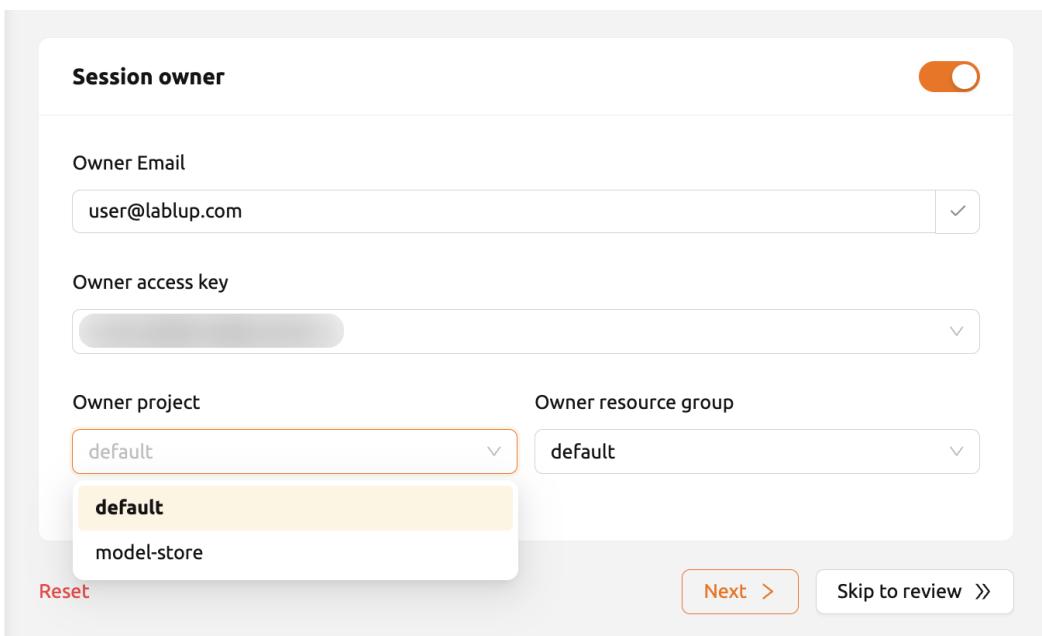
Owner project **Owner resource group**

Recent History

- 1 Session Type
- 2 Environments & Resource allocation
- 3 Data & Storage
- 4 Network
- Confirm and Launch

Buttons: Reset, Next >, Skip to review >

세션 할당을 위한 사용자 이메일을 입력하고 '검색' 버튼을 클릭하면, 해당 사용자의 access key가 자동으로 등록됩니다. 추가적으로 프로젝트와 자원 그룹을 선택해 할당할 수 있습니다.



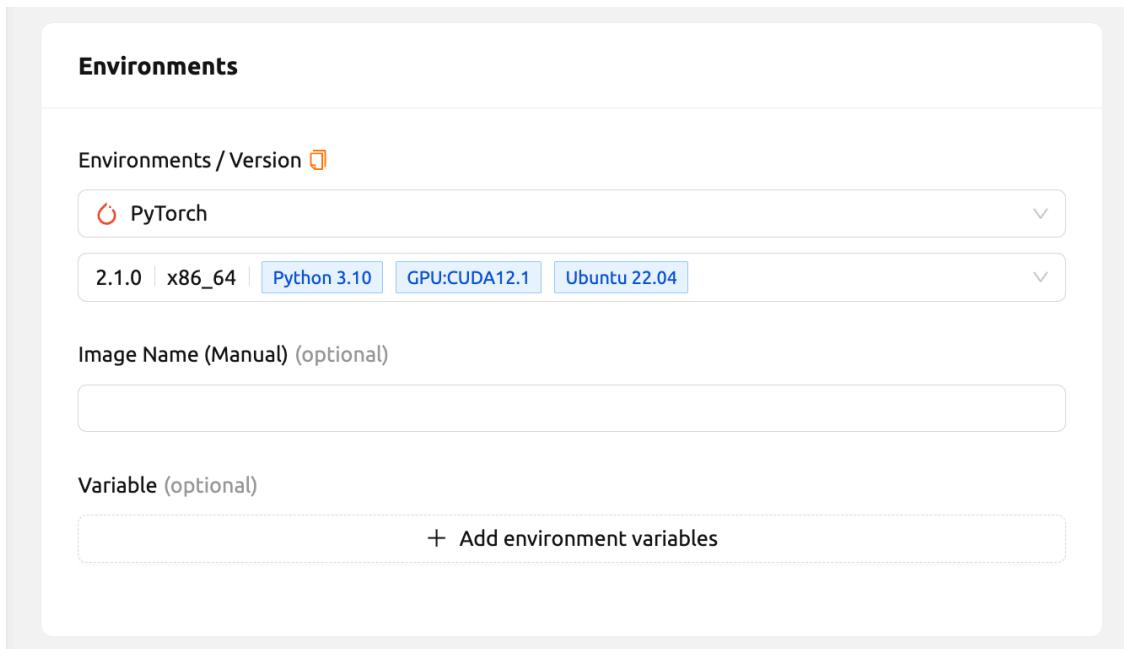
9.1.2 실행 환경 및 자원 할당

다음 페이지로 진행하려면 아래쪽의 '다음' 버튼을 클릭하거나, 우측의 '실행 환경 & 자원 할당' 버튼을 클릭하십시오. 추가 설정 없이 세션을 생성하려면 '검토로 건너뛰기' 버튼을 누르십시오. 이 경우, 다른 페이지의 설정은 모두 기본값을 사용하게 됩니다.

9.1.3 실행 환경

두 번째 페이지에서 설정할 수 있는 각 항목에 대한 자세한 설명은 다음을 참고하십시오.

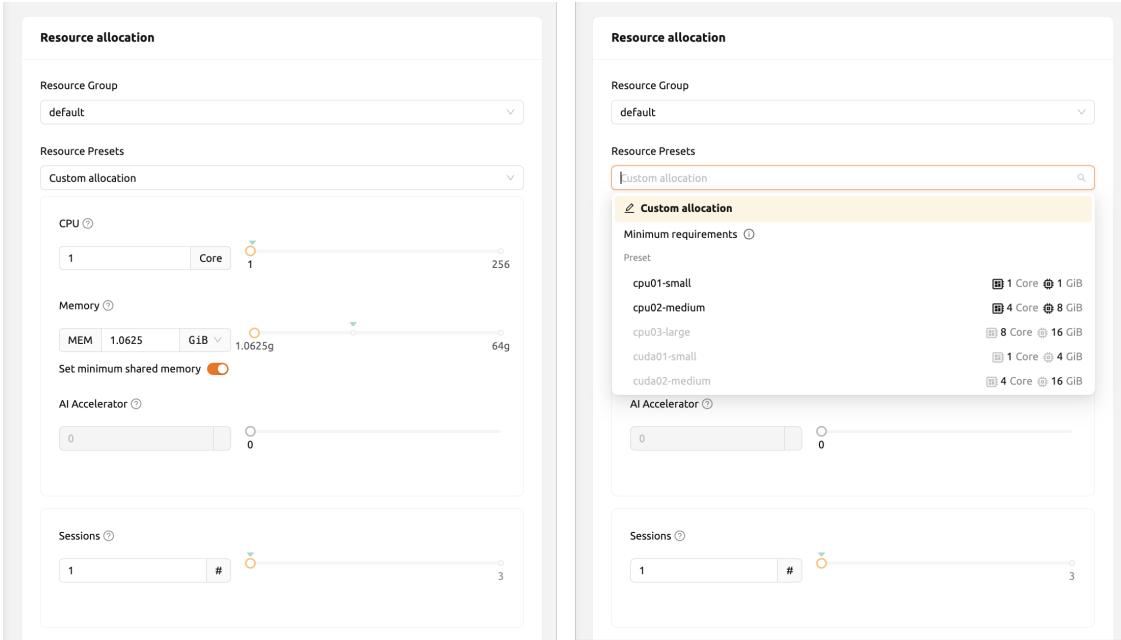
- 실행 환경: TensorFlow, PyTorch, C++ 등과 같은 연산 세션의 기본 환경을 지정합니다. TensorFlow를 선택하면 연산 세션에서 TensorFlow 라이브러리를 사용할 수 있습니다. 다른 환경을 선택하면 해당 환경이 기본적으로 설치된 연산 세션을 생성하게 됩니다.
- 버전: 사용자는 실행 환경의 버전을 지정할 수 있습니다. 하나의 환경에는 여러 버전이 존재할 수 있습니다. 예를 들어, TensorFlow는 1.15, 2.3 등 다양한 버전을 제공합니다.
- 환경 이름 (선택사항): 연산 세션에 사용할 이미지의 이름을 지정할 수 있습니다. 환경 설정에 따라 이 설정이 사용 불가능할 수도 있습니다.
- Backend.AI에서는 사용자에게 더 편리한 환경을 제공하기 위해, 세션 시작 화면에서 세션에 추가될 환경 변수 추가 기능을 지원합니다. 이 기능에서 여러분은 PATH를 비롯한 모든 환경 변수를 설정 다이얼로그에서 변수명과 변수값을 입력해서 추가할 수 있습니다.



9.1.4 자원 할당

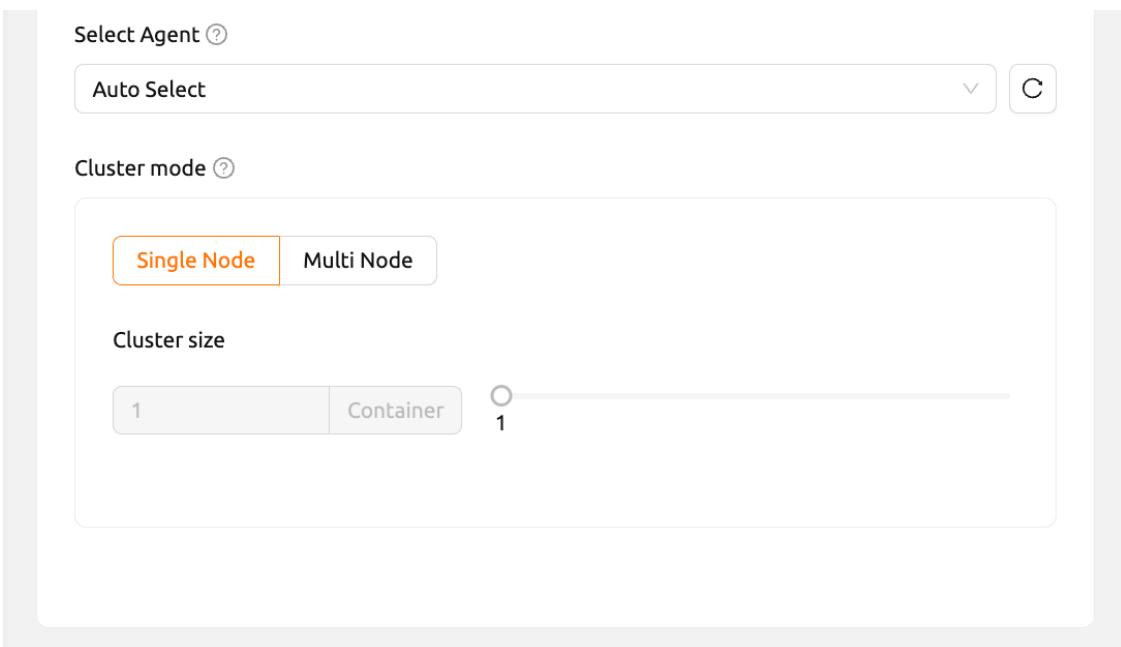
- 자원 그룹: 연산 세션을 생성할 자원 그룹을 지정합니다. 자원 그룹은 각 사용자가 접근할 수 있는 호스트 서버를 묶은 단위이며, 주로 같은 종류의 GPU 자원을 보유하고 있는 서버들을 하나의 자원 그룹으로 설정합니다. 관리자는 서버를 종류별로 구분하여 자원 그룹으로 묶고, 사용자가 사용할 수 있는 자원 그룹의 종류를 지정할 수 있습니다. 사용자는 관리자가 허가한 자원 그룹 서버에만 연산 세션을 생성할 수 있습니다. 자원 그룹이 여러 개인 경우 원하는 그룹을 선택할 수 있지만, 하나만 있는 경우에는 변경할 수 없습니다.

- 자원 프리셋: 이 템플릿은 연산 세션에 할당할 CPU, 메모리, GPU 등의 자원 세트를 미리 정의해 둔 것입니다. 관리자는 미리 자주 사용하는 자원 설정을 정의할 수 있습니다. 숫자 입력을 조정하거나 슬라이더를 움직이면 원하는 자원량을 할당할 수 있습니다.



각 항목의 의미는 다음과 같습니다. '도움 (?)' 버튼을 클릭하면 자세한 정보를 확인할 수 있습니다.

- CPU: CPU는 명령어로 지정된 기본 산술, 논리, 제어 및 입출력 (I/O) 작업을 수행합니다. 고성능 컴퓨팅 워크로드의 경우 많은 CPU가 도움이 되지만, 여러 CPU를 사용하도록 프로그램 코드를 작성해야 합니다. (그렇지 않으면 대부분의 CPU는 사용되지 않을 것입니다.)
- 컴퓨터 메모리는 임시 저장 영역입니다. 중앙 처리 장치 (CPU)에 필요한 데이터와 명령들을 보관하는 역할을 합니다. 머신 러닝 워크로드를 처리할 때 GPU를 연산 장치로 사용하는 경우, GPU 메모리의 두 배 이상의 메모리를 할당해야 합니다. 그렇지 않으면 GPU의 유휴 시간이 증가하여 성능이 저하됩니다.
- 공유 메모리: 연산 세션에 할당할 공유 메모리의 용량 (GB). RAM에 설정된 메모리 중 일부를 떼어 공유 메모리로 사용합니다. 따라서, RAM에 지정된 양보다 클 수 없습니다.
- AI 가속기: AI 가속기 (GPU 및 NPU)는 기계 학습과 관련된 행렬 / 벡터 계산에 적합합니다. AI 가속기는 훈련 및 인퍼런스 알고리즘을 몇 배나 가속화하여 기계 학습 워크로드의 실행 시간을 몇 주에서 며칠로 줄입니다.
- 세션은 지정된 환경과 자원에 따라 생성되는 계산 환경 단위입니다. 이 값을 1보다 큰 값으로 설정하면 위의 자원 설정을 이용한 여러 세션이 지정한 값 만큼 동시에 생성됩니다. 세션 시작 요청 시 사용 가능한 자원이 충분하지 않은 경우, 생성하지 못한 세션 시작 요청들은 생성 대기열에 추가됩니다.

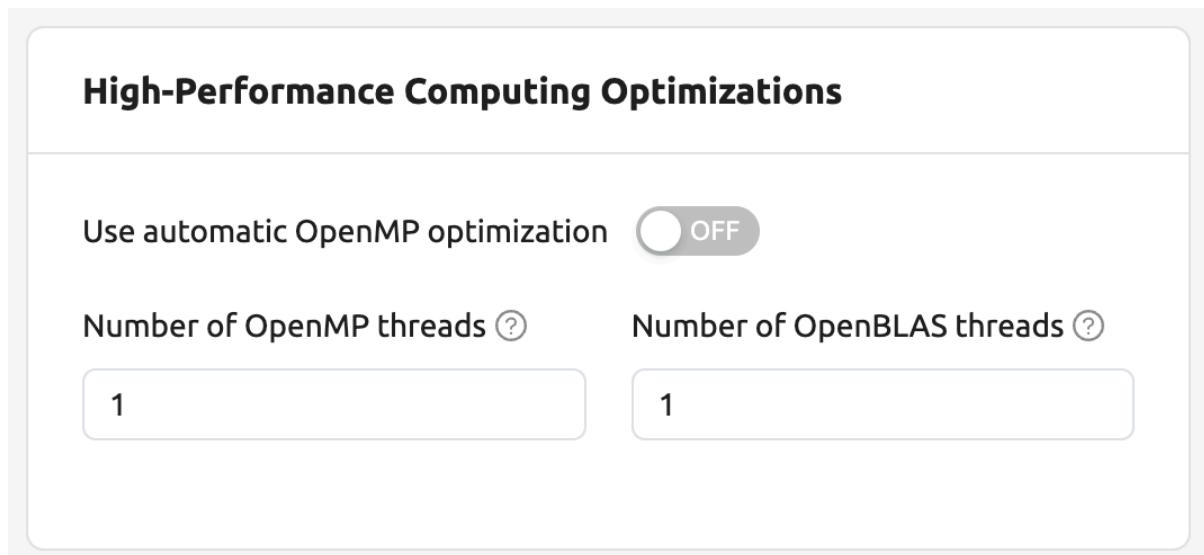


- 에이전트 선택: 사용자는 할당할 에이전트를 직접 선택할 수 있습니다. 기본적으로는 스케줄러가 에이전트를 자동으로 선택합니다. 에이전트 선택기에서는 각 에이전트의 실제 사용 가능한 자원 양을 확인할 수 있습니다. 현재 이 기능은 단일 노드, 단일 컨테이너 환경에서만 지원됩니다.
- 클러스터 모드 설정: Backend.AI 는 한 번에 여러 연산 세션을 생성할 수 있는 클러스터 모드를 지원합니다. 더 자세한 정보는 [Backend.AI 클러스터 연산 세션 개요](#) 섹션을 참고하세요

참고: 에이전트 선택 기능은 서버 환경에 따라 표시되지 않을 수 있습니다.

- 고성능 컴퓨팅 최적화: Backend.AI 는 HPC 최적화 관련 값을 설정할 수 있습니다.

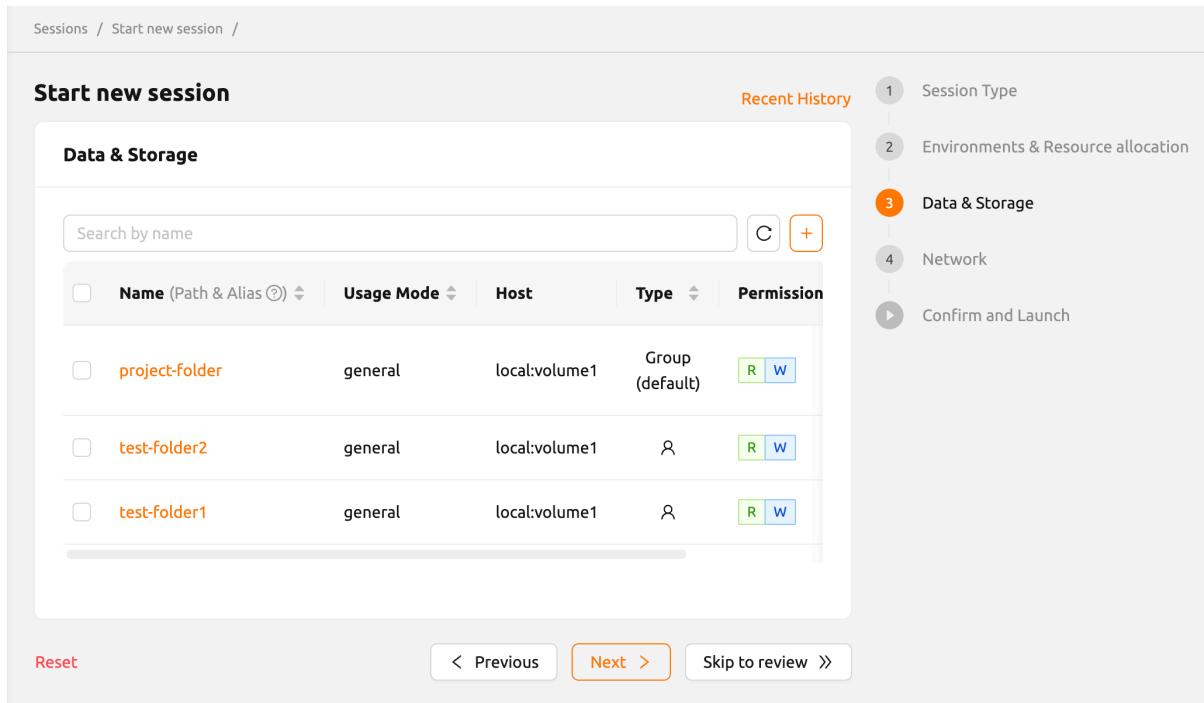
Backend.AI 는 `nthread-var` 의 내부 컨트롤 변수 설정 인터페이스를 제공합니다. Backend.AI 는 기본적으로 이 값을 세션의 CPU 코어 수와 같도록 설정해두며, 이는 일반적인 고성능 컴퓨팅 워크로드를 가속하는 효과가 있습니다. 그러나, 일부 멀티스레드 워크로드의 경우 OpenMP를 사용하는 다중 프로세스가 동시에 실행되어 비정상적으로 많은 스레드가 생성되고, 현저한 성능 저하가 발생할 수 있습니다. 이러한 문제를 해결하려면, 스레드 값을 1 또는 2로 조정하십시오.



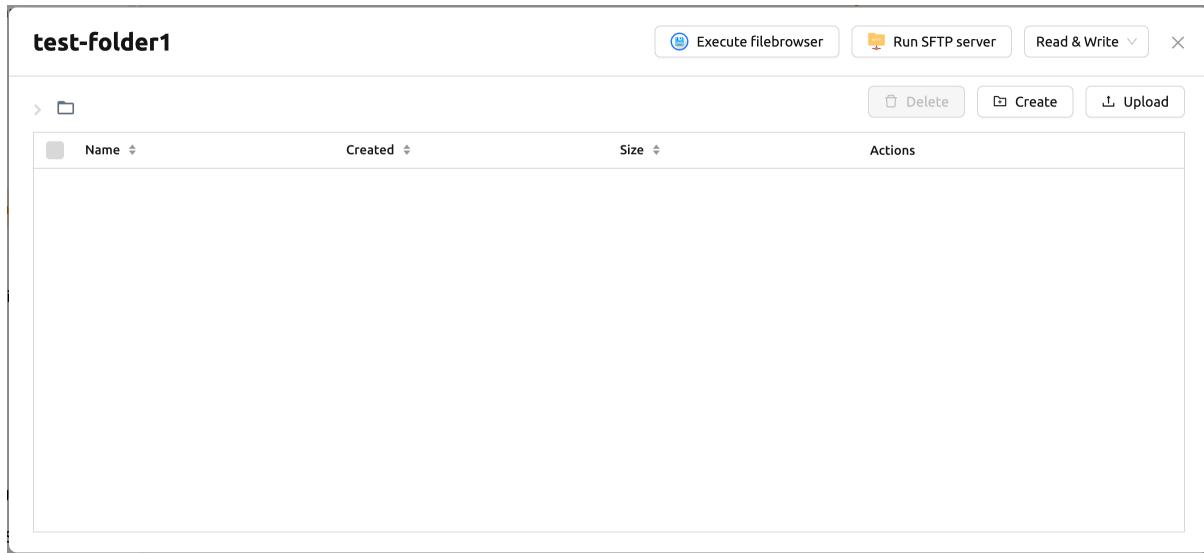
9.1.5 데이터 및 폴더

다음 페이지로 진행하려면 아래쪽의 '다음' 버튼을 클릭하거나, 우측의 '데이터 & 폴더' 버튼을 클릭하세요.

여기서는 연산 세션에 마운트 할 데이터 폴더를 지정할 수 있습니다. 연산 세션이 삭제되면 기본적으로 모든 데이터가 함께 삭제되지만, 여기서 마운트 한 폴더에 저장된 데이터는 삭제되지 않습니다. 마운트 폴더에 저장된 데이터는 다른 연산 세션을 생성할 때 다시 마운트하여 재사용 할 수도 있습니다. 폴더를 마운트하고 연산 세션을 실행하는 방법에 대한 정보는 [연산 세션에 폴더 마운트](#) 장을 참고하십시오. 여기서는 폴더를 마운트 하지 않고 그냥 지나가겠습니다. 다음 페이지로 이동합시다.



사용자는 연산 세션에 마운트 가능한 폴더 목록을 확인할 수 있습니다. 폴더 이름을 클릭해 파일 탐색기를 사용할 수 있습니다. 파일 탐색기에 대한 자세한 설명은 [폴더 내용 조회하기](#) 섹션을 참고하세요.



또한 파일 이름 검색기 옆의 '+' 버튼을 클릭하여 새로운 폴더를 생성할 수 있습니다. 새로운 폴더를 생성하는 경우, 마운트 할 폴더로 자동 선택됩니다. 폴더 생성 모달에 대한 자세한 사용 방법은 [스토리지 폴더 생성](#) 를 참고하세요.

Create a new storage folder

Folder name * (maximum 64 chars)

Location local:volume1

Usage Mode General

Type User

Permission Read & Write

Reset **Cancel** **Create**

9.1.6 네트워크

다음 페이지로 진행하려면 아래쪽의 '다음' 버튼을 클릭하거나, 우측의 '네트워크' 버튼을 클릭하세요.

- 사전 개방 포트: 사용자가 연산 세션에 사전 개방 포트를 설정할 수 있는 인터페이스를 제공합니다. 사용 방법은 세션 생성하기 전에 사전 개방 포트를 추가하는 방법 섹션을 참고하십시오.

Sessions / Start new session /

Start new session

Network

Preopen Ports ⓘ (optional)
Enter multiple values separated by either a comma (,) or a space.

Recent History

- Session Type
- Environments & Resource allocation
- Data & Storage
- Network**

Next > **Skip to review »**

Reset

9.1.7 검토 및 시작

네트워크 설정을 완료했다면, 아래쪽의 '다음' 버튼을 클릭하거나, 우측의 '검토 및 시작' 버튼을 클릭하여 마지막 페이지로 이동합니다.

마지막 페이지에 도착했습니다. 이전 페이지에서 설정한 연산 세션 환경, 자원 할당량과 마운트 정보, 설정된 환경변수, 사전 개방 포트 등의 정보를 볼 수 있습니다. 원하는 설정인지 마지막으로 확인한 후 '시작' 버튼을 클릭합니다. 만약, 변경하고 싶은 설정이 있다면 '이전' 버튼을 클릭하여 이전 페이지로 돌아갈 수 있습니다. 혹은, 각 카드 우측 상단에 있는 '수정' 버튼을 눌러 해당 페이지로 돌아갈 수 있습니다.

Sessions / Start new session /

Start new session

Session Type Edit

SessionType: interactive

Environments Edit

Project: default

Image: Python Pytorch 2.1.0 | x86_64 | **Python 3.10** | GPU: CUDA12.1 | Ubuntu 22.04 | ⋮

Resource allocation Edit

Resource Group: default

Resource Allocation Per Container: 1 Core | 1.063 GiB (SHM: 0.06GiB)

Agent: auto Number of container: 1

Cluster mode: Single Node

Total Allocation

1 Core | 1.063 GiB (SHM: 0.06GiB)

Data & Storage Edit

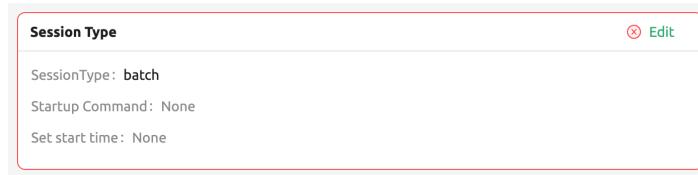
Name	Path & Alias
test-folder1	/home/work/test-folder1

Network Edit

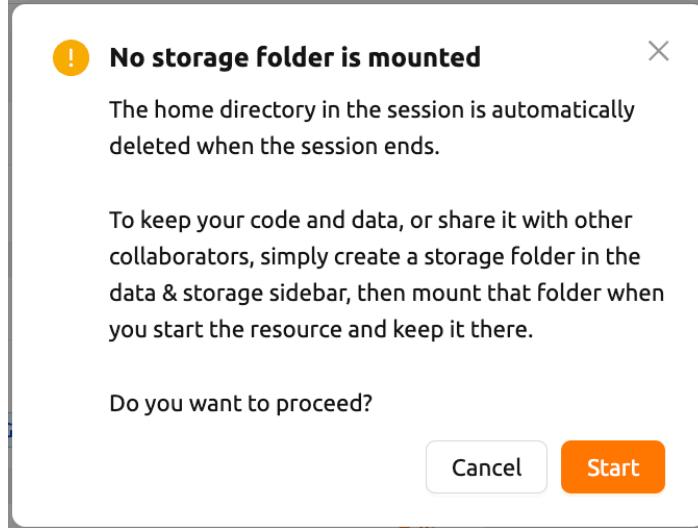
Preopen Ports: None

Reset < Previous Launch

설정에 문제가 있는 경우, 다음과 같이 오류가 표시됩니다. 설정을 수정하려면 '수정' 버튼을 클릭하십시오.



폴더 마운트 없이 시작 버튼을 클릭하면, 아무 폴더를 마운트 하지 않았다는 경고 대화 상자가 나타납니다. 폴더를 마운트할 필요가 없는 경우, 경고 대화 상자의 '시작' 버튼을 클릭하여 세션을 생성합니다.

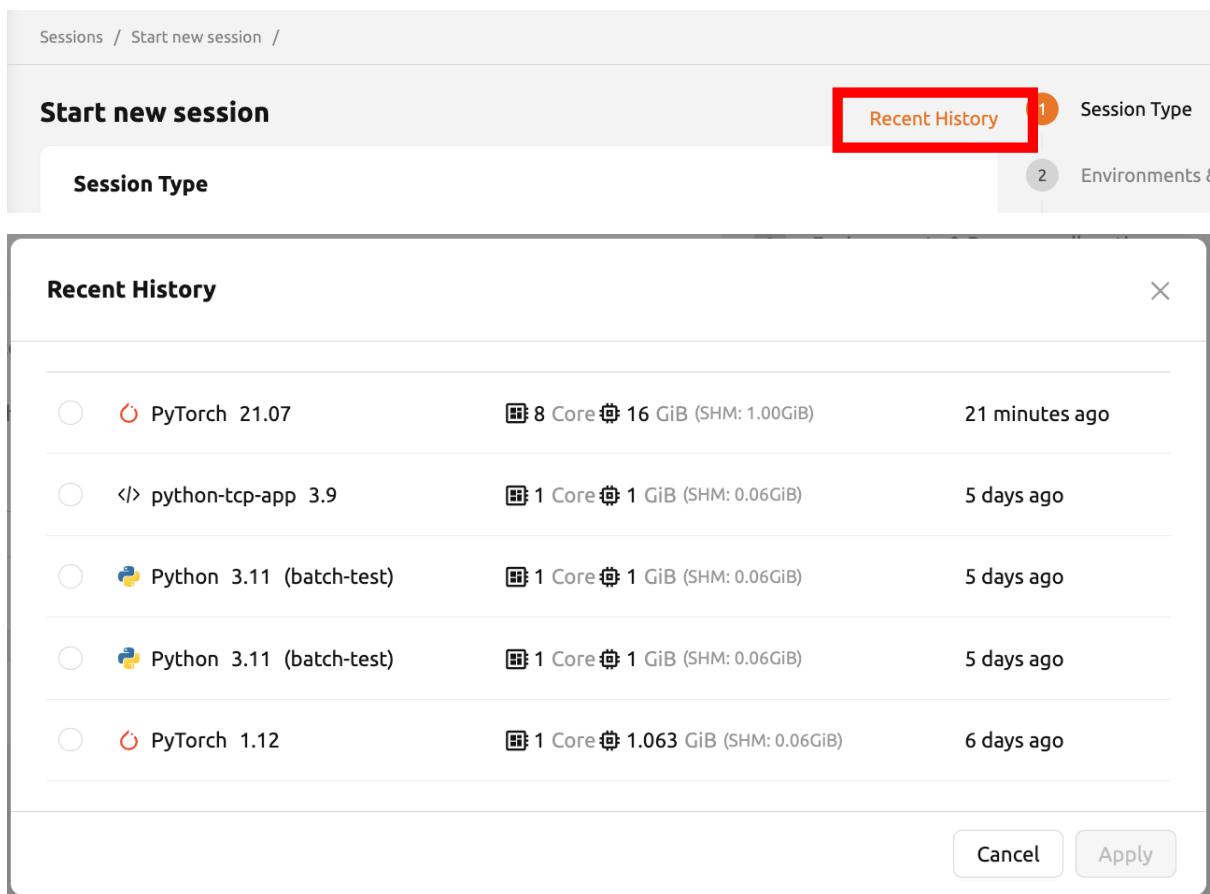


실행중 탭에서 새로운 연산 세션이 생성되는 것을 확인합시다.

실행중 탭에서 현재 실행중인 세션에 대한 정보를 확인할 수 있습니다. 여기서는 interactive, batch 탑입의 세션 모두 보여줍니다. BATCH 탭과 INTERACTIVE 탭은 각 탑입에 해당하는 세션들만 보여주는데, 이 때 종료된 세션은 해당하지 않습니다. 종료 탭에는 종료 된 세션 목록이 표시됩니다. 세션 별로 세션 환경, 할당 및 사용 자원량, 세션 시작 시간 등의 정보를 확인할 수 있습니다.

9.1.8 최근 기록

Backend.AI 24.09 버전 이후로 세션런처 페이지 우측 상단에 최근 생성된 세션 정보를 확인할 수 있는 최근 기록 기능이 추가되었습니다.



최근 기록 모달은 가장 최근에 생성된 5개의 세션에 대한 정보를 저장합니다. 생성을 원하는 기록을 선택한 후 '적용' 버튼을 클릭하면, 세션 생성의 마지막 단계인 검토 및 시작 페이지로 이동합니다.

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

참고: 간헐적인 네트워크 접속 불량 등의 문제로 세션 리스트가 정상적으로 표시되지 않는 경우가 발생할 수 있습니다. 이 때는 브라우저 페이지를 새로고침하면 해결할 수 있습니다.

9.2 세션 정보

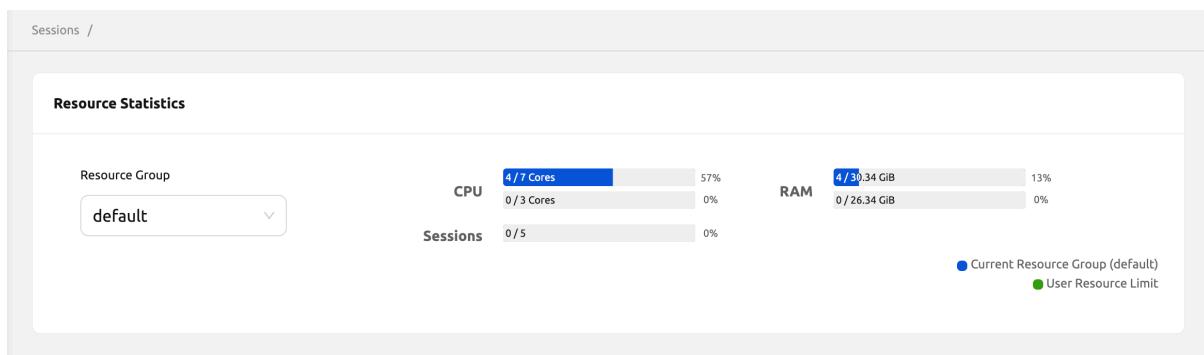
Backend.AI 는 PENDING, TERMINATED, CANCELLED 세션에 대한 상세 상태 정보를 제공합니다. 특히, PENDING 세션의 경우, 세션이 스케줄링 되지 않고 PENDING 상태에 머물러 있는 이유를 알 수 있어 유용합니다. 상세 정보를 확인하려면, 각 세션 상태를 나타내는 부분 바로 옆에 있는 '룰음표' 아이콘을 클릭하십시오.

The screenshot shows the Backend.AI WebUI interface with several windows open:

- Main Dashboard:** Shows a table of sessions with columns: RUNNING, INTERACTIVE, BATCH, INFERENCE, UPLOAD SESSIONS, FINISHED, and A session named "test-session" is highlighted, showing details like PyTorch 23.07, NVIDIA GPU Cloud, PYTORCH2.1, PY310, CUDA12.1.
- Status Info Popups:** Three separate windows titled "Status Info" provide detailed session status information:
 - Left Popup:** Status: predicate-checks-failed. Detail: Total Retries: 2. Last Try: 5/11/2022, 1:35:50 AM. Predicate Checks: 1 Failed, 6 Passed. Other items listed: reserved_time (Failed), concurrency (Passed), dependencies (Passed), keypair_resource_limit (Passed), user_group_resource_limit (Passed), domain_resource_limit (Passed), scaling_group_resource_limit (Passed).
 - Middle Popup:** Status: failed-to-start. Detail: Error: RuntimeError. Agent ID: [REDACTED]. Message: Cannot assign GPU(s) not validated by the license.
 - Right Popup:** Status: user-requested. Detail: Kernel Exit Code: -1. Session Status: terminated.

자원 통계는 화면 상단에 표시됩니다. 현재 사용중인 자원의 양과 할당 가능한 총 자원의 양을 확인할 수 있습니다. 디스플레이 바는 상단과 하단으로 나뉩니다. 상단은 현재 스케일링 그룹의 자원 할당 상태를 나타내고 하단은 접근 가능한 총 자원의 할당 상태를 나타냅니다.

- (현재 스케일링 그룹 내에서 사용자가 할당한 자원의 양) / (현재 스케일링 그룹 내에서 사용자 할당 가능한 자원의 총합)
- (사용자가 할당한 총 자원 양) / (사용자가 할당한 총 자원 양 + 현재 스케일링 그룹 내에서 사용자가 할당 가능한 자원의 총합)



참고: 만약 GPU 자원이 FGPU 로 표시되어 있다면, 이는 서버가 GPU 자원을 가상화된 형태로 제공하고 있다는 것을 의미합니다. Backend.AI 는 하나의 물리 GPU 를 여러 개로 분할해서 여러 사용자가 나누어 사용할 수 있는 가상화 기술을 지원하고 있습니다. 따라서, GPU 연산 소요가 크지 않은 작업을 수행하고자 할 경우에는 GPU 의 일부만 할당하여 연산 세션을 생성 할 수 있습니다. FGPU 가 실제로 할당하는 GPU 자원의 양은 관리자 설정에 따라 시스템 별로 다양할 수 있습니다.

예를 들어, 관리자가 하나의 GPU 를 다섯 조각으로 분할 설정한 경우, 5 FGPU 가 1 물리 GPU, 또는 1 FGPU 가 0.2 물리 GPU 를 뜻합니다. 이 때 1 FGPU 를 설정하여 연산 세션을 생성하면, 그 세션에서는 0.2 물리 GPU 에 해당하는 SM(streaming multiprocessor) 과 GPU 메모리를 활용할 수 있습니다.

9.2.1 세션 상세정보 패널

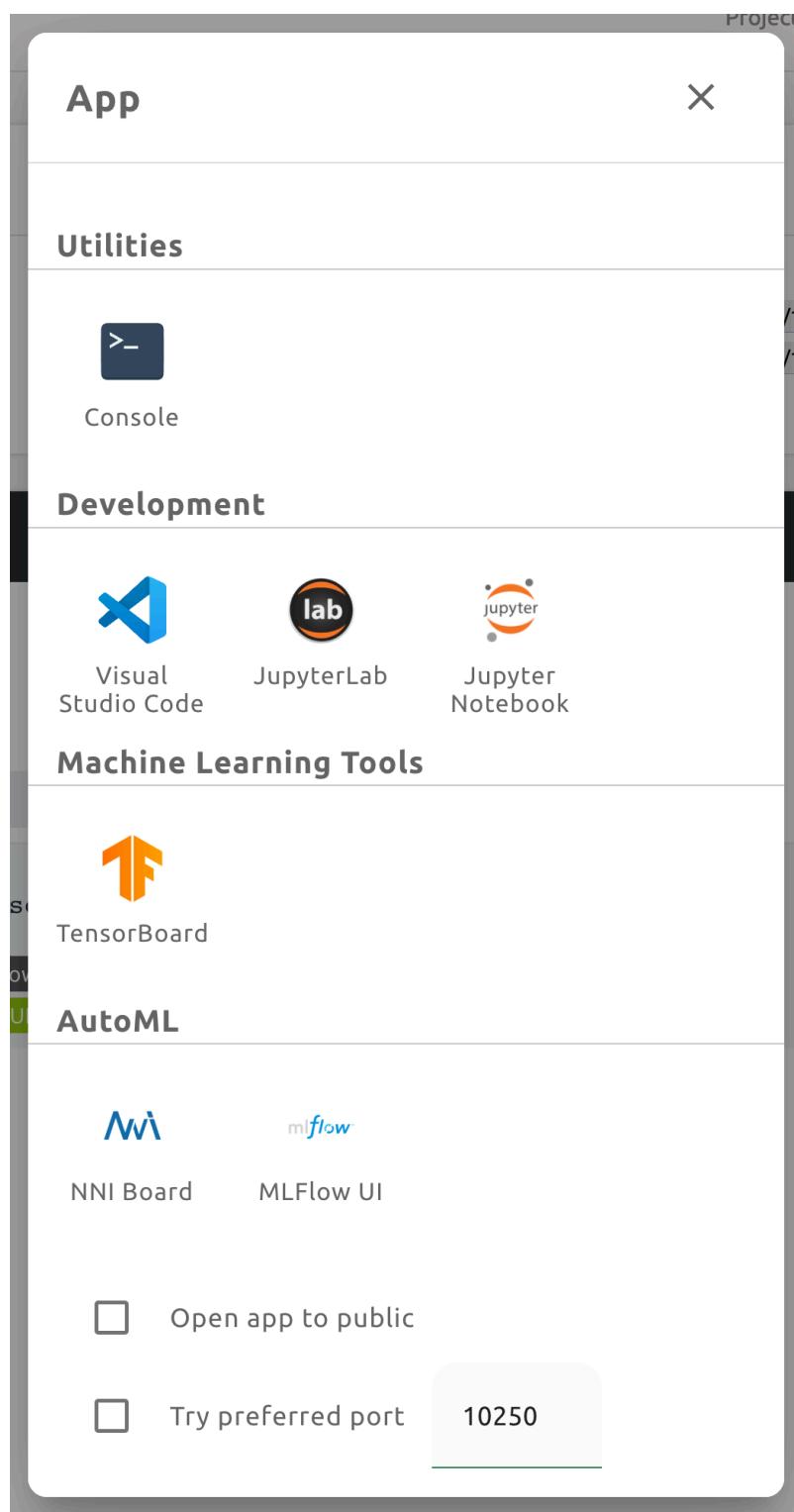
세션에 대한 상세정보를 확인하려면, 세션 리스트의 세션명을 클릭하십시오. 세션 디테일 페이지는 세션ID, 타입, 실행환경, 마운트 정보, 할당된 자원, 예약된 시간과, 경과 시간, 그리고 네트워크 I/O 를 포함한 자원 사용량과 같은 세션 정보를 보여줍니다.

The screenshot shows the Backend.AI WebUI interface. On the left, there's a sidebar with a 'Project' dropdown set to 'default'. The main area is titled 'Session Info' and displays information for a session named 'test-service1-191460df-78e6-4cb0-b266-6444de7db14f'. The session status is 'PENDING' with a note 'no-available-instances'. The session type is 'INFERENCE'. It lists environments ('Python 3.9', 'Python', 'aarch64'), resource allocation ('1 Core', '2 GiB'), and a reservation from 'Feb 4, 2025 5:29 PM' with an elapsed time of '16d 01:10:07'. Resource usage is shown as 'I/O Read: -B / Write: -B'. On the far left, a list of sessions is visible, showing entries 1 and 2.

Session ID	8541c5d4-2279-457c-9900-7e1014cc913f		
User ID	admin@lblup.com		
Status	PENDING no-available-instances	SessionType	INFERENCE
Environments	Python 3.9 Python aarch64	Mounted folders	model-folder
Resource allocation	1 Core 2 GiB	Agent	-
Reservation	Feb 4, 2025 5:29 PM Elapsed Time 16d 01:10:07		
Resource Usage	I/O Read: -B / Write: -B		

9.3 Jupyter Notebook 사용하기

이미 실행 중인 연산 세션을 사용하고 관리하는 법에 대해 살펴보겠습니다. 세션 리스트의 Control 열을 보면 몇 가지 아이콘이 있습니다. 제일 첫 아이콘을 클릭하면 다음 그림과 같이 앱 런처가 뜨면서 해당 세션이 지원하는 몇 가지 앱 서비스가 뜨게 됩니다. 앱 런처 대화창은 연산 세션 생성 직후 자동으로 열리기도 합니다.



참고: 앱 아이콘 아래에는 두 가지 체크 옵션이 있습니다. 각 항목을 체크하고 앱을 띄우면 다음과 같은 기능이 반영됩니다:

- 앱을 외부에 공개 : 앱을 외부에 공개합니다. 기본적으로 터미널 및 Jupyter Notebook 서비스와 같은 웹 서비스는 별도 인증을 거치므로 서비스 URL을 알고 있더라도 다른 사용자가 액세스 할 수 없습니다. 그러나 이 옵션을 선택하면 서비스 URL(및 포트 번호)을 아는

사람이 접근하고 사용할 수 있습니다. 물론 사용자가 서비스에 접근하려면 네트워크 경로가 있어야합니다.

- **선호 포트:** 이 옵션을 선택하지 않으면 Backend.AI에서 미리 준비한 포트 풀에서 웹 서비스의 포트 번호가 무작위로 할당됩니다. 이 항목을 체크하고 특정 포트 번호를 입력하면 입력한 포트 번호를 먼저 시도합니다. 그러나 포트가 포트 풀에 존재하지 않거나 다른 서비스가 이미 포트를 사용 중일 수 있기 때문에 원하는 포트가 항상 할당된다는 보장은 없습니다. 이 경우 포트 번호는 임의로 할당됩니다.

시스템 설정에 따라, 이 옵션들은 보이지 않을 수도 있습니다.

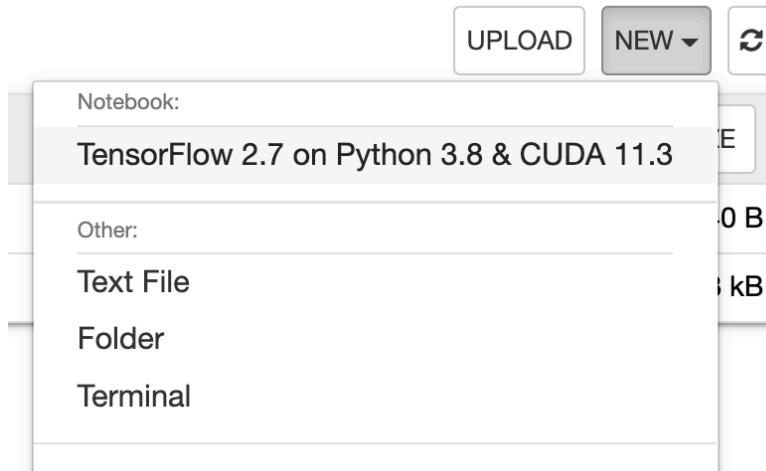
Jupyter Notebook 을 클릭해봅시다.

	NAME	LAST MODIFIED	FILE SIZE
<input type="checkbox"/>	DO_NOT_STORE_PERSISTENT_FILES_HERE.md	14일 전	540 B
<input type="checkbox"/>	id_container	3분 전	1.68 kB

새로운 창이 뜨면서 Jupyter Notebook 이 실행되는 것을 확인할 수 있습니다. 이 Notebook 은 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 사용 법은 공식 문서 등을 참고하시기 바랍니다.

Notebook 의 파일 탐색기에서 `id_container` 파일은 private SSH key 를 담고 있습니다. 필요할 경우 다운로드 하여 컨테이너로의 SSH/SFTP 접속에 이용할 수 있습니다.

우측 상단의 NEW 버튼을 클릭한 후 Backend.AI 용 Notebook 을 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.

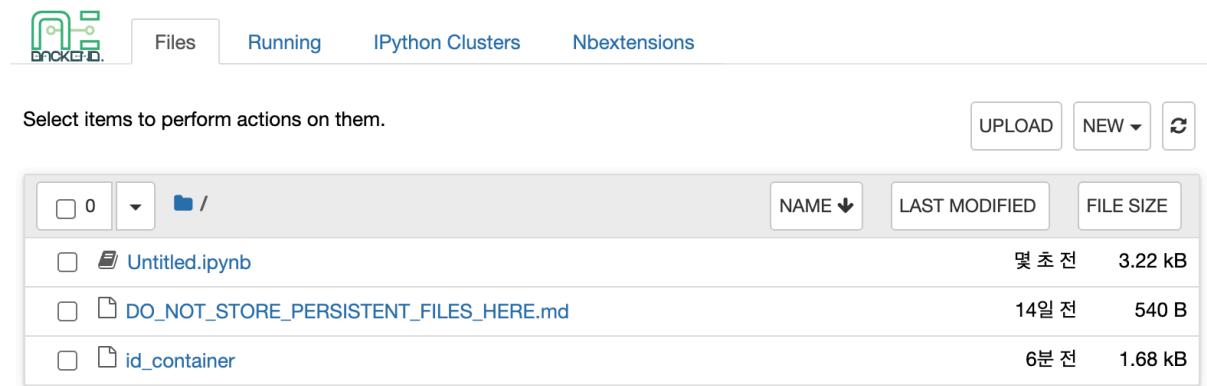
```
In [*]: import tensorflow as tf
mnist = tf.keras.datasets.mnist
(x_train, y_train), (x_test, y_test) = mnist.load_data()
x_train, x_test = x_train / 255.0, x_test / 255.0

model = tf.keras.models.Sequential([
    tf.keras.layers.Flatten(input_shape=(28, 28)),
    tf.keras.layers.Dense(128, activation='relu'),
    tf.keras.layers.Dropout(0.2),
    tf.keras.layers.Dense(10, activation='softmax')
])

model.compile(optimizer='adam',
              loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])
model.fit(x_train, y_train, epochs=20)
model.evaluate(x_test, y_test)

Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.npz
11493376/11490434 [=====] - 1s 0us/step
Epoch 1/20
1875/1875 [=====] - 3s 2ms/step - loss: 0.4814 - accuracy: 0.8590
Epoch 2/20
1875/1875 [=====] - 3s 1ms/step - loss: 0.1501 - accuracy: 0.9544
Epoch 3/20
1875/1875 [=====] - 3s 1ms/step - loss: 0.1089 - accuracy: 0.9678
Epoch 4/20
1028/1875 [=====>.....] - ETA: 1s - loss: 0.0855 - accuracy: 0.9726
```

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.



9.4 웹 터미널 활용

세션 목록 페이지로 돌아갑니다. 이번에는 터미널을 시작하겠습니다. 컨테이너의 tttyd 앱을 사용하여 터미널 아이콘(제어판의 두 번째 버튼)을 클릭합니다. 터미널이 새 창에 나타나고 다음 그림과 같이 셸 명령을 실행하여 연산 세션에 접근 할 수 있습니다. 명령 사용에 익숙하다면 다양한 Linux 명령을 쉽게 실행할 수 있습니다. Jupyter Notebook에서 자동으로 생성 된 Untitled.ipynb 파일이 ls 명령과 함께 나열되는 것을 볼 수 있습니다. 이는 두 앱이 동일한 컨테이너 환경에서 실행되고 있음을 보여줍니다.

```

work@main1[1Ty7ILKi-session]:~$ pwd
/home/work
work@main1[1Ty7ILKi-session]:~$ ipython
Python 3.8.8 (default, Feb 20 2021, 21:09:14)
Type 'copyright', 'credits' or 'license' for more information
IPython 7.16.1 -- An enhanced Interactive Python. Type '?' for help.

In [1]: import tensorflow as tf
2021-07-27 15:40:54.786984: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcudart.so.11.0

In [2]: tf.__version__
Out[2]: '2.4.1'

In [3]: exit
llwork@main1[1Ty7ILKi-session]:~$ ll
total 72
drwxr-xr-x 11 work work 4096 Jul 27 15:38 .
drwxr-xr-x  1 root root 4096 Jul 27 15:31 ..
-rw-r--r--  1 work work 147 Jul 27 15:31 .bash_profile
-rw-r--r--  1 work work 885 Jul 27 15:31 .bashrc
drwxr-xr-x  4 work work 4096 Jul 27 15:35 .cache
drwxr-xr-x  3 work work 4096 Jul 27 15:31 .config
drwxr-xr-x  2 work work 4096 Jul 27 15:36 .ipynb_checkpoints
drwxr-xr-x  5 work work 4096 Jul 27 15:31 .ipython
drwxr-xr-x  3 work work 4096 Jul 27 15:34 jupyter
drwxr-xr-x  3 work work 4096 Jul 27 15:37 keras
drwxr-xr-x  4 work work 4096 Jul 27 15:34 .local
drwx----- 2 work work 4096 Jul 27 15:31 .ssh
-rw-r--r--  1 work work 2066 Jul 27 15:31 .tmux.conf
-rw-r--r--  1 work work 501 Jul 27 15:31 .vimrc
drwxr-xr-x  2 work work 4096 Jul 27 15:35 virtual_documents
-rw-r--r--  1 work work 540 Jul 13 08:47 DO_NOT_STORE_PERSISTENT_FILES_HERE.md
-rw-r--r--  1 work work 3217 Jul 27 15:38 Untitled.ipynb
-rw----- 1 work work 1675 Jul 27 15:31 id_container
work@main1[1Ty7ILKi-session]:~$ 

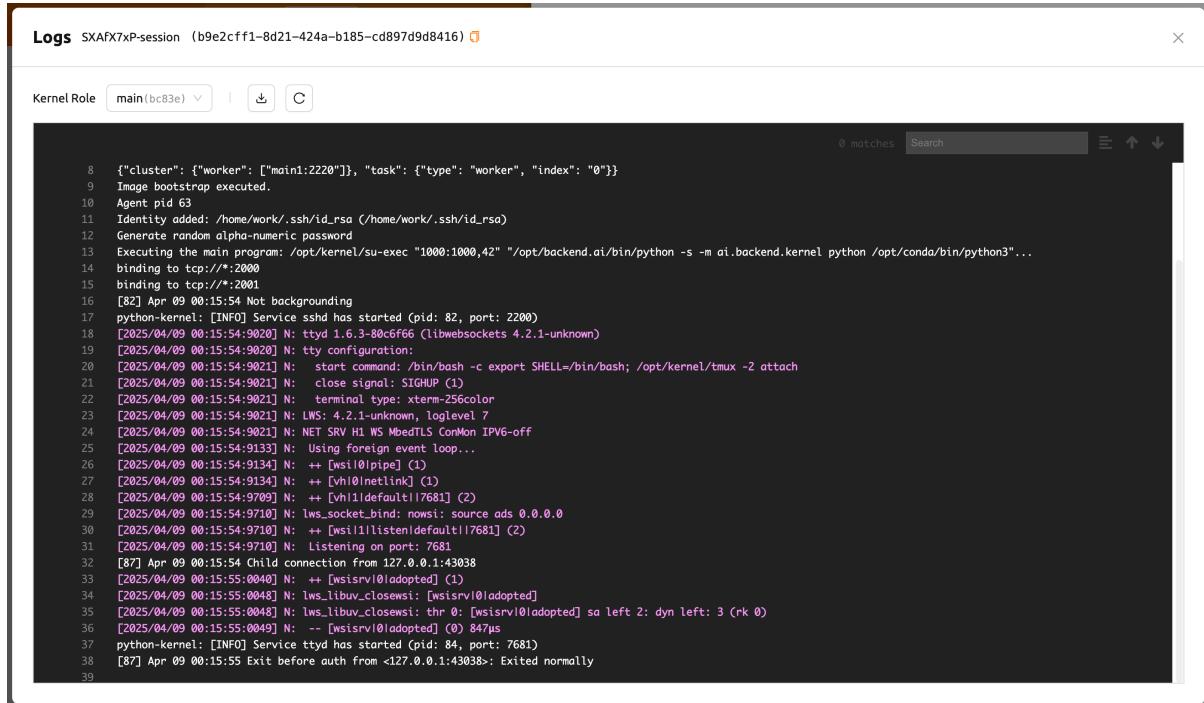
```

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.

9.5 연산 세션 로그 조회

돌아가고 있는 연산 세션의 Control 열의 마지막 아이콘을 클릭하면 연산 세션의 로그를 조회할 수 있습니다.



```

Logs SXAFX7xP-session (b9e2cff1-8d21-424a-b185-cd897d9d8416) 🔍

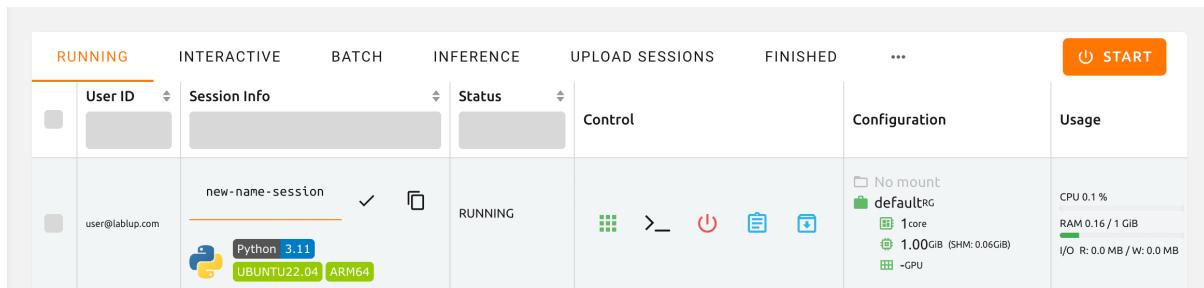
Kernel Role main(bc83e) | ⌂ C
0 matches Search ⌈ ⌋ ↑ ↓

8 {"cluster": {"worker": ["main1:2220"]}, "task": {"type": "worker", "index": "0"}}
9 Image bootstrap executed.
10 Agent pid 63
11 Identity added: /home/work/.ssh/id_rsa (/home/work/.ssh/id_rsa)
12 Generate random alpha-numeric password
13 Executing the main program: /opt/kernel/su-exec "1000:1000,42" "/opt/backend.ai/bin/python -s -m ai.backend.kernel python /opt/conda/bin/python3"...
14 binding to tcp://*:2000
15 binding to tcp://*:2001
16 [82] Apr 09 00:15:54 Not backgrounding
17 python-kernel: [INFO] Service sshd has started (pid: 82, port: 2200)
18 [2025/04/09 00:15:54:9020] N: tttyd 1.6.3-80c6f66 (libwebsockets 4.2.1-unknown)
19 [2025/04/09 00:15:54:9020] N: tty configuration:
20 [2025/04/09 00:15:54:9021] N: start command: /bin/bash -c export SHELL=/bin/bash; /opt/kernel/tmux -2 attach
21 [2025/04/09 00:15:54:9021] N: close signal: SIGHUP (1)
22 [2025/04/09 00:15:54:9021] N: terminal type: xterm-256color
23 [2025/04/09 00:15:54:9021] N: LWS: 4.2.1-unknown, loglevel 7
24 [2025/04/09 00:15:54:9021] N: NET SRV H1 WS MbedTLS ConMon IPV6-off
25 [2025/04/09 00:15:54:9133] N: Using foreign event loop...
26 [2025/04/09 00:15:54:9134] N: ++ [wsil0|pipe] (1)
27 [2025/04/09 00:15:54:9134] N: ++ [vh0@inetlink] (1)
28 [2025/04/09 00:15:54:9709] N: ++ [vh1|ldefault|17681] (2)
29 [2025/04/09 00:15:54:9710] N: lws_socket_bind: nowsi: source ads 0.0.0.0
30 [2025/04/09 00:15:54:9710] N: ++ [wsil1|llisten|default|17681] (2)
31 [2025/04/09 00:15:54:9710] N: Listening on port: 7681
32 [87] Apr 09 00:15:54 Child connection from 127.0.0.1:43038
33 [2025/04/09 00:15:55:0040] N: ++ [wsisrv|0|adopted] (1)
34 [2025/04/09 00:15:55:0048] N: lws.libuv_closeswi: [wsisrv|0|adopted]
35 [2025/04/09 00:15:55:0048] N: lws.libuv_closeswi: thr 0: [wsisrv|0|adopted] sa left 2: dyn left: 3 (rk 0)
36 [2025/04/09 00:15:55:0049] N: - [wsisrv|0|adopted] (0) 847µs
37 python-kernel: [INFO] Service tttyd has started (pid: 84, port: 7681)
38 [87] Apr 09 00:15:55 Exit before auth from <127.0.0.1:43038>: Exited normally
39

```

9.6 실행중인 세션 이름 변경하기

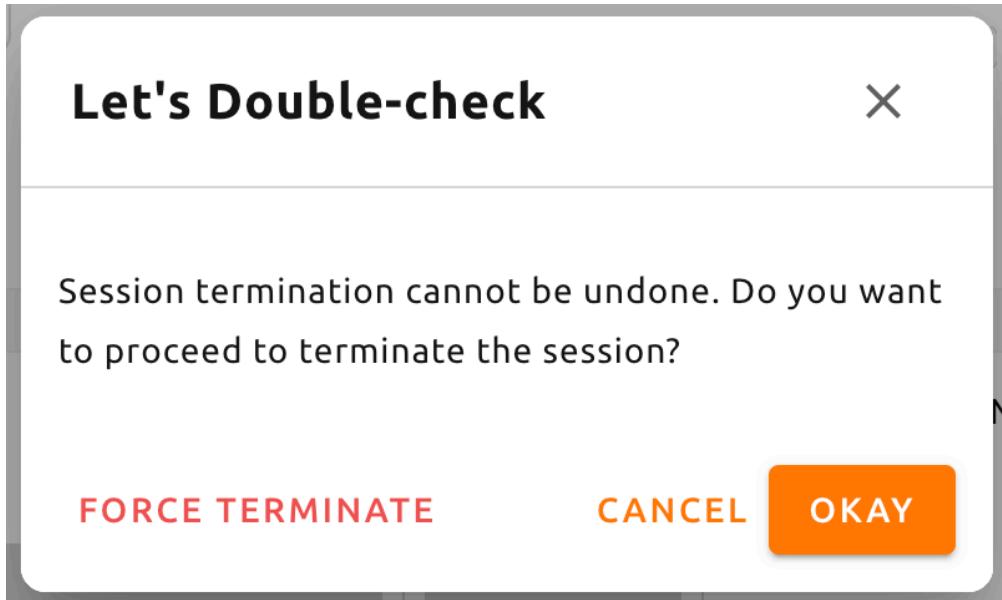
활성 상태에 있는 세션 이름을 변경할 수 있습니다. 세션 정보 열에 있는 편집 아이콘을 클릭한 후 새 이름을 입력하고 '확인' 버튼을 누르기만 하면 됩니다. 다만, 새로운 세션 이름도 [세션 이름 작성 규칙](#)은 따라야 합니다.



RUNNING	INTERACTIVE	BATCH	INFERENCE	UPLOAD SESSIONS	FINISHED	...	START
User ID	Session Info	Status	Control	Configuration	Usage		
	new-name-session	RUNNING	grid icon _ icon red power icon self icon copy icon	No mount default#RG 1core 1.00GB (SHM: 0.06GB) -GPU	CPU 0.1 % RAM 0.16 / 1 GiB I/O R: 0.0 MB / W: 0.0 MB		
	user@lablup.com	Python 3.11 UBUNTU22.04 ARM64					

9.7 연산 세션 삭제하기

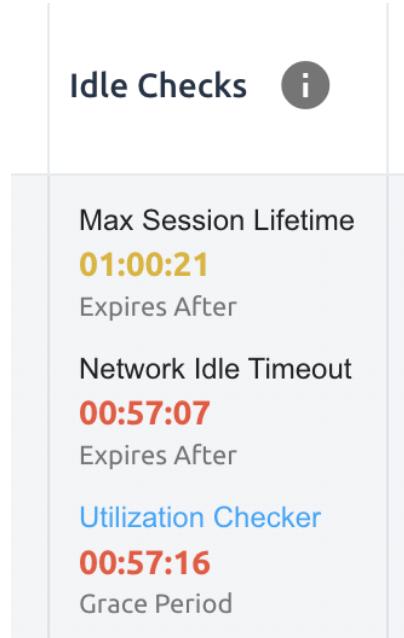
특정 세션을 삭제하기 위해서는 빨간색 '전원 아이콘'을 누르면 됩니다. 연산 세션 내부 폴더의 데이터는 연산 세션이 종료되는 순간 같이 삭제되므로, 계속 유지해야 하는 데이터라면 마운트 한 폴더에 옮기거나 처음부터 폴더에 업로드하는 것을 권합니다.



9.8 유휴 상태 검사

Backend.AI는 최대 세션 수명 시간, 네트워크 트래픽 기반 유휴 시간, 사용량 기반 자원 수거를 기준으로 세션이 자동으로 삭제될 수 있습니다.

연산 세션 종료 기준은 유휴 상태 검사 컬럼에서 확인할 수 있습니다.



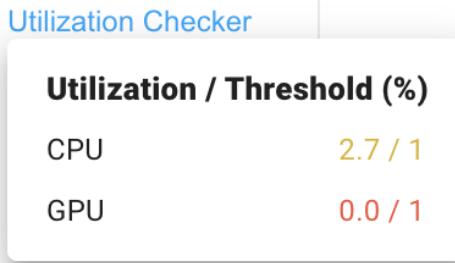
각 항목의 의미는 다음과 같으며, 우측의 정보(i) 버튼을 클릭해서 자세한 설명을 확인할 수도 있습니다.

- 최대 세션 수명 시간: 세션 생성 후 이 시간이 지나면 세션을 강제 종료합니다. 이는 세션이 무한히 실행되는 것을 방지하기 위한 조치입니다.
- 네트워크 트래픽 기반 유휴 시간: 사용자(브라우저)와 연산 세션 사이에 이 시간 동안 아무런 네트워크 트래픽이 없을 경우 연산 세션을 삭제합니다. 사용자와 연산 세션 사이의 트래픽은 앱 (터미널, Jupyter 등)을 통해 상호작용(키보드 입력, Jupyter 셀 생성 등) 하는 경우 지속적으로 발생합니다. 만약, 연산 세션을 띄운 채 일정 시간 동안 아무런 입력을 하지 않으면 자동 삭제 조건을 만족하게 됩니다. 연산 세션에서 작업을 수행 중인 프로세스가 있더라도, 사용자와의 상호작용이 없는 경우에는 삭제 대상입니다.
- 사용량 기반 자원 수거: 연산 세션에 할당된 자원을 자원의 활용률을 기준으로 회수합니다. 연산 세션의 삭제 여부는 다음 두 가지 요소에 따라 결정됩니다:
 - 유예 기간: 이 기간 동안은 자원 사용량 기반 체커가 작동하지 않습니다. 즉, 유예 기간이 지나기 전까지는 사용량이 기준보다 낮더라도 연산 세션이 삭제되지 않습니다. 하지만 유예 기간이 지나고 나면, 설정된 유휴 시간(idle timeout) 동안의 평균 자원 사용률이 기준에 미치지 못하는 경우 해당 세션이 삭제될 수 있습니다. 유예 기간은 세션 종료가 이루어지지 않는 것을 보장하는 시간일 뿐입니다. 이는 주로 사용률이 낮은 GPU 자원을 효율적으로 관리하기 위한 조치입니다.
 - 자원 사용량 기준: 연산 세션의 자원 사용량이 일정 시간(idle timeout) 동안 설정된 기준값을 넘지 못하면, 해당 세션은 자동으로 삭제됩니다. 예를 들어, 가속 장치 사용률 기준을 1%로 설정했다면 idle timeout 시간 동안 평균 가속 장치 사용률이 1% 미만인 연산 세션은 삭제 대상이 됩니다. 값이 설정되지 않는 자원은 자동 삭제 기준에서 제외

됩니다.

참고: 유예 기간이 지난 후에는 사용량이 낮으면 언제든지 삭제될 수 있습니다. 자원을 잠깐 사용했다고 해서 유예 기간이 연장되지 않습니다. 오직 현 시점으로부터 지난 idle timeout 시간 동안의 평균 자원 사용률만이 고려됩니다.

사용량 기반 자원 수거에 마우스를 가져가면, 사용률과 수거 기준값을 보여주는 툴팁이 나타납니다. 현재 사용률이 수거 기준값에 접근할수록(사용량이 저조할수록) 글자 색이 노란색, 빨간색 순으로 변하게 됩니다.



참고: 환경 설정 값에 따라, 유휴 상태 검사 기준과 사용량 기반 자원 수거 툴팁에 나타나는 자원 항목이 다르게 보일 수 있습니다.

9.9 세션 생성하기 전에 환경 변수를 추가하는 방법

Backend.AI에서는 사용자에게 더 편리한 환경을 제공하기 위해, 세션 시작 화면에서 세션에 추가될 환경 변수 추가 기능을 지원합니다. 이 기능에서 여러분은 PATH 를 비롯한 모든 환경 변수를 환경 변수 설정 다이얼로그에서 환경 변수명과 환경 변수 값을 입력해서 추가할 수 있습니다.

더 많은 환경 변수를 추가하고 싶을 경우, 입력 필드 첫번째 행의 오른쪽에 있는 '+ Add environment variables' 버튼을 클릭하면 됩니다. 또한 환경 변수를 지우고 싶을 경우, 역시 지우고자 하는 행의 '-' 버튼을 클릭하면 됩니다.

Variable (optional)

Variable	Value	⊖
----------	-------	---

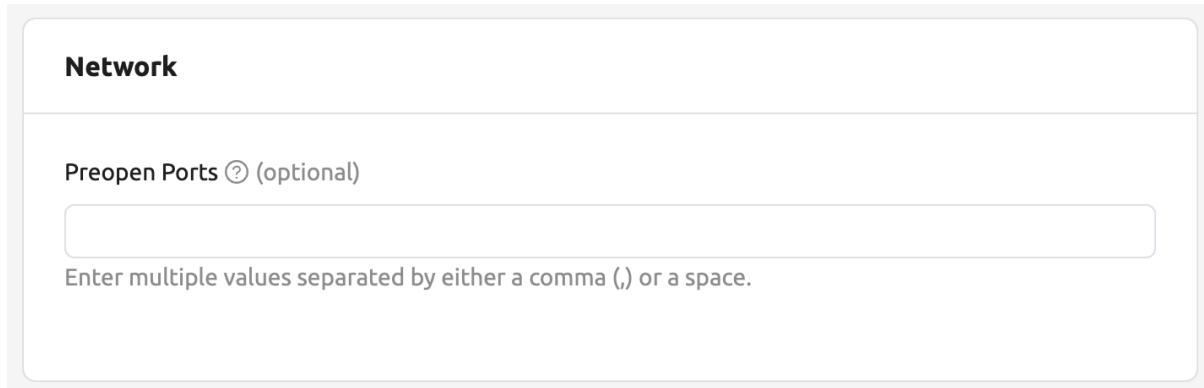
+ Add environment variables

환경 변수 명과 값을 같은 행의 입력 필드에 입력할 수 있습니다.

9.10 세션 생성하기 전에 사전 개방 포트를 추가하는 방법

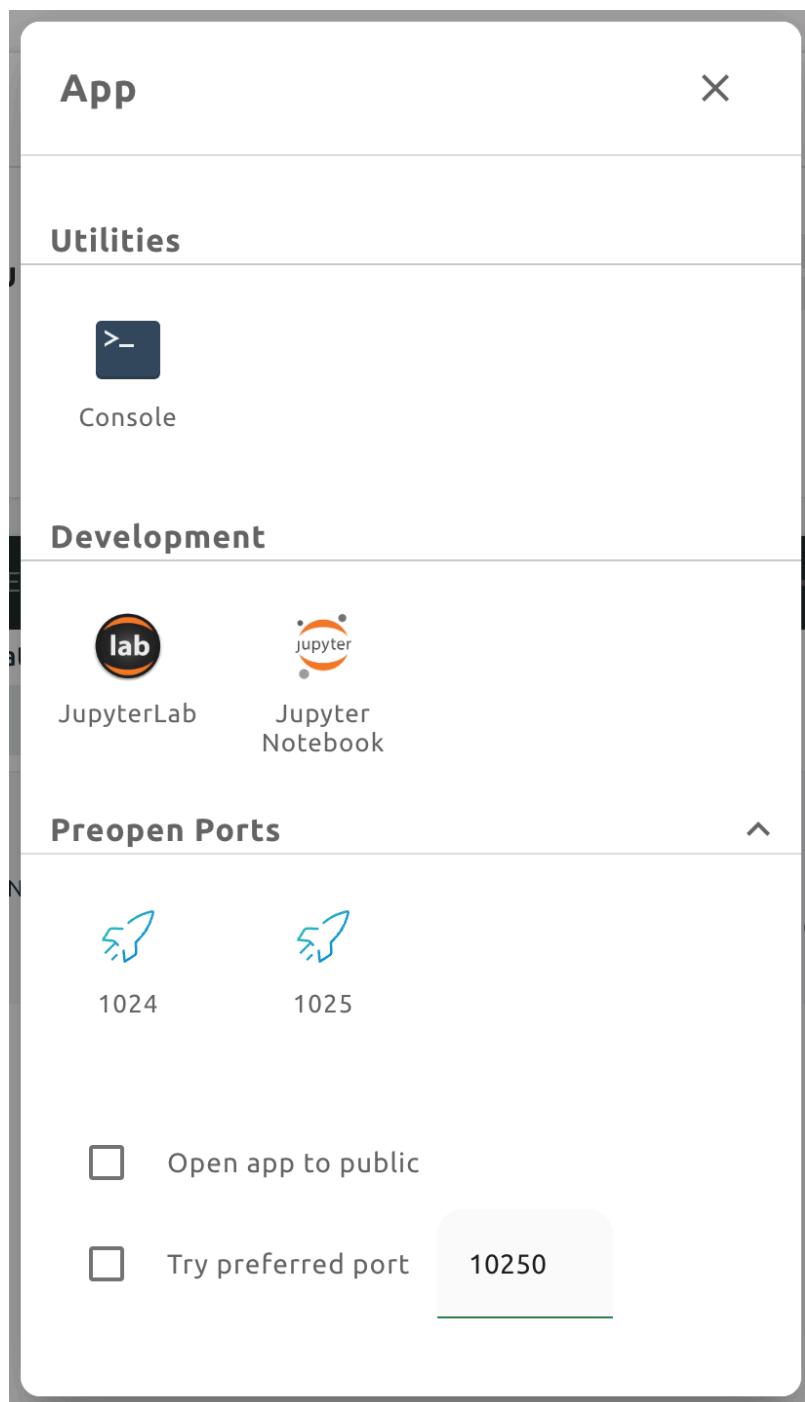
Backend.AI는 컨테이너 시작 전 사전 개방 포트를 설정하는 것을 지원합니다. 이 기능을 사용하면, 서빙 포트를 노출하기 위해 별도의 이미지를 추가로 빌드할 필요가 없습니다.

사전 개방 포트를 추가하려면 쉼표(,)나 공백으로 구분하여 여러 값을 입력할 수 있습니다.



해당 다이얼로그에서 사전 개방 포트를 추가하거나, 작성한 사전 개방 포트를 갱신, 삭제할 수 있습니다. 더욱 자세한 설명이 필요한 경우, 다이얼로그 헤더 부분에 있는 '도움말 (?)' 버튼을 클릭해주세요.

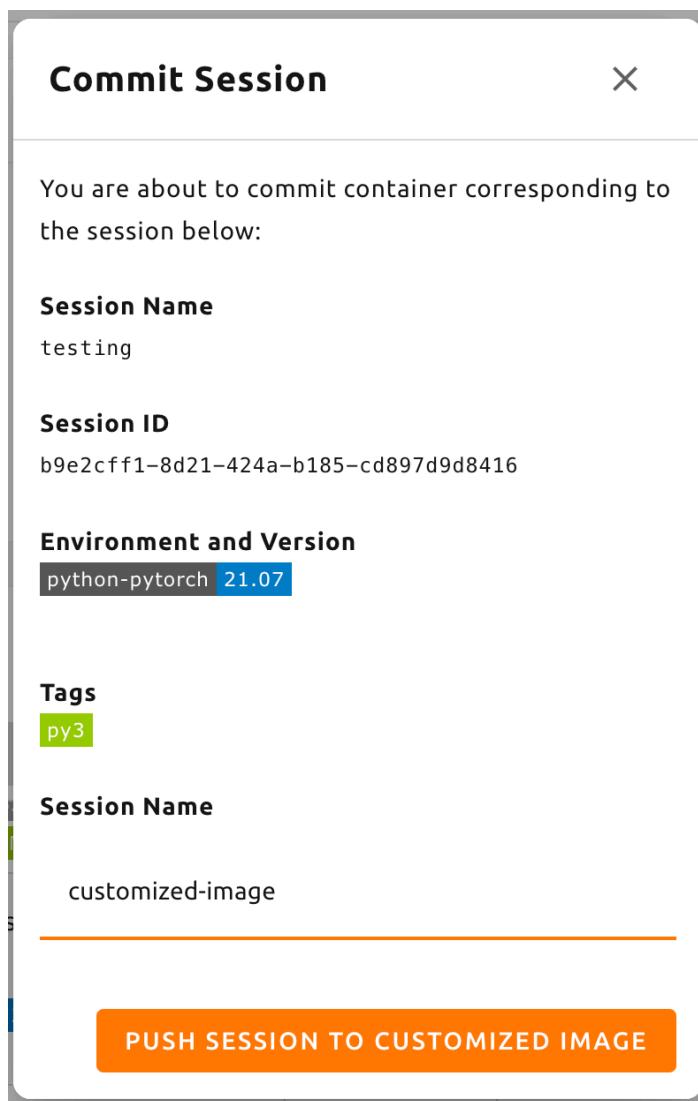
입력란에 1024 ~ 65535 사이의 포트값을 입력한 뒤, 엔터 키를 누르세요. 각 포트는 쉼표(,)로 구분되며, 여러 포트를 설정할 수 있습니다. 설정된 사전 개방 포트값은 세션 앱 런처에서 확인할 수 있습니다.



참고: 사전 개방 포트는 **컨테이너 내부 포트**입니다. 따라서, 다른 앱들과 달리 세션 앱 런처에서 사전 개방 포트를 클릭하면 빈 페이지가 나타납니다.

9.11 세션 커밋 저장하기

Backend.AI는 24.03 버전부터 “세션을 이미지로 변환하는 기능”을 지원합니다. RUNNING 세션을 커밋하면 세션의 현재 상태가 새 이미지로 저장됩니다. RUNNING 세션의 제어 열에 있는 커밋 버튼을 클릭하면 세션 정보를 보여주는ダイ얼로그가 표시됩니다. 세션 이름을 입력한 후 새 이미지로 세션을 변환할 수 있습니다. 세션 이름은 4자에서 32자까지의 알파벳 문자, 하이픈(-), 밑줄(_)만 포함해야 합니다.



입력 필드에 세션 이름을 작성한 후 '세션을 사용자 정의된 이미지로 푸시' 버튼을 클릭하세요. 이 방법으로 생성된 사용자 정의 이미지는 향후 세션 생성에서 사용할 수 있습니다. 그러나 이미지 커밋을 위해 컨테이너에 마운트된 디렉토리는 외부 리소스로 간주되어 최종 이미지에 포함되지 않습니다. /home/work 디렉토리가 마운트된 폴더(스크래치 디렉토리)임을 기억해주세요.

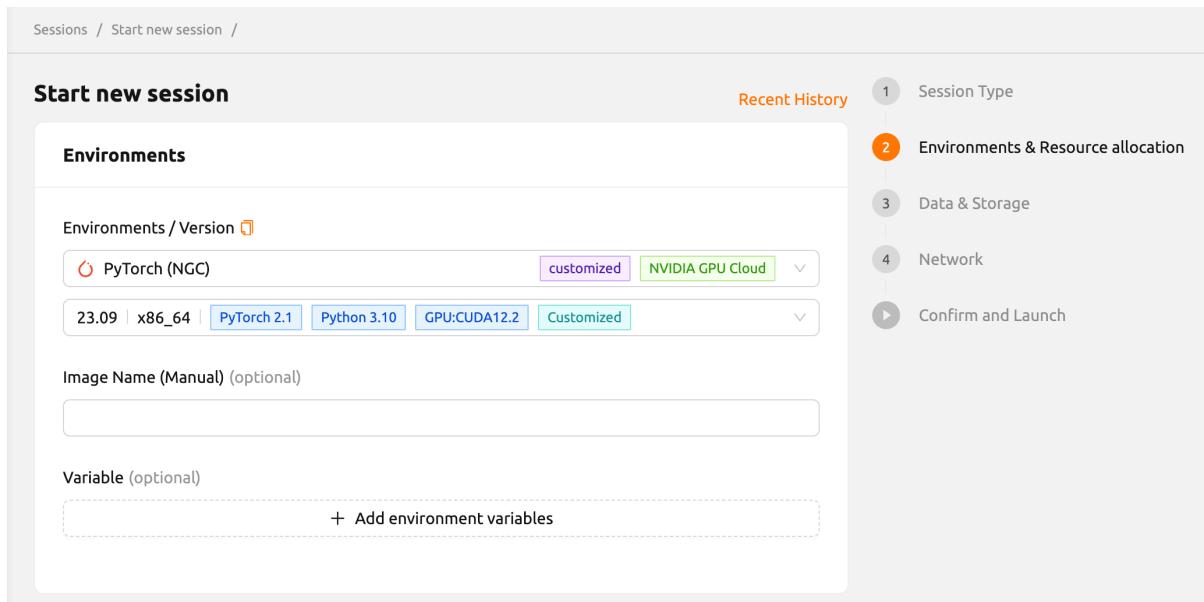
참고: 현재 Backend.AI는 세션이 INTERACTIVE 모드일 때만 “세션을 이미지로 변환” 기능을 지원합니다. 커밋 프로세스 중에 예기치 않은 오류를 방지하기 위해, 세션 종료를 명령하더라도

실제로는 종료되지 않을 수 있습니다. 진행 중인 프로세스를 중지하려면 세션을 확인하고 강제로 종료하세요.

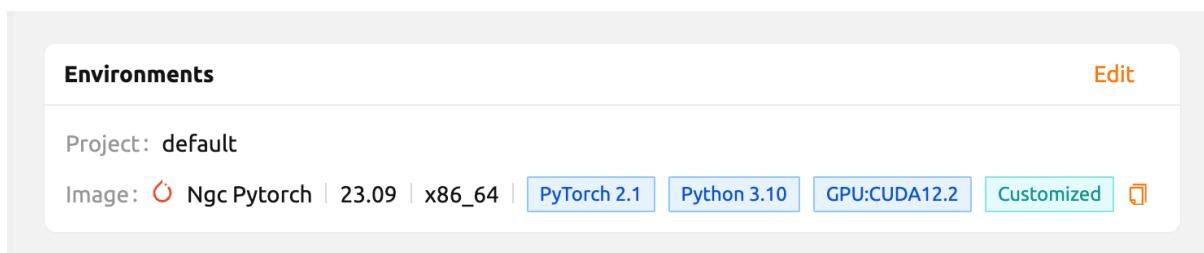
참고: 사용자 자원 정책에 의해 “세션을 이미지로 변환” 작업 횟수가 제한될 수 있습니다. 이 경우, 기존 사용자 정의 이미지를 제거 한 후 다시 시도하거나 관리자에게 문의하세요.

9.12 진행 중인 세션의 변환된 이미지 활용

진행 중인 세션을 이미지로 변환하면, 다음 세션 생성 시 세션 런처의 환경 설정에서 해당 이미지를 선택하여 사용할 수 있습니다. 이 이미지는 다른 사용자에게 공개되지 않으며, 현재 세션의 상태를 그대로 유지하여 사용하는 데 유용합니다. 변환된 이미지는 `Customized<session name>` 태그가 붙습니다.



향후 세션 생성을 위해 환경 이름을 수동으로 입력하려면, 복사 아이콘을 누릅니다.



9.13 웹 터미널 고급 사용법

위에서 사용 해보았던 웹 기반 터미널은 내부적으로 tmux 라는 유ти리티를 사용하고 있습니다. tmux 는 하나의 쉘 내에서 여러 개의 창을 띄워 다중 작업 할 수 있도록 지원하는 terminal multiplexer 로, 쉘이 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.

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 쉘을 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 진입 합니다. 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 드래그하고 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.

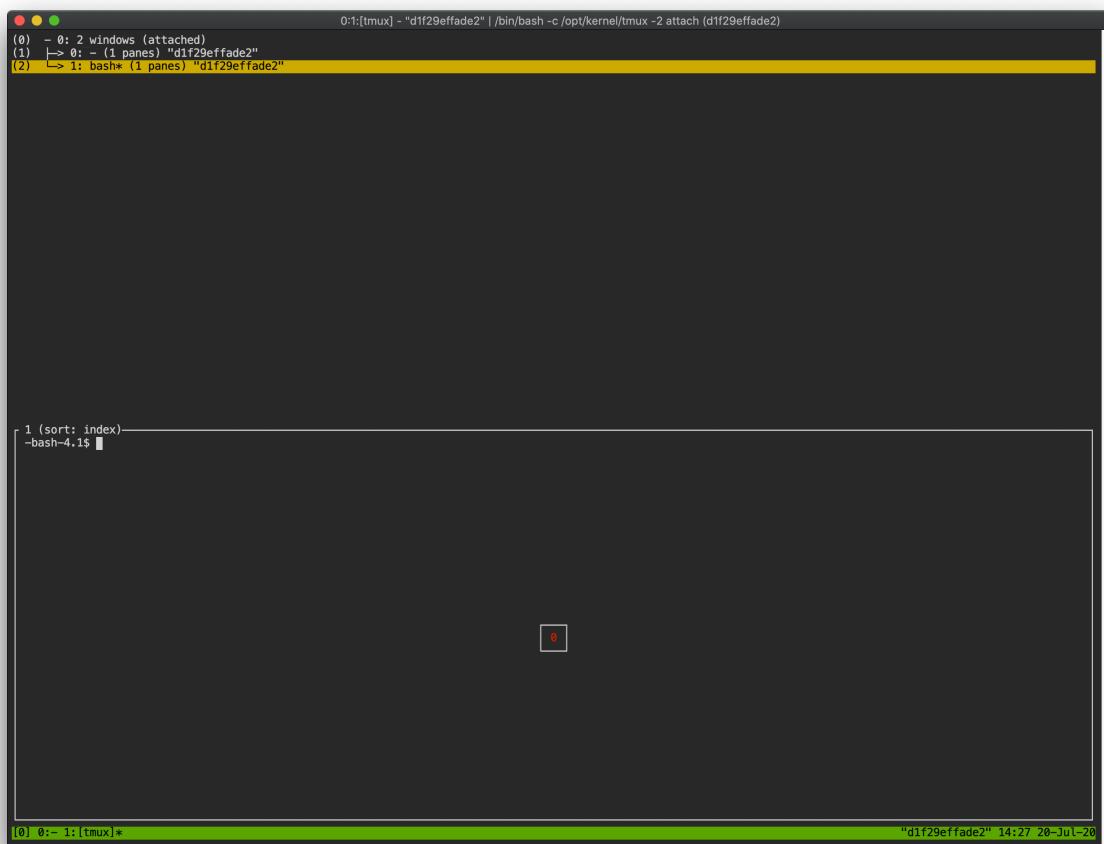
참고: Ctrl-B 키는 tmux 의 기본 제어 키입니다. 만약 홈 디렉토리의 .tmux.conf 를 수정하여 다른 제어 키를 설정한 경우에는, Ctrl-B 대신 설정된 키 조합을 눌러야 합니다.

참고: 윈도우즈 환경에서는 다음 단축키를 참고하세요.

- 복사: Shift 키를 누른 상태에서 마우스 우클릭해서 드래그
 - 붙여넣기: Ctrl-Shift-V 키를 누름
-

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.

tmux 의 가장 큰 장점은 하나의 터미널 창에서 여러 개의 쉘을 띄우고 사용할 수 있다는 점입니다. 백문이 불여일견이니 Ctrl-B 키를 누른 후 c 키를 이어서 눌러봅시다. 기존 창의 내용이 사라지고 새로운 쉘 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 쉘 리스트가 조회되는 것을 볼 수 있습니다. 여기서 0: 으로 시작하는 쉘이 처음 보던 쉘 환경이고, 1: 로 시작하는 쉘은 방금 새로 생성한 쉘입니다. 위/아래 방향 키를 이용해서 쉘 사이를 이동할 수 있습니다. 0: 번 쉘에 커서를 가져다 두고 엔터 키를 눌러 선택 해보겠습니다.



첫 번째 쉘 환경이 나타나는 것을 볼 수 있습니다. 이러한 방식으로 웹 터미널 내에서 여러 쉘 환경을 사용할 수 있습니다. 현재 쉘을 종료하려면 exit 명령을 입력하거나 Ctrl-B x 키를 누른 다음 y 를 입력하십시오.

정리하면 다음과 같습니다:

- Ctrl-B c: 새로운 tmux 쉘 생성

- Ctrl-B w: tmux 쉘 조회 및 이동/선택
- exit 또는 Ctrl-B x: 현재 tmux 쉘 종료

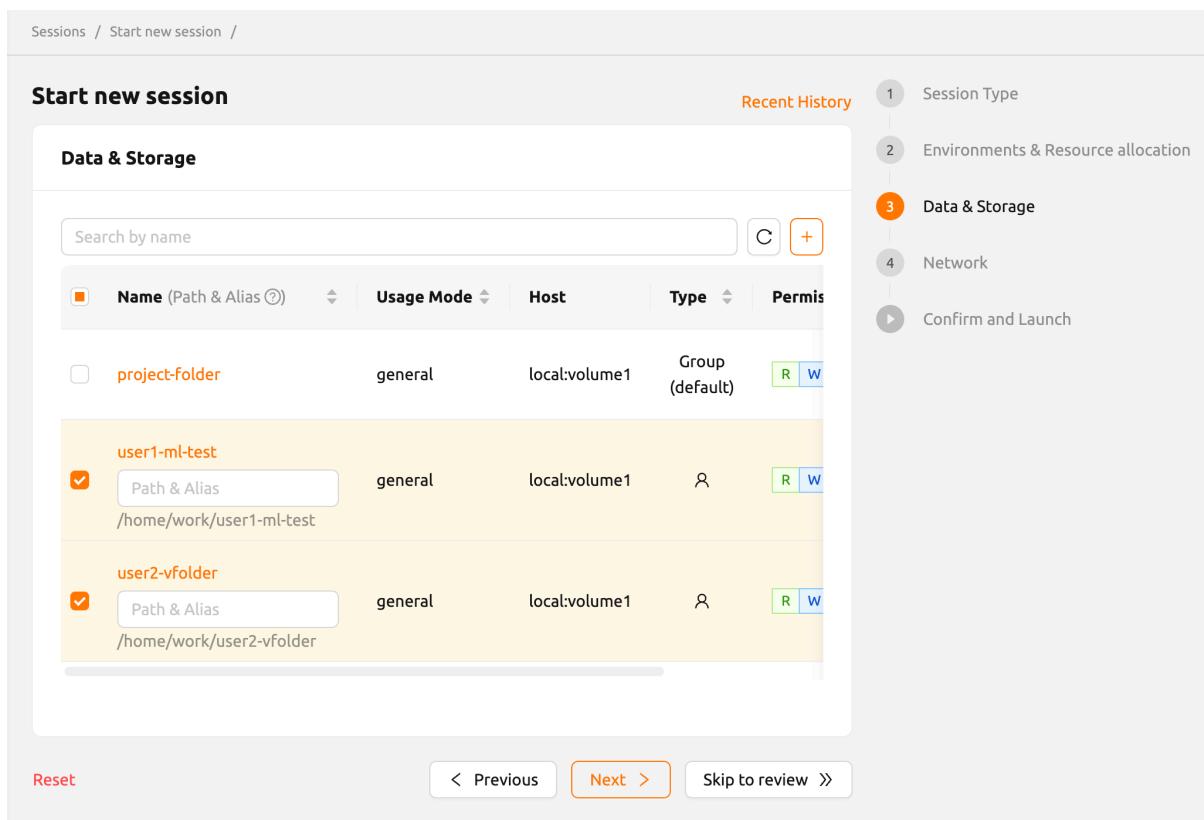
위 명령을 조합하여 여러 개의 쉘에서 동시에 다양한 작업을 수행할 수 있습니다.

CHAPTER 10

연산 세션에 폴더 마운트

Backend.AI는 스토리지 폴더 연산 세션 생성 시 마운트 할 수 있는 기능을 제공합니다. 연산 세션을 시작하면 사용자는 `/home/work/` 폴더에 접근 권한을 가지게 되는데, `/home/work/` 및 그 하위에 생성된 일반 폴더와 파일은 연산 세션이 삭제되면 함께 사라집니다. 이는 연산 세션이 컨테이너에 기반하여 동적으로 생성/삭제 되기 때문으로, 컨테이너 내부의 데이터를 컨테이너 수명 주기와 상관없이 보존하기 위해서는 별도의 호스트 폴더를 컨테이너에 마운트 한 뒤, 마운트 된 폴더 안에 파일을 생성해야 합니다.

'세션' 페이지에서 '시작' 버튼을 누르면 새로운 세션을 생성할 수 있습니다. '세션 타입', '실행 환경 & 자원 할당' 단계를 작성한 후, '데이터 & 폴더' 단계로 이동하면 사용자가 마운트할 수 있는 폴더의 리스트를 확인할 수 있습니다. 해당 리스트에서 마운트 하고자 하는 폴더를 선택하여 추가할 수 있으며, 여러 개의 폴더를 선택하면 다수의 폴더를 마운트 할 수 있습니다. 해당 문서에서는 `user1-ml-test` 와 `user2-vfolder` 두 개의 폴더를 마운트한 후 세션을 생성하는 법에 대해 설명합니다.

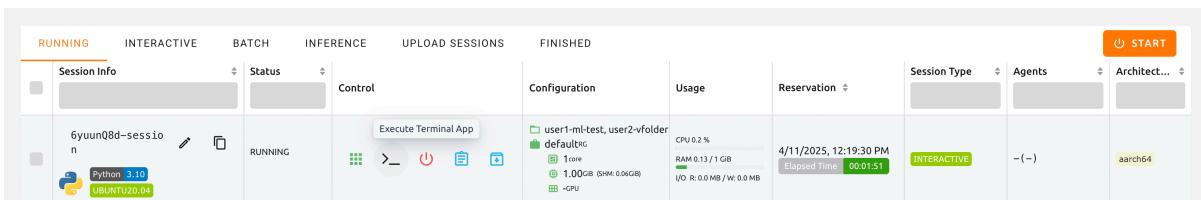


참고: 해당 프로젝트 내의 데이터와 폴더 정보를 살펴보면 사용 모드, 해당 폴더가 속해 있는 스토리지 호스트, 권한 등의 정보를 확인할 수 있습니다. 참고로, '데이터 & 폴더' 단계에서는 현재 사용자가 마운트 가능한 데이터 폴더만 출력됩니다. 다른 프로젝트에 속한 폴더는 해당 메뉴를 통해 조회할 수 없습니다.

참고: '데이터 & 폴더' 단계에서 폴더 이름을 클릭하면, '폴더' 정보를 한 눈에 확인할 수 있습니다. 해당 메뉴에서는 생성되어 있는 폴더를 확인하고, 새로운 폴더를 생성하고, 파일을 업로드할 수 있습니다. 자세한 설명은 [폴더 내용 조회하기](#)를 참조하세요.

여기에 더해, '+' 버튼을 클릭하여 새로운 폴더를 생성할 수도 있습니다. 세션 런처에서 새로운 폴더를 생성하는 자세한 방법을 확인하시려면, [Storage 폴더 생성](#) 을 참조하세요.

생성된 세션에서 터미널 아이콘을 클릭하면 터미널 창을 띄울 수 있습니다. 터미널에서 `ls` 명령을 내리면 `user1-ml-test` 폴더와 `user2-vfolder` 폴더가 홈 디렉토리 하위에 마운트되는 것을 확인할 수 있습니다.



참고: 선택한 폴더는 기본적으로 연산 세션 내의 /home/work/ 하위에 폴더의 이름으로 마운트 됩니다. 예를 들어, 폴더의 이름이 test 일 경우, /home/work/test 에 마운트 됩니다. 만약, 연산 세션 내 마운트 경로를 변경하고 싶다면 '경로 & 대체 이름' 입력 필드에 절대 경로를 입력합니다. 만약, test 폴더의 마운트 경로를 /workspace 로 입력한 경우에는 test 폴더가 연산 세션 내 /workspace 경로에 마운트 됩니다. 상대 경로를 입력하면 /home/work/ 아래에 마운트 됩니다.

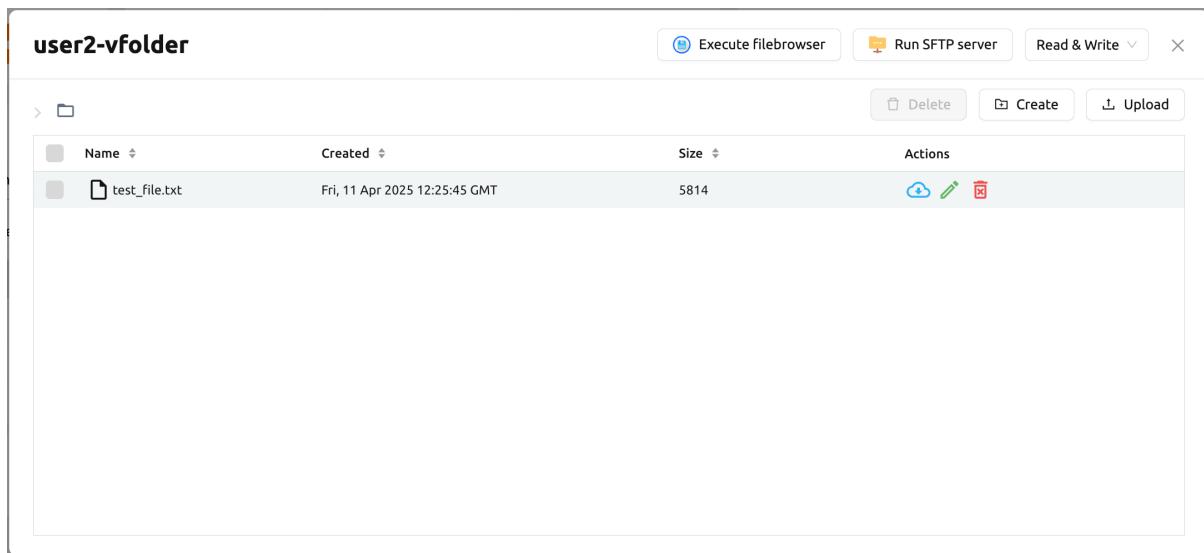
Backend.AI에서는 연산 세션을 삭제해도 폴더 안의 파일을 보존할 수 있는 옵션을 제공합니다. 아래 예제를 통해 관련된 내용을 확인할 수 있습니다.

우선, user2-vfolder 하위에 test_file 파일을 생성합니다. 해당 파일의 내용은 《file inside user2-vfolder》로 채우겠습니다.

```
work@main1[6yuunQ8d-session]:~$ ls
DO_NOT_STORE_PERSISTENT_FILES_HERE.md  id_container  user1-ml-test  user2-vfolder
work@main1[6yuunQ8d-session]:~$ ls user2-vfolder/
work@main1[6yuunQ8d-session]:~$ echo "file inside user2-vfolder" > user2-vfolder/test_file
work@main1[6yuunQ8d-session]:~$ ls user2-vfolder/
test_file
work@main1[6yuunQ8d-session]:~$ cat user2-vfolder/test_file
file inside user2-vfolder
work@main1[6yuunQ8d-session]:~$
```

user2-vfolder 를 대상으로 ls 명령을 실행해보면 파일이 생성된 것을 확인할 수 있습니다. 참고로, cat 명령을 사용하면 파일 내용을 확인할 수 있습니다.

이제 세션을 삭제한 후 Data & Storage 페이지로 이동합니다. user2-vfolder 폴더를 찾아 클릭하여 파일 탐색기를 열고 test_file 파일이 존재하는 것을 확인합니다. '작업' 탭의 파일 다운로드 버튼을 클릭하여 파일을 로컬 머신에 내려받은 후, 내용이 "file inside user2-vfolder"임을 확인합니다.



연산 세션을 생성할 때 폴더를 마운트한 후, 마운트 된 폴더에서 파일 작업을 수행하면 연산 세션이 종료된 이후에도 데이터를 보존할 수 있습니다.

10.1 자동 마운트 폴더를 이용하여 연산 세션 환경 구성하기

연산 세션에 사전 설치되어 있지 않은 새 프로그램이나 라이브러리를 필요로 하는 경우, 연산 세션의 수명 주기와 관계 없는 Storage 폴더 특성과 [자동 마운트 폴더](#)를 이용해서 패키지를 설치하고, 연산 세션의 종류에 상관없이 일정한 환경을 구성할 수 있습니다.

10.1.1 pip를 통한 파이썬 패키지 설치

.local이라는 이름의 폴더를 생성하면 파이썬 사용자 패키지를 해당 폴더에 설치하고 공유할 수 있습니다. pip `로 패키지를 설치하면 사용자 홈 폴더 하위에 있는 ``.local 폴더에 패키지를 설치하기 때문입니다. (자동 마운트 폴더는 사용자 홈폴더 하위에 마운트 됩니다.) 따라서, 연산 환경과 상관없이 ``tqdm``이라는 파이썬 패키지의 설치 상태를 항상 유지하려면 터미널에서 다음과 같은 명령을 내릴 수 있습니다.

```
pip install 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. 이를 해결하기 위해, .bashrc 편집기능을 통해 PYTHONPATH

를 분기하는 방식의 적용이 가능합니다. 해당 방법을 사용하면, PYTHONPATH 환경 변수에 지정된 경로에 사용자 pip 패키지를 설치할 수 있습니다.

10.1.2 Homebrew를 통해 패키지 설치

Ubuntu의 apt 나 CentOS의 yum 같은 패키지 매니저는 일반적으로 root 권한을 필요로 합니다. Backend.AI의 연산 세션은 보안을 위해 sudo 권한 및 root 접근이 기본적으로 막혀 있기 때문에 (설정에 따라 접근이 허용될 수도 있습니다), 패키지 설치를 위해 sudo 접근이 필요하지 않은 [Homebrew on Linux](#) 사용을 권장하고 있습니다.

Homebrew는 다음과 같이 구성할 수 있습니다.

- Data & Storage 페이지에서 .linuxbrew 폴더 생성
- 연산 세션을 생성(.linuxbrew 폴더가 자동으로 /home/linuxbrew/.linuxbrew 위치에 마운트)
- 연산 세션 내에서 Homebrew 설치(아직 설치하지 않은 경우)

```
$ /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

- Homebrew 패키지는 다음과 같이 설치할 수 있습니다.

```
$ brew install hello
$ hello
Hello, world!
```

brew 는 패키지를 /home/linuxbrew/.linuxbrew 디렉토리 하위에 설치하는데, .linuxbrew 라는 데이터 폴더가 있다면 연산 세션 생성 시 자동으로 그 위치에 마운트합니다. 따라서, .linuxbrew 라는 자동 마운트 폴더를 생성해두면 연산 세션을 삭제하고 새로운 연산 세션을 생성하더라도 기존에 설치했던 Homebrew 패키지를 다시 사용할 수 있습니다.

CHAPTER 11

저장 폴더 공유 및 접근 제어

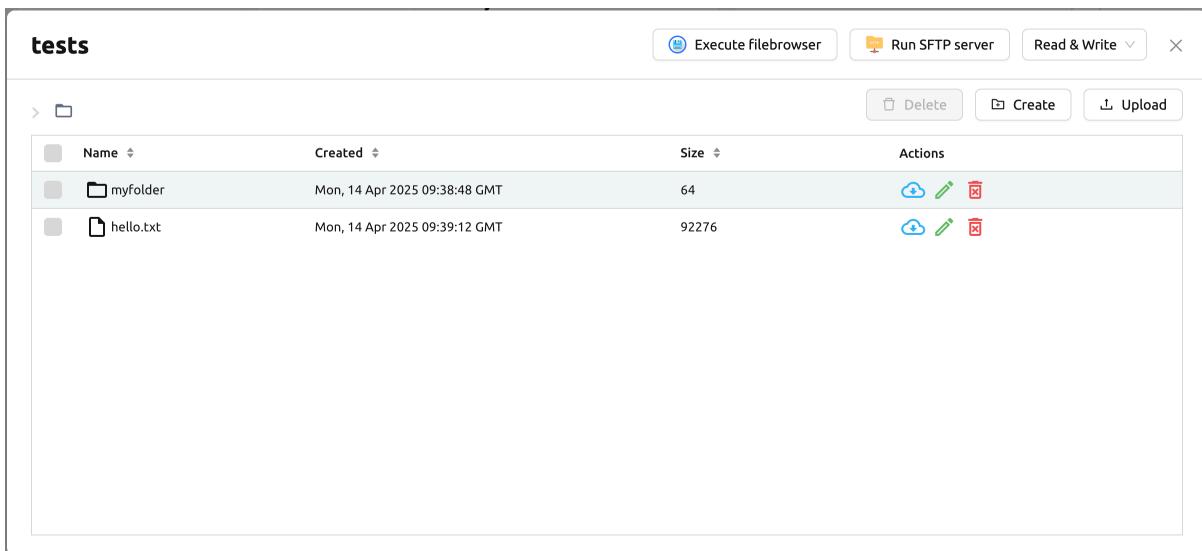
Storage 폴더의 내용을 다른 사용자 또는 프로젝트 멤버와 공유하여 공동 작업이나 학습을 수행할 필요가 있을 수 있습니다. 이를 위해 Backend.AI에서는 스토리지 폴더를 손쉽게 공유할 수 있습니다.

11.1 개인 스토리지 폴더를 다른 사용자와 공유하기

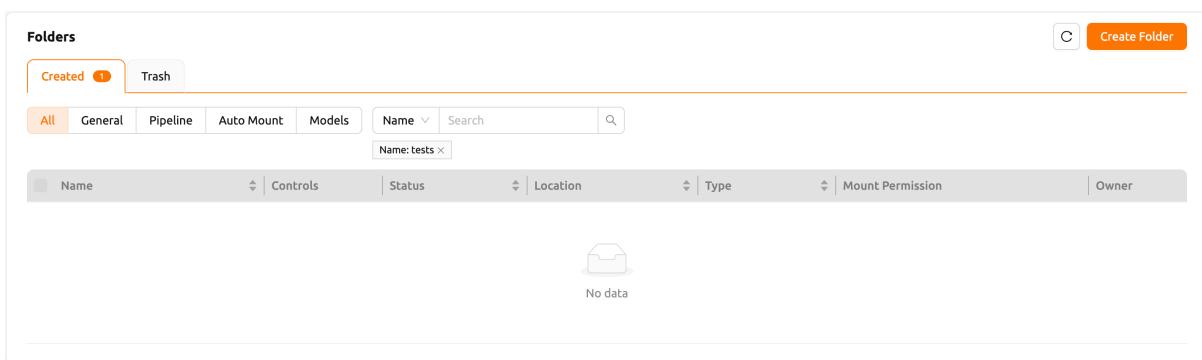
개인의 스토리지 폴더를 다른 사용자와 공유하는 방법에 대해 알아보겠습니다. 먼저 사용자 A의 계정으로 로그인한 뒤 데이터 페이지로 이동합니다. 여러 폴더 중 `tests` 폴더를 사용자B에게 공유하려고 합니다.

Name	Controls	Status	Location	Type	Mount Permission	Owner
tests			READY	localvolume1	R W	User

`tests` 폴더 안에는 `hello.txt`, `myfolder` 와 같은 파일 및 디렉토리가 있는 것을 볼 수 있습니다.

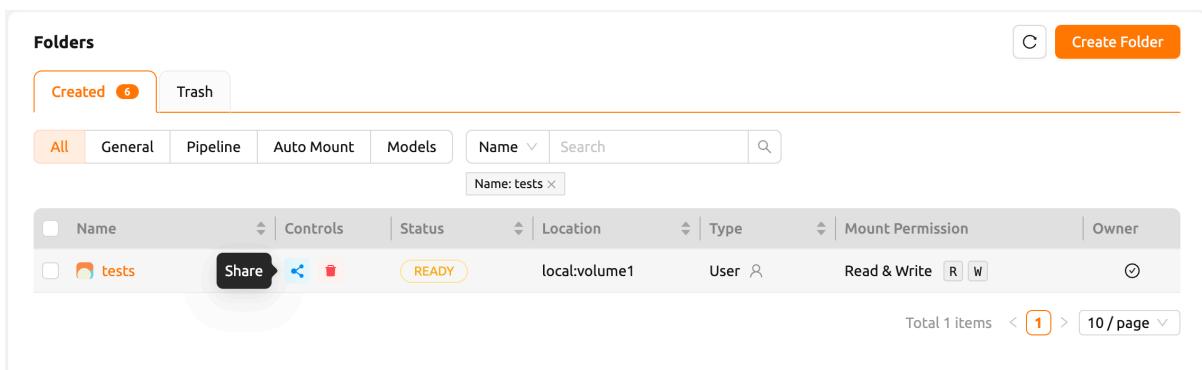


사용자B의 계정으로 로그인 했을 때 `tests` 폴더가 조회되지 않는 것을 확인합니다.



경고: 만약 사용자B 계정에 `tests`라는 이름의 폴더가 이미 존재한다면 사용자A의 `tests` 폴더는 사용자B에게 공유할 수 없습니다.

사용자A의 계정으로 전환한 후, 리스트에 있는 `tests` 폴더의 제어 열에서 공유 버튼을 클릭합니다.



공유 모달의 '사용자 초대' 항목에 사용자B의 계정 이메일을 입력하고, 원하는 공유 권한을 설정합니다. '읽기 전용' 권한을 선택하면 사용자B는 조회만 가능하며, '읽기 및 쓰기' 권한을 선택하면 조회뿐만 아니라 쓰기 작업도 수행할 수 있습니다.

The screenshot shows the 'Modify Permissions' dialog. At the top left is the title 'Modify Permissions' and a close button 'X'. Below it is the 'Invite User' section with fields for 'Email' (userB@lblup.com) and 'Permission' (Read only), followed by an 'Add' button. The main area is titled 'Shared Users' with a question mark icon. It contains a table with columns 'Invitee email' and 'Permission'. The table shows a single row with a placeholder icon and the message 'No data'.

다시 사용자B의 계정으로 전환한 뒤, 요약 페이지에 접속합니다. 요약 페이지의 초대 영역에서 폴더 초대가 도착한 것을 확인할 수 있으며, '수락' 버튼을 클릭해 초대를 수락합니다.

The screenshot shows the 'Invitation' dialog. At the top left is the title 'Invitation'. Below it is a message from 'userA@lblup.com' with details: 'Folder Name: tests' and 'Permission: R O'. At the bottom are two buttons: 'DECLINE' and 'ACCEPT', with 'ACCEPT' being orange.

데이터 페이지로 이동한 뒤 tests 폴더가 리스트에 출력되는 것을 확인하세요. 만약 리스트에 보이지 않는다면 브라우저 페이지 갱신을 해보시기 바랍니다. 초대를 수락했기 때문에 이제 사용자B

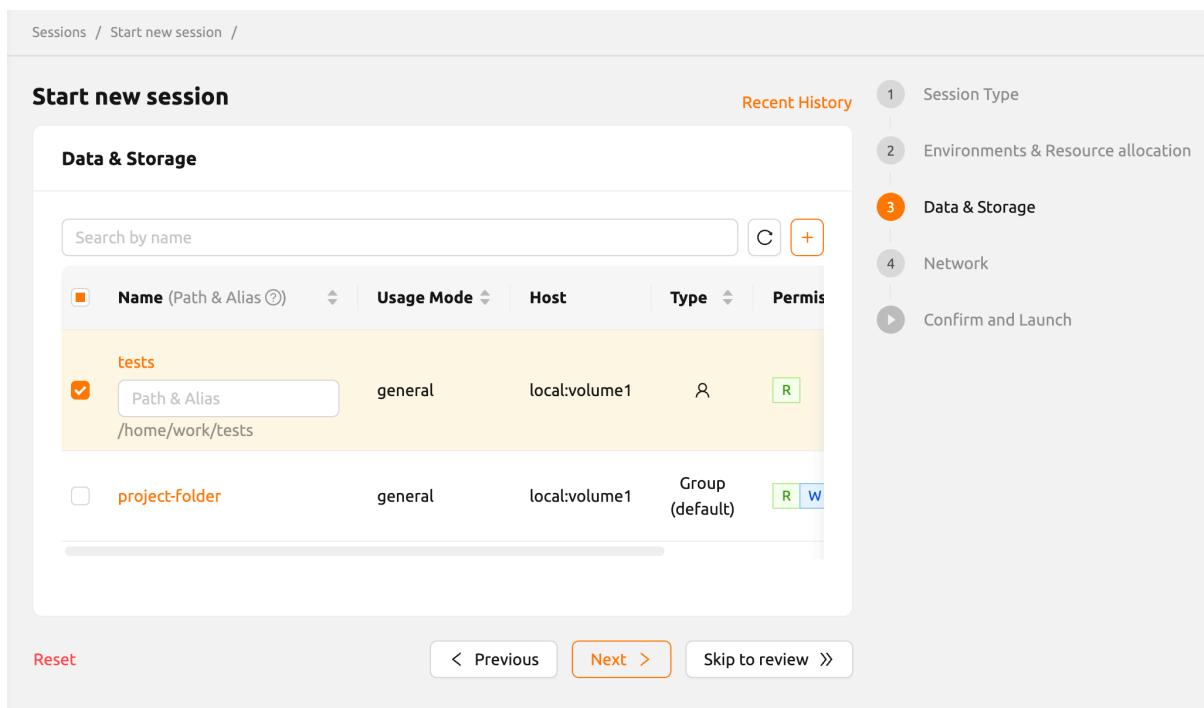
계정에서 사용자A의 test 폴더 내용을 조회할 수 있습니다. 사용자B가 생성한 다른 폴더와 비교해 tests 폴더는 제어 열에 관리 아이콘이 빠진 것을 확인할 수 있습니다. 마운트 권한 열에는 읽기 전용 표시만 띠 있는 것도 볼 수 있습니다.

Name	Controls	Status	Location	Type	Mount Permission	Owner
tests		READY	local:volume1	User	Read only R	
project-folder		READY	local:volume1	Project	Read & Write R W	

tests 의 Control 패널에서 폴더 아이콘을 클릭하여 tests 폴더 내부를 탐색 해 보겠습니다. 사용자 A의 계정에서 확인한 hello.txt 와 myfolder 를 다시 확인할 수 있습니다.

Name	Created	Size	Actions
myfolder	Mon, 14 Apr 2025 09:38:48 GMT	64	
hello.txt	Mon, 14 Apr 2025 09:39:12 GMT	92276	

이 스토리지 폴더를 사용자 B의 계정으로 마운트하여 컴퓨팅 세션을 생성 해 보겠습니다.



참고: Backend.AI 24.09 버전부터 세션 런처(NEO)의 개선된 버전을 기본 세션 런처로 제공합니다. 이전의 대화식 세션 런처를 사용하고 싶다면, [사용자 설정](#)을 참고하세요. Classic 세션 런처의 사용 방법에 대한 자세한 내용은 다음 링크를 참고하세요.

세션 생성 후 웹 터미널을 띄워 `tests` 폴더가 홈폴더에 마운트 된 것을 확인합니다. `tests` 폴더 내용은 조회가 되지만, 파일을 생성하거나 삭제하려는 시도는 허용되지 않습니다. 사용자A가 읽기 전용으로 공유했기 때문입니다. 만약 쓰기 권한까지 포함하여 공유했다면 사용자B도 `tests` 폴더에 파일을 생성할 수 있습니다.

```
work@main1[lgK4mq10-session]:~$ ls
DO_NOT_STORE_PERSISTENT_FILES_HERE.md  id_container  tests
work@main1[lgK4mq10-session]:~$ rm -r tests
rm: cannot remove 'tests': Read-only file system
work@main1[lgK4mq10-session]:~$
```

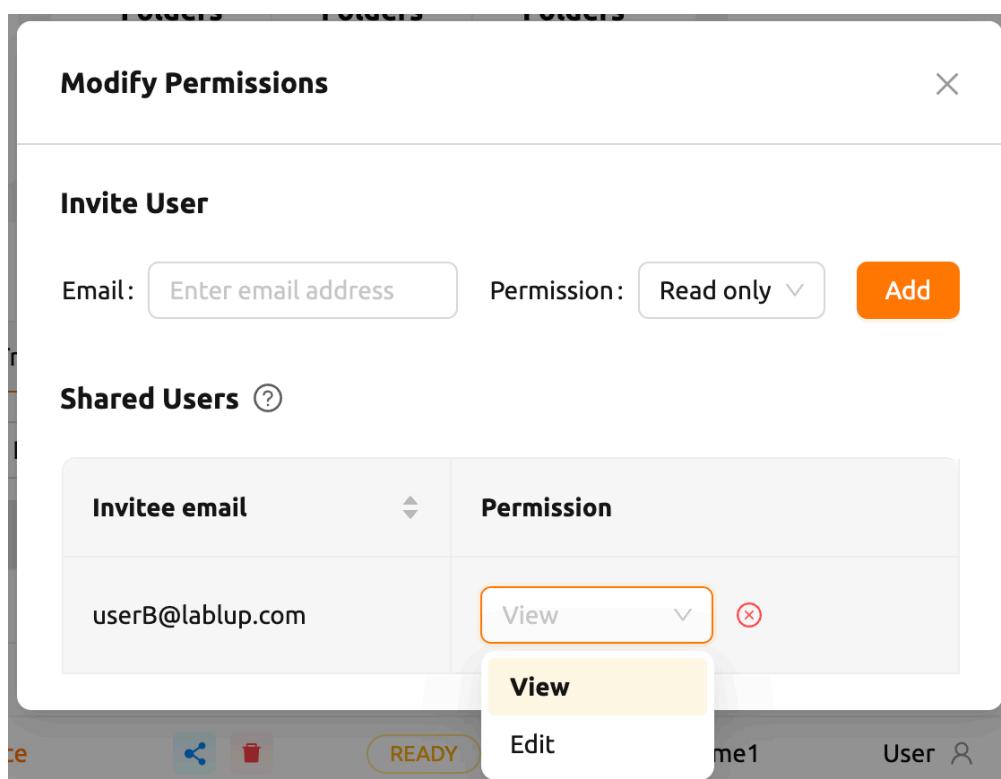
이렇게 개인 폴더를 Backend.AI 이메일 계정 기반으로 다른 사용자에게 공유할 수 있습니다.

참고: Backend.AI는 또한 프로젝트 구성원에게 공유 프로젝트 폴더를 제공합니다. 자세한 내용을 보려면 [프로젝트 구성원과 프로젝트 스토리지 폴더 공유](#)로 이동하십시오.

11.2 공유 폴더 권한 조정

마찬가지로, 폴더 공유 모달에서 공유된 사용자의 권한을 수정할 수 있습니다. 권한 선택란을 클릭해 공유 권한을 설정합니다. 권한 옆의 'x' 버튼을 클릭하면, 초대한 사용자를 제거할 수도 있습니다.

- 읽기 전용: 초대된 사용자는 폴더에 대한 읽기 전용 접근 권한을 갖습니다.
- 수정: 초대된 사용자에게 폴더에 대한 읽기 및 쓰기 권한이 있습니다. 사용자는 폴더나 파일을 삭제할 수 없습니다.



참고: 사용자가 수정 권한을 부여한 경우에도 폴더 자체 이름 변경은 소유자만 사용할 수 있습니다. 편집 권한은 폴더 이름 변경을 제공하지 않습니다.

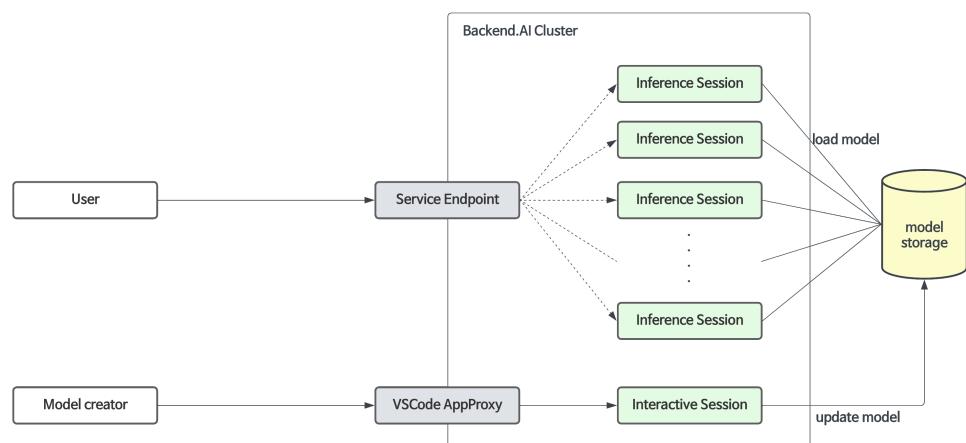
CHAPTER 12

모델 서빙

12.1 모델 서비스

참고: 이 기능은 엔터프라이즈 전용 기능입니다.

Backend.AI(백엔드에이아이)에서는 모델 학습 단계의 개발 환경 구축 및 자원 관리를 쉽게 해주는 것뿐만 아니라, 모델을 완성한 후 추론 서비스로 배포하고자 할 때에 최종 사용자(예: AI 기반 모바일 앱 및 웹서비스 백엔드 등)가 추론 API를 호출할 수 있게 하는 모델 서비스* 기능을 23.09 버전부터 정식 지원합니다.



모델 서비스는 기존의 학습용 연산 세션 기능을 확장하여, 자동화된 유지·보수 및 스케일링을 가능하게 하고 프로덕션 서비스를 위한 영구적인 포트 및 엔드포인트 주소 매핑을 제공합니다. 개발자나 관리자가 연산 세션을 수동으로 생성·삭제할 필요 없이 모델 서비스에 필요한 스케일링 파라미터를 지정해주기만 하면 됩니다.

12.2 23.03 및 이전 버전에서 모델 서비스를 구성하는 방법과 한계

모델 서빙에 특화된 모델 서비스 기능은 23.09에서 정식 지원하지만, 그 이전 버전들에서도 제한적인 모델 서비스 기능을 활용할 수 있습니다.

예를 들어, 23.03 버전에서는 다음과 같은 방법으로 학습용 연산 세션을 변용하여 모델 서비스를 구성할 수 있습니다:

1. 세션 생성시 사전 개방 포트를 추가해서 모델 서빙을 위한 세션 내 실행중인 서버 포트와 매핑하도록 설정(사전 개방 포트 사용법에 대한 설명은 [세션 생성하기 전에 사전 개방 포트를 추가하는 방법](#)을 참고하십시오.)
2. 앱을 외부에 공개 체크하여 사전 개방된 포트에 매핑된 서비스를 외부로 공개하도록 설정(앱을 외부에 공개에 대한 자세한 설명은 [앱을 외부에 공개](#)를 참고하십시오.)

대신 23.03 버전에서는 다음과 같은 제한 사항들이 있습니다:

- 세션이 외부 요인(유휴 시간 만료, 시스템 오류 등)으로 종료될 경우 자동으로 복구되지 않습니다.
- 세션을 새로 실행할 때마다 앱 포트가 변경됩니다.
- 세션을 반복적으로 재실행할 경우, 유휴 포트가 고갈될 수 있습니다.

23.09의 정식 모델 서비스 기능은 위와 같은 제한 사항들을 해결합니다. 따라서 23.09 버전부터는 가급적 모델 서빙 메뉴를 통해 모델 서비스를 생성·관리하는 것이 좋습니다. 사전 개방 포트 기능을 활용하는 방법은 개발 및 테스트 과정에서만 사용하는 것을 권장합니다.

12.3 모델 서비스를 사용하기 위한 단계 안내

모델 서비스를 사용하기 위해서는 크게 아래와 같은 단계를 따라야 합니다:

1. 모델 정의 파일 생성하기
2. 모델 정의 파일을 모델 타입 폴더에 업로드하기
3. 모델 서비스 생성하기/유효성 검사하기

4. (모델 서비스가 공개되지 않은 경우) 토큰 발급하기
5. (엔드유저 전용) 모델 서비스에 대응하는 엔드포인트에 접속하여 서비스 확인하기
6. (필요한 경우) 모델 서비스 수정하기
7. (필요한 경우) 모델 서비스 종료하기

12.3.1 모델 정의 파일 생성하기

참고: 24.03 버전부터는 모델 정의 파일명을 지정할 수 있습니다. 단, 모델 정의 파일 경로를 입력하는 란에 아무것도 입력하지 않을 경우, 시스템에서는 모델 파일명을 `model-definition.yml` 또는 `model-definition.yaml`로 간주하게 됩니다.

모델 정의 파일은 Backend.AI 시스템이 추론용 세션을 자동으로 시작, 초기화하고 필요에 따라 스케일링할 때 필요한 설정 정보를 담고 있는 파일입니다. 이 파일을 추론 서비스 엔진을 담고 있는 컨테이너 이미지와는 독립적으로 모델 타입 폴더에 저장합니다. 이를 통해 모델을 실행하는 엔진이 다양한 모델을 필요에 따라 바꿔가며 서비스할 수 있도록 하며, 모델이 변경될 때마다 컨테이너 이미지를 새로 빌드 및 배포하지 않아도 되도록 해줍니다. 네트워크 스토리지에서 직접 모델 정의와 모델 데이터를 불러오므로, 자동 스케일링 시 배포 과정을 더 단순화 및 효율화할 수 있습니다.

모델 정의 파일은 다음과 같은 형식을 따릅니다:

```
models:
  - name: "simple-http-server"
    model_path: "/models"
    service:
      start_command:
        - python
        - -m
        - http.server
        - --directory
        - /home/work
        - "8000"
    port: 8000
    health_check:
      path: /
      max_retries: 5
```

모델 정의 파일에 대한 키-값 설명

참고: "(필수)" 항목이 기재되어 있지 않은 항목은 선택 입력입니다.

- name (필수): 모델 명을 정의합니다.
- model_path (필수): 모델이 정의된 경로를 지정합니다.
- service: 서비스할 파일(명령어 스크립트, 코드 포함)에 대한 정보를 정리해두는 항목입니다.
 - pre_start_actions: start_command 항목 내 명령어 이전에 실행되어야 할 명령어 또는 액션을 정리해두는 항목입니다.
 - * action: 서비스 액션 안내 참고
 - * args/*: 서비스 액션 안내 참고
 - start_command (필수): 모델 서빙시 실행할 명령어를 지정합니다.
 - port (필수): 모델 서빙시 실행할 명령어에 대응해 컨테이너 기준 열어둘 포트를 지정합니다.
 - health_check: 서비스가 지정된 기간마다 에러 없이 실행되고 있는지 확인하는 항목입니다.
 - * path: 모델 서빙시 서비스가 제대로 실행되고 있는지를 확인하는 경로를 지정합니다.
 - * max_retries: 모델 서빙시 서비스에 요청이 간 뒤, 응답이 오지 않았을 때 재시도를 몇 번 할 것인지 지정합니다.

Backend.AI 모델 서빙에서 지원하는 서비스 액션 안내

- write_file: 입력받은 파일 명으로 파일을 생성, 내용을 추가하는 액션입니다. mode 값에 아무것도 적지 않을 경우 기본 접근 권한은 644 입니다.
 - arg/filename: 파일명을 적습니다.
 - body: 파일에 추가할 내용을 적습니다.
 - mode: 파일의 접근권한을 적습니다.
 - append: 파일에 내용 덮어쓰기/덧붙이기 설정을 True / False 로 적습니다.
- write_tempfile: 임시파일명(확장자는 .py)을 갖는 파일을 생성, 내용을 추가하는 액션입니다. mode 값에 아무것도 적지 않을 경우 기본 접근 권한은 644 입니다.
 - body: 파일에 추가할 내용을 적습니다.

- mode: 파일의 접근권한을 적습니다.
- run_command: 명령어를 실행한 결과(오류포함)를 아래와 같은 형태로 반환하게 됩니다. (out: 명령어 실행 결과, err: 명령어 실행 중 오류 발생시 출력되는 오류 메시지)
 - args/command: 실행할 명령어를 배열형태로 적습니다. (e.g. python3 -m http.server 8080 명령어는 ["python3", "-m", "http.server", "8080"])
- mkdir: 입력한 경로에 따라 디렉토리를 생성하는 액션입니다.
 - args/path: 디렉토리를 만들 경로를 지정합니다.
- log: 입력한 메시지에 따라 로그를 출력하는 액션입니다.
 - args/message: 로그에 표시할 메시지를 적습니다.
 - debug: 디버그 모드인 경우 True, 아닌 경우 False로 적습니다.

12.3.2 모델 정의 파일을 모델 타입 폴더에 업로드하기

모델 정의 파일(model-definition.yaml)을 모델 타입 폴더에 업로드하기 위해서는 가상 폴더를 생성해야 합니다. 이 때, 가상폴더 생성시 타입은 일반 이 아닌 모델 타입으로 선택하여 생성합니다. 생성하는 방법은 데이터 페이지의 [Storage 폴더 생성](#) 부분을 참고하십시오.

새 폴더 추가

폴더 이름 *

위치

사용 방식

 General Model

종류

 User Project

권한

 Read & Write Read Only

복사 가능 여부

초기화

취소

생성

폴더를 생성한 뒤, 데이터 페이지에서 '모델' 탭을 선택, 방금 생성한 모델 탑아 폴더 아이콘을 클릭하여 폴더 탐색기를 엽니다. 이후 모델 정의 파일을 업로드 합니다. 폴더 탐색기에 대한 자세한 사용 방법은 [폴더 탐색기](#) 를 확인하세요.

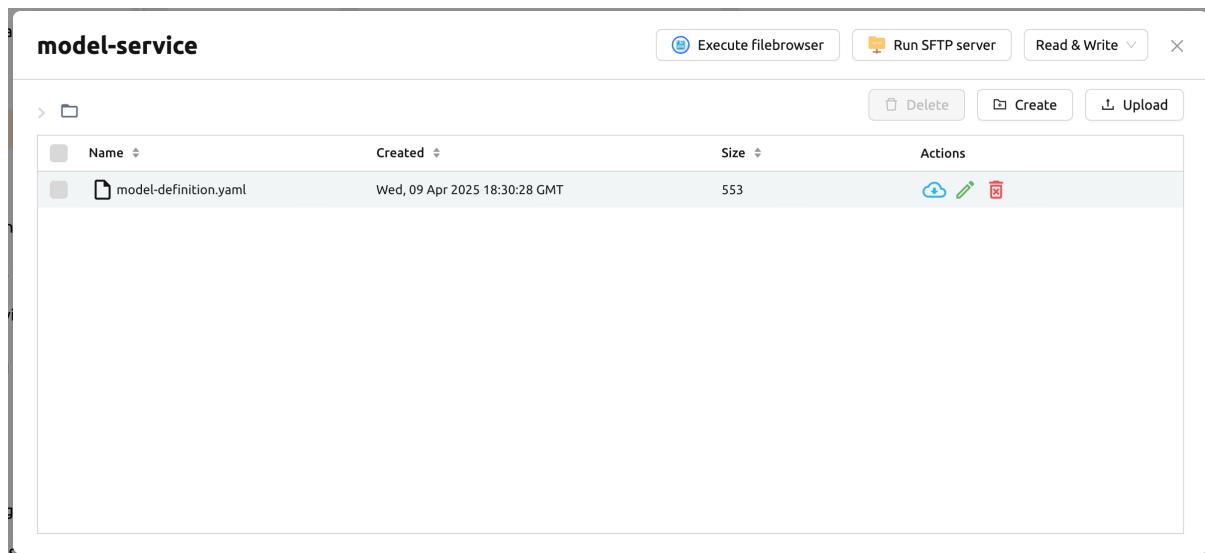
Folders

Created 4 Trash

Create Folder

Name	Status	Location	Type	Mount Permission	Owner
model-service	READY	local:volume1	User	Read & Write	

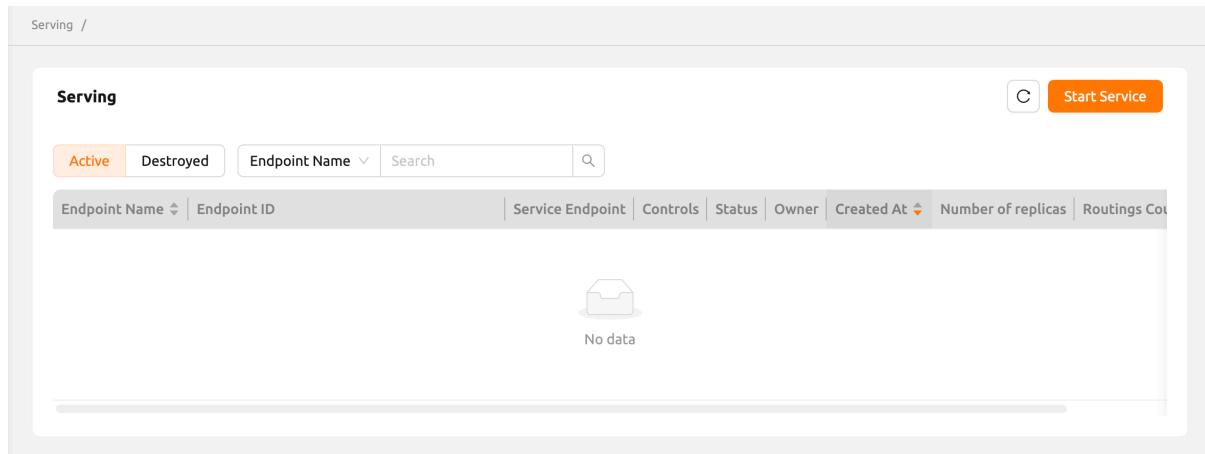
Total 1 items < 1 > 10 / page



12.3.3 모델 서비스 생성하기/유효성 검사하기

모델 정의 파일까지 모델 탑입의 가상 폴더에 모두 업로드하였다면, 본격적으로 모델 서비스를 생성할 준비가 된 것입니다.

모델 서빙 페이지에서 '서비스 시작' 버튼을 클릭하면, 서비스 생성에 필요한 설정을 입력하는 페이지로 이동합니다.



먼저, 서비스 이름을 입력합니다. 각 항목에 대한 자세한 설명은 아래 내용을 참고 하십시오.

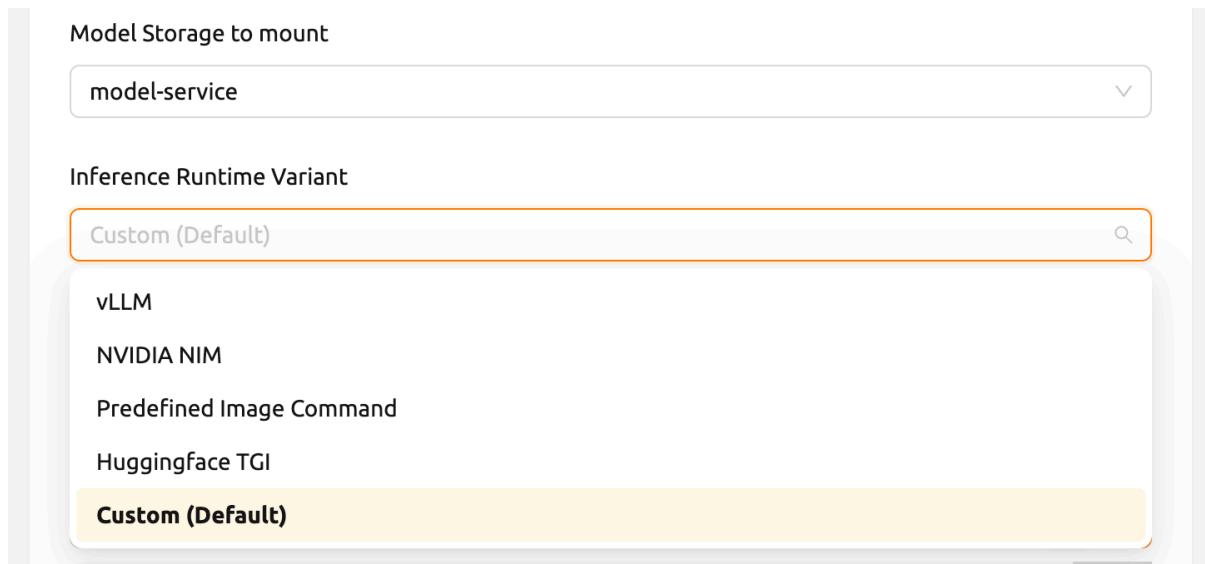
- 앱을 외부에 공개: 모델 서비스 생성 후 서비스하고자 하는 서버에 별도의 토큰이 없이도 접근할 수 있도록 하는 옵션. 기본적으로 비활성화 되어있음.
- 마운트할 모델 스토리지 폴더: 모델 정의 파일이 들어있는 마운트 할 모델 폴더를 선택.
- 인퍼런스 런타임 종류: 모델 서비스 탑입을 네가지로 나눔: 'vLLM', 'NVIDIA NIM', '미리 정의된 이미지 명령어', '커스텀'.

The screenshot shows the 'Start New Service' configuration interface. At the top, there are navigation links: 'Serving / Start New Service /'. The main form includes the following fields:

- Service Name:** A text input field.
- Open To Public (optional):** A toggle switch that is currently off.
- Model Storage to mount:** A dropdown menu set to 'model-service'.
- Inference Runtime Variant:** A dropdown menu set to 'Custom (Default)'.
- Mount Destination For Model Folder (optional):** A text input field containing '/models'.
- Model Definition File Path (optional):** A text input field containing 'model-definition.yaml'.
- Additional Mounts (optional):** A table listing three existing mounts:

Name	Usage Mode	Host	Type	Permission	Created
project-folder	general	local:volume1	Group (default)	R W	04/08/2024
test-folder2	general	local:volume1	X	R W	04/08/2024
test-folder1	general	local:volume1	X	R W	04/08/2024

예를 들어 'vLLM'이나 'NVIDIA NIM', 또는 '미리 정의된 이미지 명령어'를 모델 서비스의 런타임 배리언트로 선택한다면, 'model-definition' 파일을 마운트할 모델 폴더에 업로드할 필요가 없습니다. 그 대신, 별도의 환경변수를 설정해야 할 수도 있습니다. 더 자세한 정보를 확인하려면, [모델 배리언트: 쉽게 서빙하는 다양한 모델 서비스 포스팅](#)을 확인하시기 바랍니다.



- 모델 폴더 마운트할 경로: 서비스의 라우팅에 대응하는 세션에 마우트 될 모델 스토리지 경로에 별칭 추가. 기본값은 `/models`.
- 모델 정의 파일 경로: 모델 스토리지 경로에 있는 모델 정의 파일을 정하는 옵션으로, 기본값은 `model-definition.yaml`.
- 추가 마운트: 세션과 마찬가지로, 서비스에 폴더를 추가로 마운트할 수 있음. 단, 모델 폴더는 추가 마운트가 불가능하고, 일반/데이터 사용 모드의 폴더만 마운트할 수 있음.

The screenshot shows the 'Additional Mounts' section. It includes fields for 'Mount Destination For Model Folder (optional)' (set to '/models') and 'Model Definition File Path (optional)' (set to 'model-definition.yaml'). Below these are two input fields: 'Search by name' and a 'Create' button. A table lists three existing mounts:

Name (Path & Alias)	Usage Mode	Host	Type	Permission	Create
project-folder	general	local:volume1	Group (default)	R W	04/08,
test-folder2	general	local:volume1	Group (default)	R W	04/08,
test-folder1	general	local:volume1	Group (default)	R W	04/08,

그리고, 복제본 수와 이미지 환경, 자원 그룹을 선택합니다. 자원 그룹은 모델 서비스에 할당될 수 있는 자원의 집합입니다.

- 복제본 수: 해당 설정은 현재 서비스에 대해 유지할 라우팅 세션 수를 결정하는 기준이 됨.

만약 해당 설정값을 변경하면, 매니저가 기존에 실행되고 있는 복제본의 개수를 참조하여 새로운 복제본을 생성하거나, 실행중인 세션을 종료할 수 있음.

- 실행 환경 / 버전: 모델 서비스에서 서비스 전용 서버의 실행 환경을 설정, 현재는 서비스 내 라우팅이 여러 개여도 단일 실행 환경으로만 실행되도록 지원하고 있음. (추후 업데이트 예정)

The screenshot shows the 'Create Model Service' form. It includes fields for 'Number of replicas' (set to 1), 'Environments / Version' (selected as '</> python-tcp-app'), 'Image Name (Manual) (optional)' (empty), and 'Resource Group' (selected as 'default').

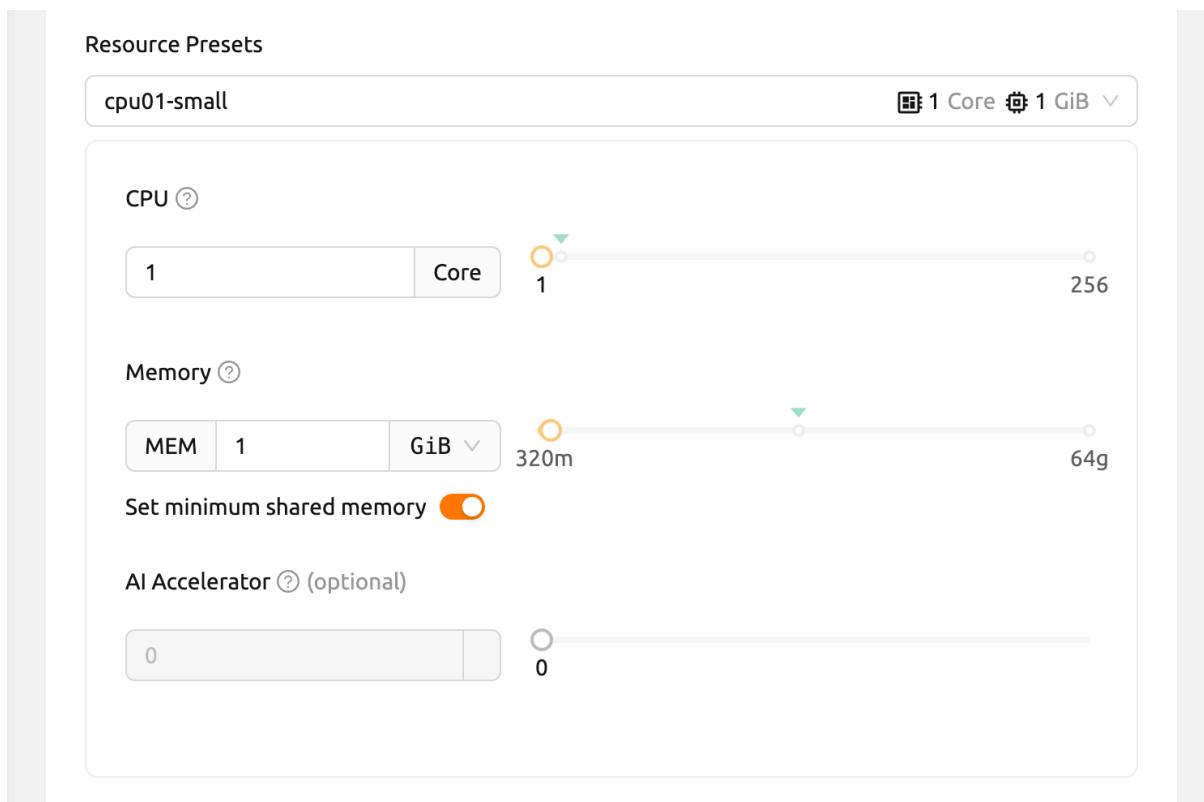
Number of replicas: 1

Environments / Version: </> python-tcp-app

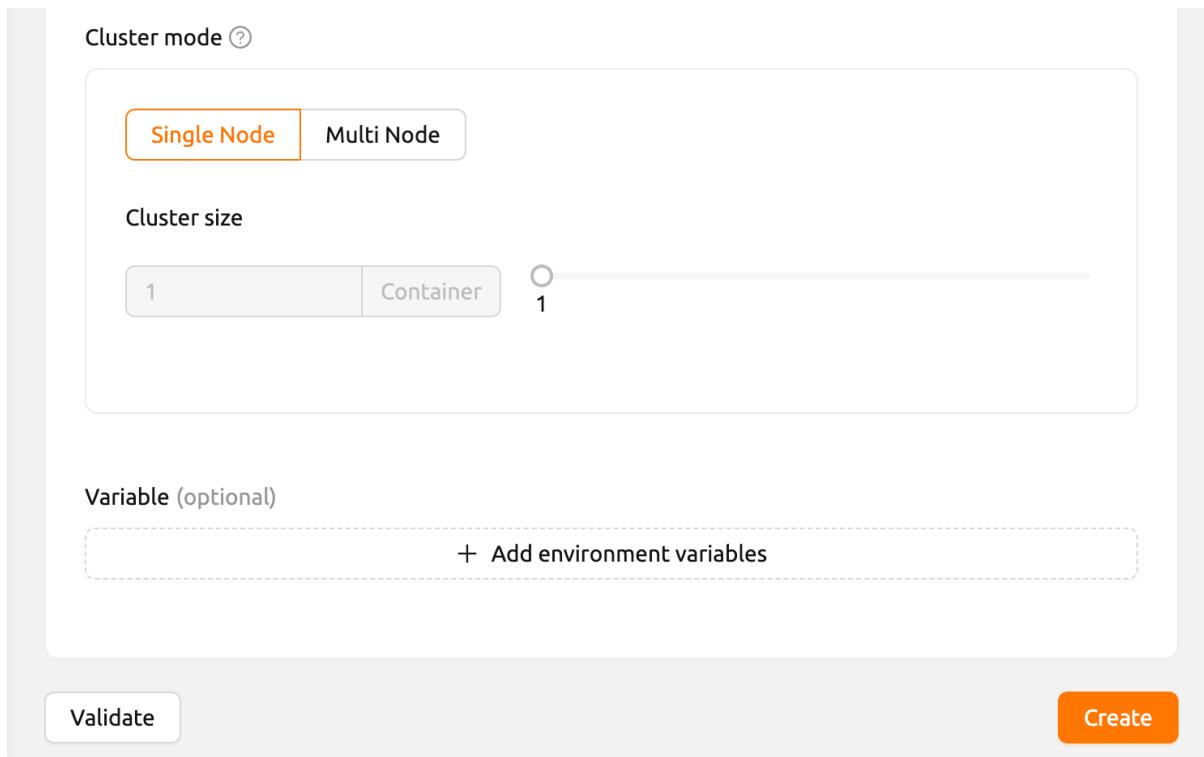
Image Name (Manual) (optional):

Resource Group: default

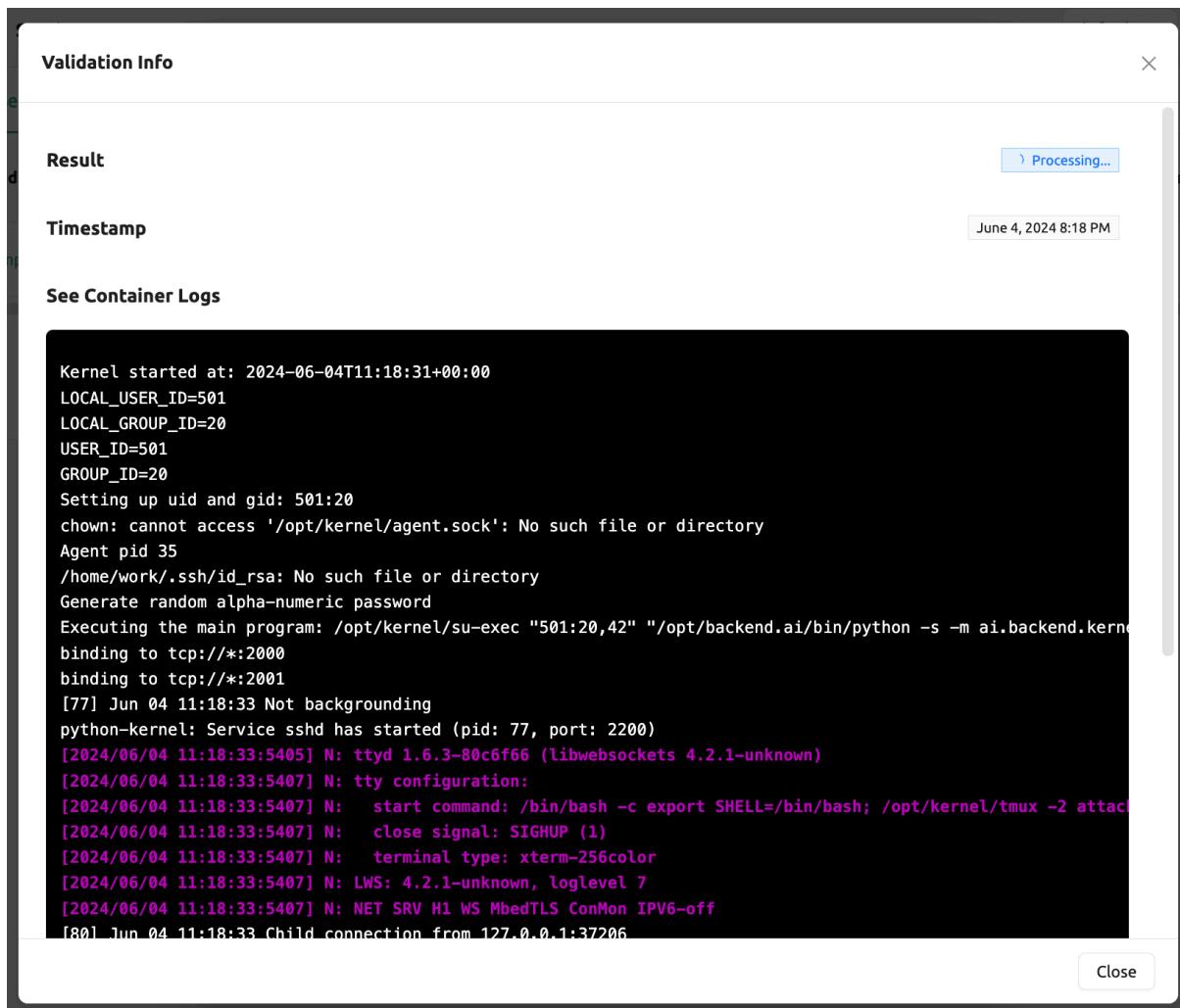
- 자원 프리셋: 모델 서비스에서 할당하고자 하는 자원량을 선택할 수 있음. 자원에는 CPU, RAM, 그리고 GPU로 알려져있는 AI 가속기가 해당.



- 단일 노드: 세션을 실행할 때에 관리 노드와 워커 노드들이 하나의 물리 노드 또는 가상 머신에 배치되는 경우.
- 다중 노드: 세션을 실행할 때 하나의 관리 노드와 하나 이상의 워커 노드가 여러 물리 노드 또는 가상 머신에 나누어 배치되는 경우.
- 환경 변수: 이 섹션에서는, 모델 서비스를 시작할 때 설정되는 환경 변수를 설정할 수 있음. 런타임 배리언트를 사용해서 모델 서비스를 생성할 때 유용함. 몇몇 런타임 배리언트는 시작 전 환경 변수 설정이 필요함.



모델 서비스를 생성하기 전, Backend.AI에서는 성공적으로 실행이 가능한지 아닌지(실행중 발생하는 어떤 에러로 인해 실행이 불가능한 경우)를 체크하는 유효성 검사를 지원합니다. 서비스 런쳐의 좌측 하단에 위치한 Validate 버튼을 클릭하면 유효성 검사 이벤트를 확인하는 팝업창이 새로 뜨게 됩니다. 이 팝업 창에서, 여러분은 컨테이너 로그로 상태를 확인할 수 있습니다. 결과 값이 Finished로 뜨게 되면, 유효성 검사가 끝난 것을 의미합니다.



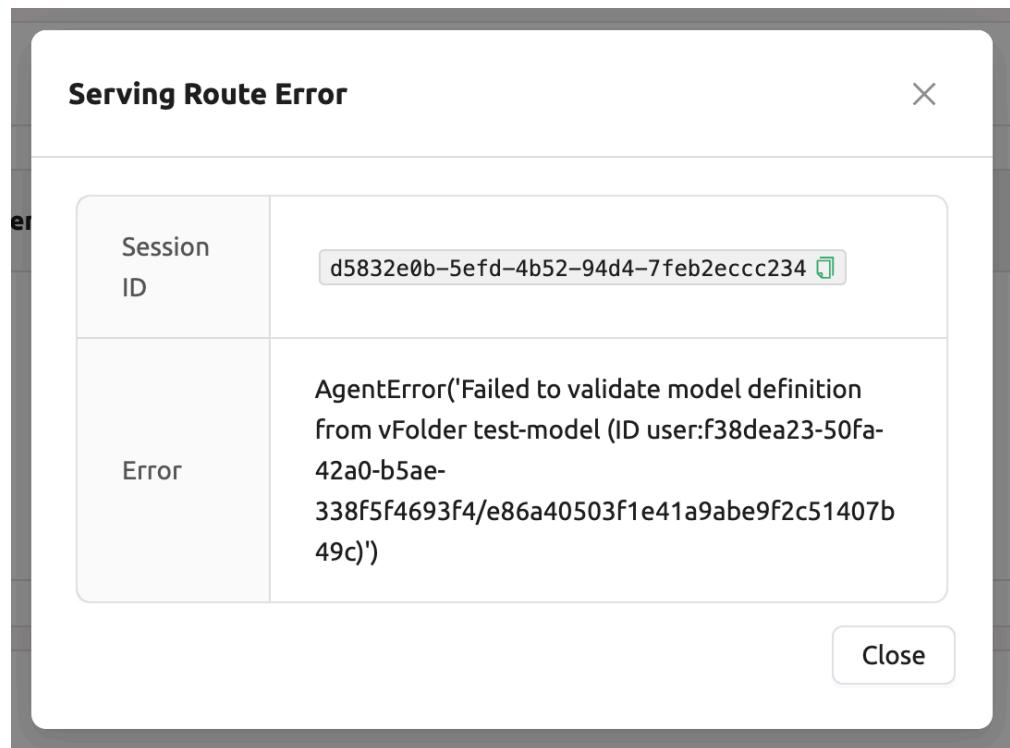
참고: Finished 결과 값이 실행이 성공적으로 종료되었음을 의미하지는 않습니다. 그 대신, 컨테이너 로그를 반드시 확인하시기 바랍니다.

모델 서비스 생성에 실패한 경우

만일 모델 서비스의 상태가 UNHEALTHY로 되어 있는 경우, 모델 서비스를 정상적으로 실행할 수 없는 상태라고 볼 수 있습니다.

생성이 안 되는 이유 및 해결법은 대개 다음과 같습니다:

- 모델 서비스 생성 시 너무 적은 양의 자원을 라우팅에 할당한 경우
 - 해결법: 해당 서비스를 우선 종료하고, 이전 설정보다 많은, 충분한 양의 자원을 할당하도록 설정하여 서비스를 재생성합니다.
- 모델 정의 파일(model-definition.yml)의 형식이 잘못된 경우



- 해결법: 모델 정의 파일 형식을 확인하고, 키-값이 잘못된 경우, 수정하여 저장된 경로에 업로드해 덮어쓰기합니다. 이후 아래와 같이 오류 지우고 재시도 버튼을 클릭하여 라우트 정보에 쌓인 에러를 모두 삭제하고, 재시작해 모델 서비스의 라우팅이 정상적으로 동작할 수 있도록 합니다.

Service Info

Endpoint Name	test	Status	UNHEALTHY	Endpoint ID	141ce475-9330-428a-b2db-b3c4fbdbf6c8
Session Owner	user@labelup.com	Number of replicas	1	Service Endpoint	No Service Endpoint
Open To Public	✓	Resources	default	8 Core	48 GiB
Model Storage	→ /models	Additional Mounts		Variable	-
Image	cr.backend.ai/testing/python:pytorch2.1-py310-cuda12.2@x86_64				

12.3.4 오토스케일링 규칙

실행 중인 서비스에 대해 오토스케일링 규칙을 설정할 수 있습니다. 설정된 규칙에 따라 서비스 사용량이 적을 때는 복제본 수를 줄여 자원을 절약할 수 있고, 사용량이 많을 때는 복제본 수를 늘려 요청 지연이나 실패를 방지할 수 있습니다.

The screenshot shows a table titled "Auto Scaling Rules" with the following columns: Type, Metric Source, Condition, Controls, Step Size, Min / MAX Replicas, CoolDown Seconds, Last Triggered, and Created At. There is one row in the table, which is currently empty. A small icon of a document and the text "No data" are displayed below the table.

'규칙 추가' 버튼을 클릭하면 규칙을 추가할 수 있는 모달이 나타나며, 각 항목의 의미는 다음과 같습니다:

- 규칙을 정의합니다. 규칙의 적용 범위에 맞게 스케일 업 또는 스케일 다운 중 하나를 선택합니다.
- 메트릭 소스: 메트릭이 수집될 소스를 선택합니다. Inference Framework 또는 Kernel 중에서 선택할 수 있습니다.
 - Inference Framework: 모든 복제본에 대해 계산된 평균값입니다. 모든 앱 프록시 인스턴스가 추론 메트릭을 보고하는 경우에만 지원됩니다.
 - Kernel: 해당 엔드포인트를 구성하는 모든 커널의 값에 대해 계산된 평균값입니다.
- 조건: 오토스케일링 규칙이 작동하는 조건을 설정합니다.
 - 메트릭: 비교에 사용될 메트릭의 이름입니다. 실행 환경에서 지원하는 메트릭이라면 자유롭게 입력할 수 있습니다.
 - 비교 방식: 실시간 메트릭 값이 임계값과 비교되는 기준을 정의합니다.
 - * LESS_THAN: 현재 메트릭 값이 설정된 기준값보다 낮아지면 해당 규칙이 실행됩니다.
 - * LESS_THAN_OR_EQUAL: 현재 메트릭 값이 설정된 기준값보다 낮아지거나 같을 경우 해당 규칙이 실행됩니다.
 - * GREATER_THAN: 현재 메트릭 값이 설정된 기준값을 초과하면 해당 규칙이 실행됩니다.
 - * GREATER_THAN_OR_EQUAL: 현재 메트릭 값이 설정된 기준값을 초과하거나 같을 경우 해당 규칙이 실행됩니다.

- 기준값: 스케일링 조건이 충족되는지를 판단하기 위한 기준값입니다.
- 단계 크기: 규칙이 실행될 때 변경될 복제본 수입니다. 양수 또는 음수로 설정할 수 있으며, 음수로 설정된 경우 복제본 수가 감소합니다.
- 최대/최소 복제본 수: 해당 엔드포인트의 복제본 수에 대한 최대값/최소값을 설정합니다. 계산된 복제본 수가 최대값을 초과하거나 최소값 미만이 되는 경우, 규칙은 실행되지 않습니다.
- 쿨다운 시간(초): 규칙이 한 번 실행된 후, 일정 시간(초) 동안 동일한 규칙이 다시 적용되지 않도록 하는 대기 시간입니다.

Add Auto Scaling Rule

Type

Scale Up Scale Down

Metric Source

Kernel

Condition ?

Metric Name	> ▾	Threshold	%
-------------	-----	-----------	---

Step Size

1

Max Replicas

5

CoolDown Seconds

300

Cancel OK

12.3.5 토큰 발급하기

모델 서비스를 성공적으로 실행한 경우, 상태는 HEALTHY에 속하게 됩니다. 이 경우 모델 서비스 탭에서 해당하는 엔드포인트 명을 클릭해 모델 서비스의 상세 정보를 확인할 수 있습니다. 이후 모델 서비스의 라우팅 정보에서 서비스 엔드포인트를 확인할 수 있는데, 이 엔드포인트는 서비스 생성시 외부에 공개할 수 있는 'Open to Public' 값이 활성화 된 경우, 엔드포인트가 공개되어 별도의 토큰이 없이도 최종 엔드유저가 엔드포인트에 접근할 수 있습니다. 하지만 비활성화 된 경우는 토큰 발급 후 토큰을 아래와 같은 형태로 추가해서 서비스가 정상적으로 실행되고 있는지 확인할 수 있습니다.

Generated Tokens				
Token	Status	Expired Date		Created At
				No data

라우팅 정보의 생성된 토큰 목록 우측에 있는 'Generate Token' 버튼을 클릭합니다. 이후 토큰 생성을 위한 모달이 뜨면, 만료일을 입력합니다.

Generate new token

The default token expiration time is 24 hours later.

* Expired Date: 2025-04-10 18:58:53

이후 발급되는 토큰은 생성된 토큰 목록에 추가됩니다. 토큰 항목의 '복사' 버튼을 클릭하여 토큰을 복사하고, 아래와 같은 키의 값으로 추가하면 됩니다.

Generated Tokens			
Token	Status	Expired Date	Created At
eyJhbGciOiJIUzI1N... ⓘ	Valid	Apr 10, 2025 7:07:51 PM	Apr 9, 2025 7:07 PM
eyJhbGciOiJIUzI1N... ⓘ	Expired	Mar 28, 2025 11:46:12 PM	Mar 27, 2025 11:46 PM
eyJhbGciOiJIUzI1N... ⓘ	Expired	Mar 28, 2025 11:45:57 PM	Mar 27, 2025 11:45 PM

키	값
Content-Type	application/json
Authorization	BackendAI

12.3.6 (엔드유저 전용) 모델 서비스에 대응하는 엔드포인트에 접속하여 서비스 확인하기

모델 서빙이 완료되려면 실제 최종 엔드 유저에게 모델 서비스가 실행되고 있는 서버에 접근할 수 있도록 정보를 공유하여야 합니다. 이 때 서비스 생성시 Open to Public 값이 활성화한 경우라면 라우팅 정보 페이지의 서비스 엔드포인트 값을 공유하면 됩니다. 만일 비활성화 한 채로 서비스를 생성한 경우라면 서비스 엔드포인트 값과 앞서 생성한 토큰 값을 공유하면 됩니다.

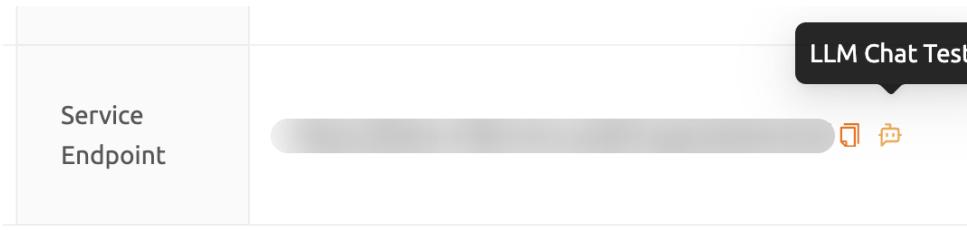
`curl` 명령어를 사용해서 모델 서빙 엔드포인트에 보내는 요청이 제대로 동작하는지 아닌지 확인할 수 있습니다.

```
$ export API_TOKEN=<token>
$ curl -H "Content-Type: application/json" -X GET \
$ -H "Authorization: BackendAI $API_TOKEN" \
$ <model-service-endpoint>
```

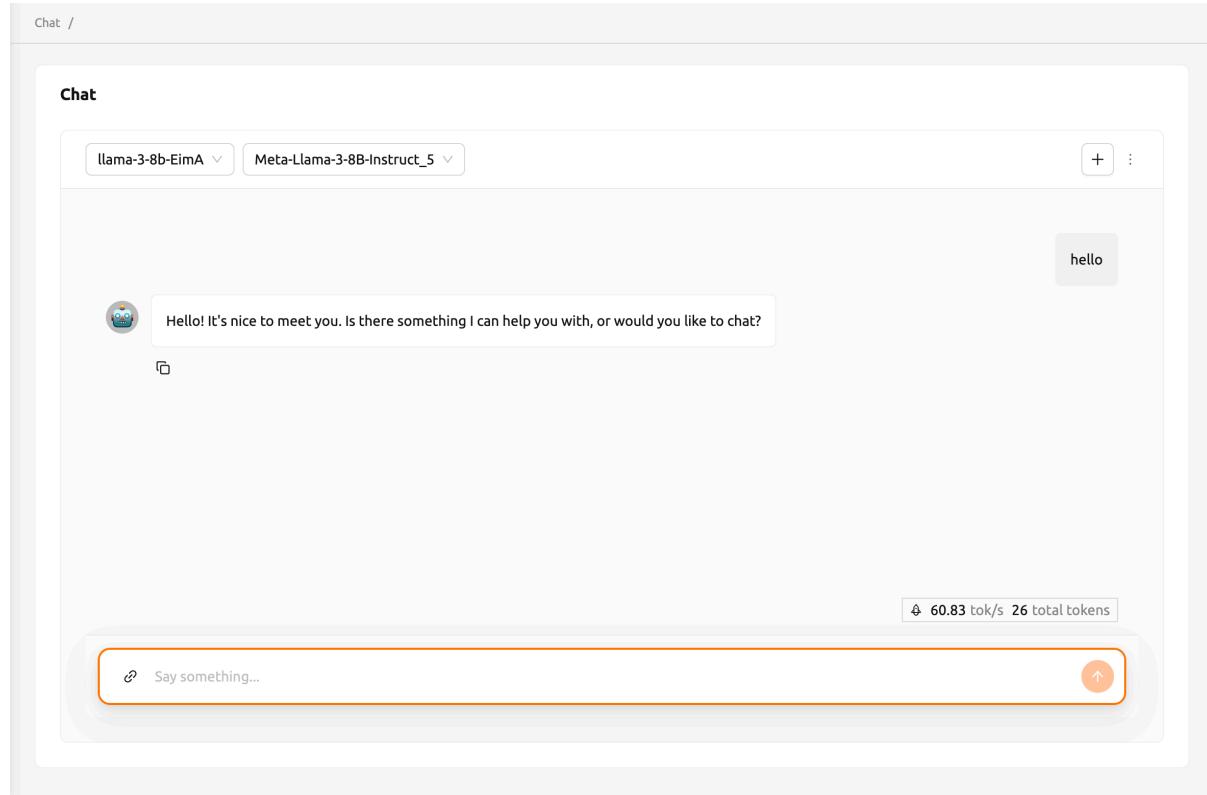
경고: 기본적으로 엔드 유저는 엔드포인트에 접근이 가능한 네트워크 망에 있어야 합니다. 만일 폐쇄망에서 서비스를 생성한 경우, 폐쇄망 내 접근이 가능한 엔드 유저만 접근이 가능합니다.

12.3.7 거대 언어 모델 (LLM) 사용하기

LLM(Large Language Model) 모델 서비스를 생성했다면, 실시간으로 LLM을 사용해 볼 수 있습니다. 서비스 엔드포인트 열에 있는 'LLM 채팅 테스트' 버튼을 클릭하여 테스트를 수행할 수 있습니다.



버튼을 클릭하면 채팅 페이지로 이동되며, 사용자가 생성한 모델이 자동으로 선택됩니다. 채팅 페이지에 제공된 채팅 인터페이스를 통해 LLM 모델을 테스트할 수 있습니다. 채팅 기능에 대한 자세한 내용은 [채팅 페이지](#)를 참고하세요.

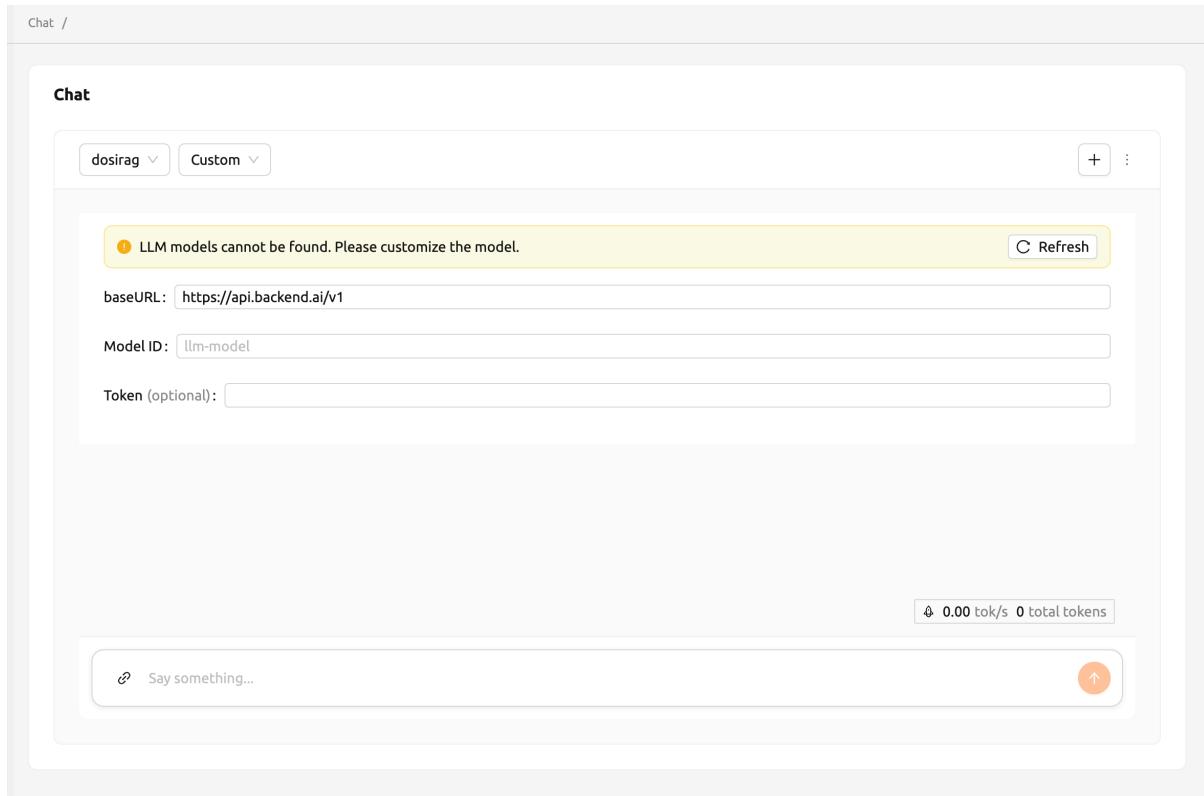


API에 연결하는 데 문제가 발생하는 경우, 채팅 페이지에서 모델 설정을 수동으로 구성할 수 있는 옵션이 표시됩니다. 모델을 사용하려면 다음 정보를 입력해야 합니다.

- baseURL: 모델이 위치한 서버의 기본 URL. 해당 URL에는 버전 정보가 포함되어 있어야 합니다. 예를 들어, OpenAI API를 사용하는 경우, <https://api.openai.com/v1> 을 입력해야 합니다.
- Model ID: 사용하려는 모델의 고유 식별자. 모델을 지정하려면 고유 식별자를 제공해야 합니다.

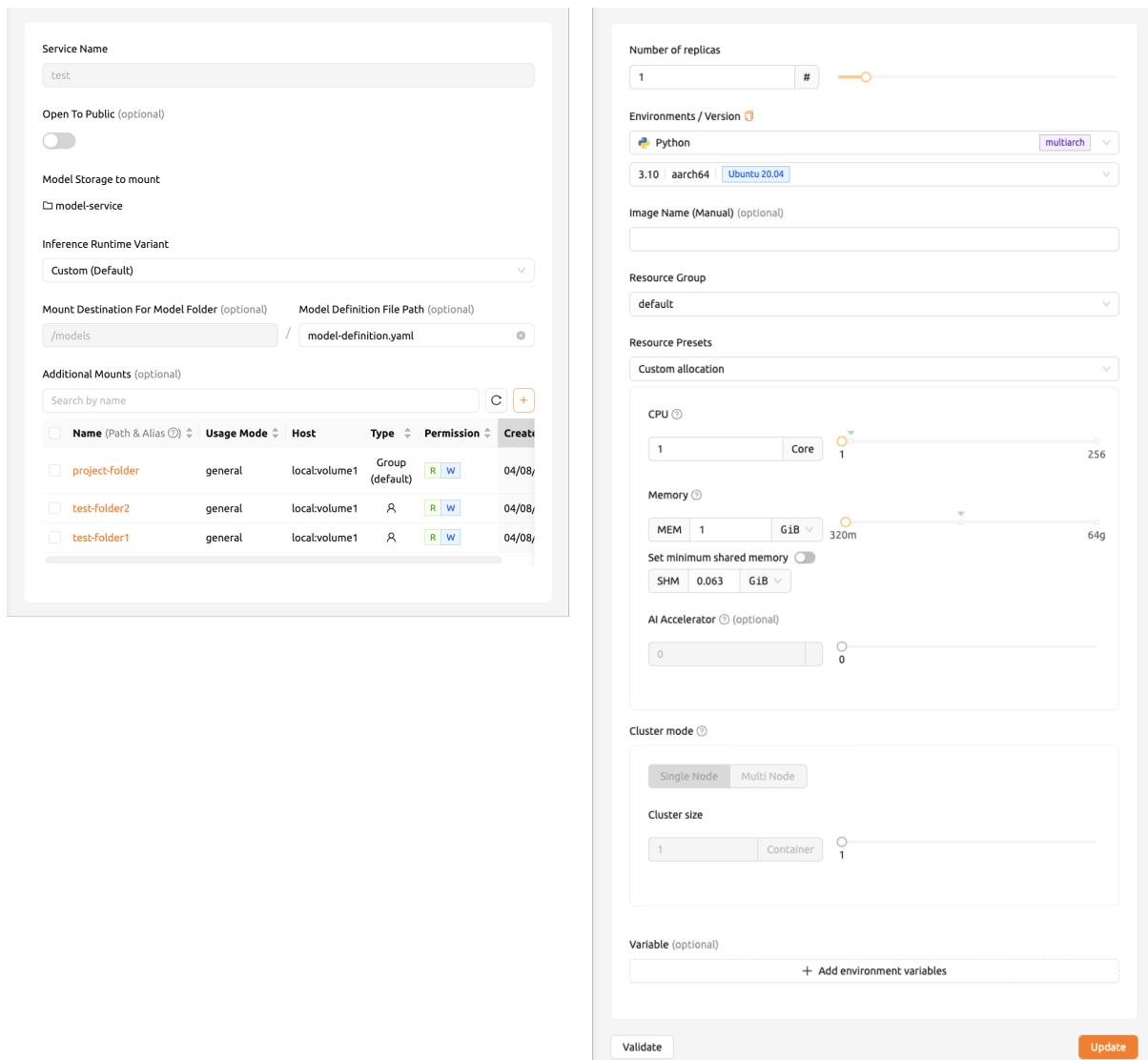
합니다. 예를 들어, GPT-4의 경우 gpt-4o를 입력합니다.

- Token (선택사항): 모델 서비스에 접근하기 위한 인증 키. 토큰은 Backend.AI 뿐만 아니라 다양한 서비스에서 생성할 수 있습니다. 토큰 생성 방법은 서비스에 따라 다를 수 있습니다. 자세한 내용은 해당 서비스의 안내를 참조하십시오. Backend.AI에서 생성한 서비스를 사용하는 경우, 토큰 생성 방법은 [토큰 발급하기](#) 섹션을 참조하십시오.



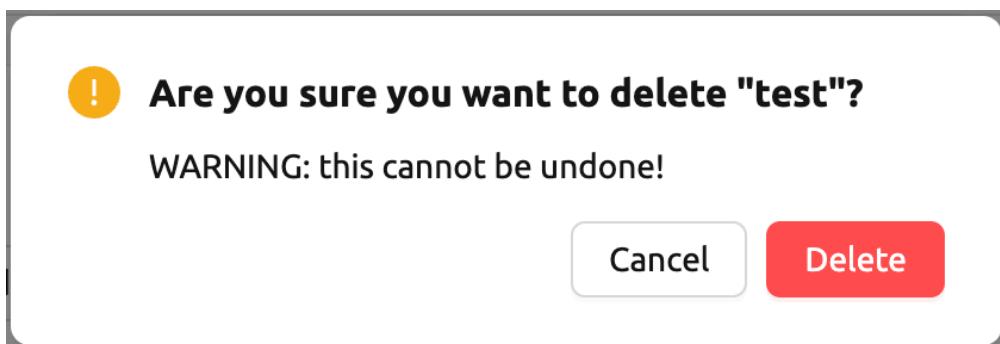
12.3.8 모델 서비스 수정하기

모델 서비스를 수정하기 위해선, 제어 탭의 설정 아이콘을 클릭합니다. 수정 모달은 모델 서비스를 시작하는 모달과 형태가 동일하며, 이전에 입력했던 필드들이 적용되어 있습니다. 원하는 필드를 수정하고 확인 버튼을 클릭하게 되면, 변경 사항이 적용됩니다.



12.3.9 모델 서비스 종료하기

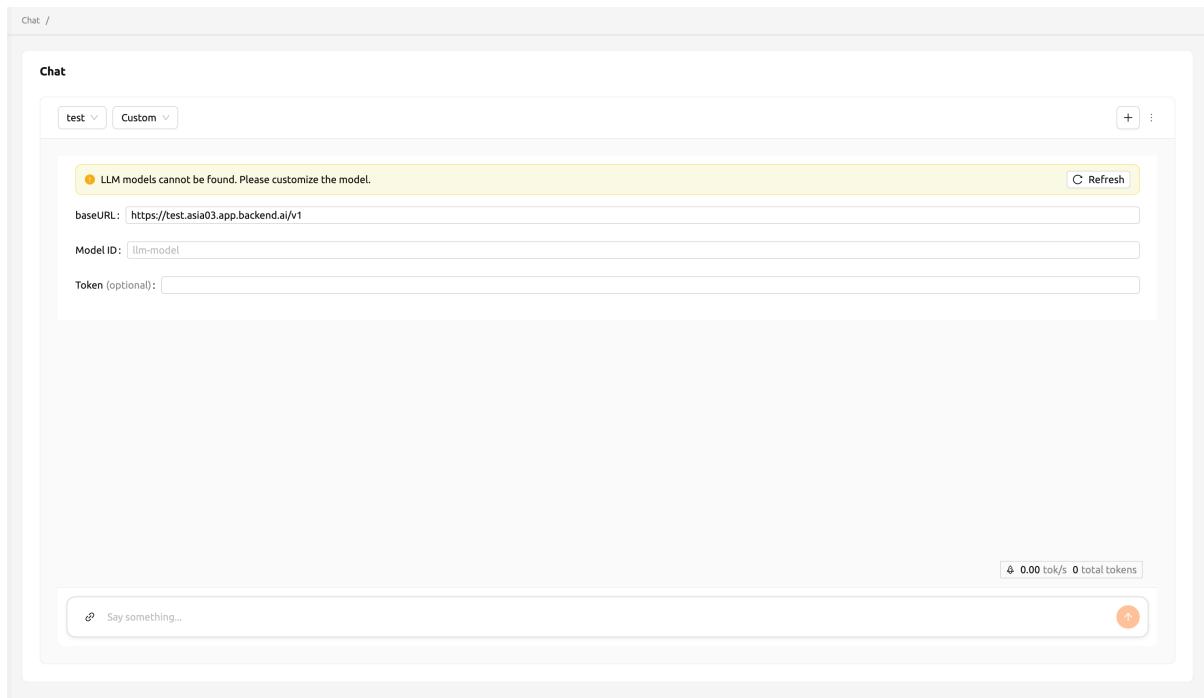
모델 서비스는 주기적으로 스케줄러를 실행하여, 원하는 세션 수와 실제 대응하는 라우팅 수가 원하는 세션 수에 맞춰지도록 스케줄링합니다. 다만, 이 경우 Backend.AI 스케줄러에 부하가 가는 것은 불가피 합니다. 따라서 모델 서비스를 더 이상 사용하지 않는 경우라면, 모델 서비스를 종료하는 것이 좋습니다. 모델 서비스를 종료하려면, 제어 탭에서 휴지통 아이콘을 클릭합니다. 이후 모델 서비스를 종료하는 것이 맞는지 확인하는 모달이 뜨게 됩니다. 삭제 버튼을 누를 경우 모델 서비스는 종료됩니다. 종료된 모델 서비스의 경우 모델 서비스 목록에서 제거됩니다.



CHAPTER 13

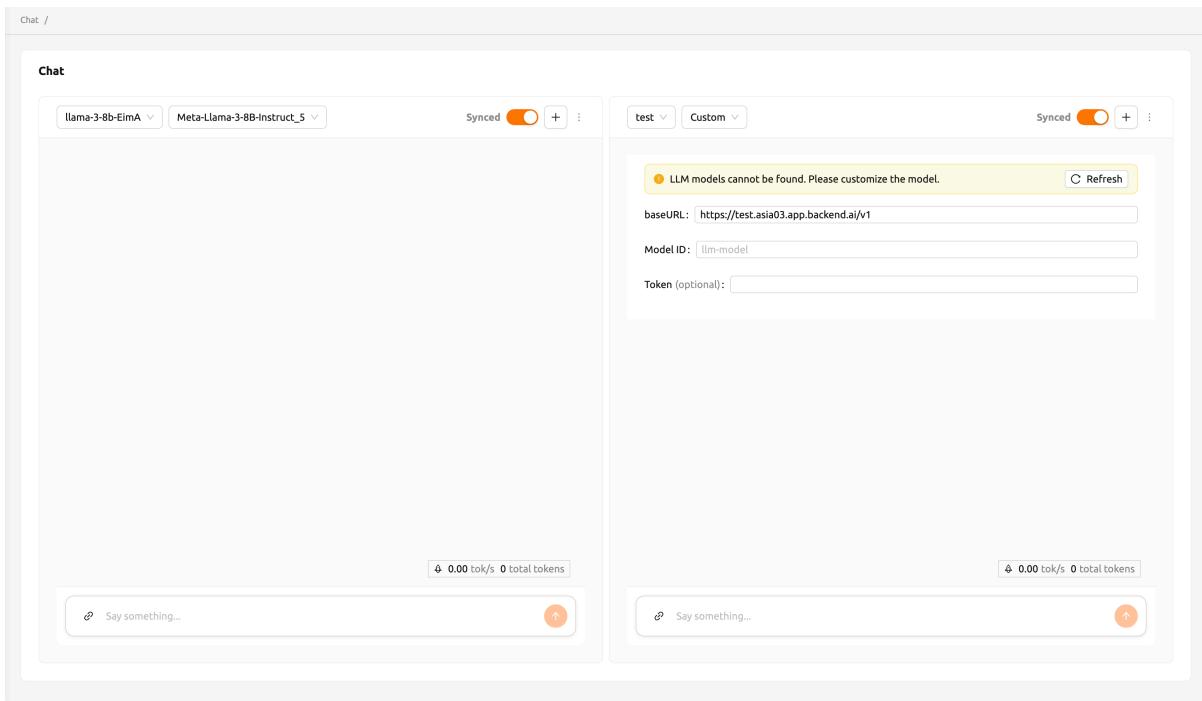
채팅 페이지

Backend.AI 25.05 버전 이후부터, 'LLM Playground' 기능이 별도의 '채팅' 페이지로 제공됩니다. 채팅 페이지를 통해 사용자는 Backend.AI가 제공하는 서비스뿐만 아니라 여러 LLM 모델을 직접 체험해 볼 수 있습니다.



13.1 모델 선택하기

채팅 페이지 카드의 왼쪽 상단에서 엔드포인트와 모델을 선택할 수 있습니다. 엔드포인트 선택란을 클릭하면, 엔드포인트를 직접 검색하거나 목록에서 선택할 수 있으며, 모델도 동일한 방식으로 선택할 수 있습니다. 단, 선택한 엔드포인트에 해당하는 모델이 없을 경우, 모델은 Custom으로 설정됩니다. 모델 정보를 불러올 수 없는 경우에는 사용자가 직접 모델을 수동으로 설정해야 합니다.

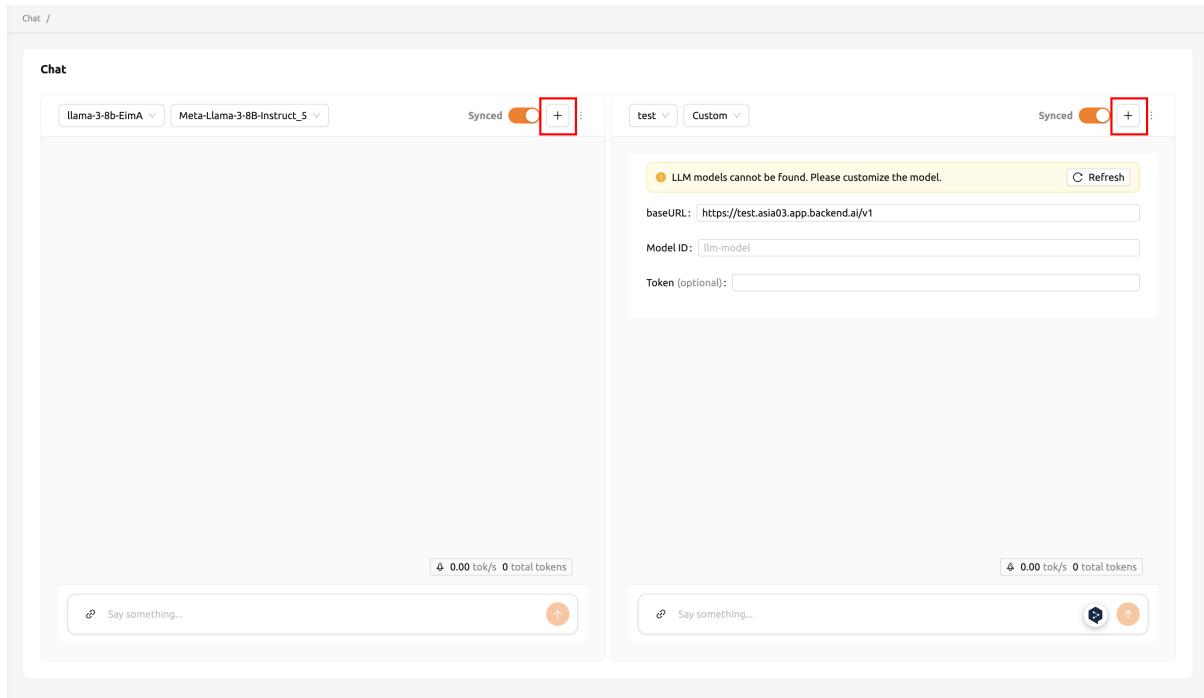


사용자 정의 모델 설정에 필요한 입력 사항은 다음 설명을 참조하십시오.

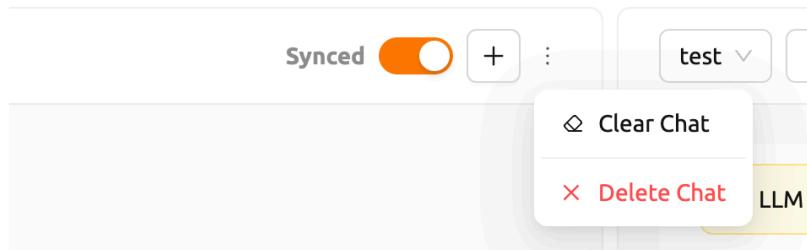
- baseURL: 모델이 위치한 서버의 기본 URL. 해당 URL에는 버전 정보가 포함되어 있어야 합니다. 예를 들어, OpenAI API를 사용하는 경우, <https://api.openai.com/v1> 을 입력해야 합니다.
- Model ID: 사용하려는 모델의 고유 식별자. 모델을 지정하려면 고유 식별자를 제공해야 합니다. 예를 들어, GPT-4의 경우 gpt-4o를 입력합니다.
- Token (선택사항): 모델 서비스에 접근하기 위한 인증 키. 토큰은 Backend.AI 뿐만 아니라 다양한 서비스에서 생성할 수 있습니다. 토큰 생성 방법은 서비스에 따라 다를 수 있습니다. 자세한 내용은 해당 서비스의 안내를 참조하십시오. Backend.AI에서 생성한 서비스를 사용하는 경우, 토큰 생성 방법은 [토큰 발급하기](#) 섹션을 참조하십시오.

13.2 채팅 카드 추가 및 삭제

우측 상단의 '+' 버튼을 클릭하여 새로운 채팅 카드를 추가할 수 있습니다.

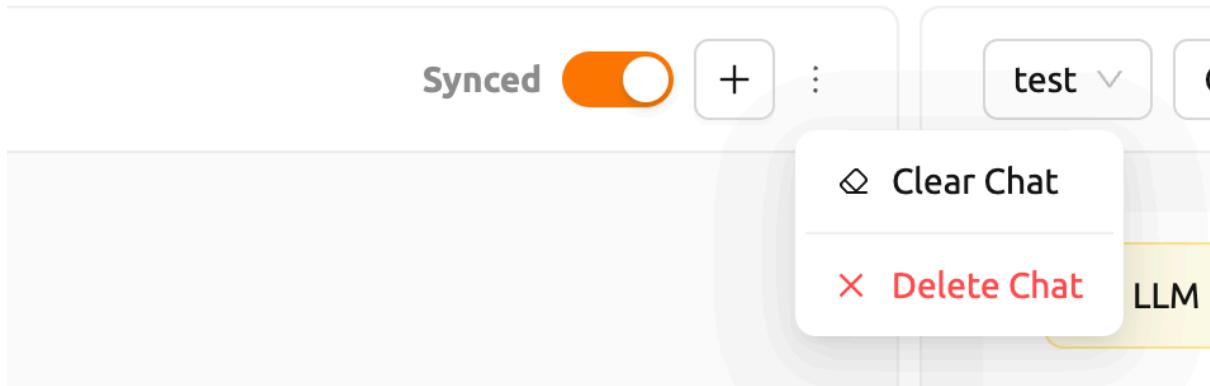


각 채팅 세션을 삭제하려면 카드 우측 상단에 있는 '더보기' 버튼을 클릭하십시오. 버튼을 클릭하면 드롭다운 메뉴가 표시되며, '채팅 삭제' 버튼을 통해 해당 채팅 세션을 삭제할 수 있습니다. 입력한 내용이 존재하는 경우에는 입력한 내용이 모두 삭제되므로 주의해 주세요.



13.3 채팅 기록 삭제

'더보기' 버튼을 클릭하면 '채팅 지우기' 버튼이 나타납니다. 이 버튼을 클릭하면 카드에 있는 모든 채팅 기록이 삭제됩니다. 단, 카드의 세션은 종료되지 않습니다.



13.4 입력 연동

채팅 카드의 우측 상단의 '동기화' 버튼을 사용하면 '동기화'가 활성화된 모든 채팅 카드의 입력을 동기화할 수 있습니다. '동기화'가 활성화된 채로, '엔터'를 누르거나 어떤 카드의 '전송' 버튼을 클릭하면 현재 입력 중인 카드의 입력이 일괄 적용됩니다. 이 기능은 하나의 입력값을 통해 다양한 모델의 응답 결과를 비교하고 싶을 때 유용하게 사용할 수 있습니다.

Chat

Synced + :

llama-3-8b-EinA Meta-Llama-3-8B-Instruct_5 Synced + :

bubble sort algorithm

Bubble sort is a simple sorting algorithm that works by repeatedly stepping through the list, comparing each pair of adjacent items and swapping them if they are in the wrong order. The pass through the list is repeated until the list is sorted.

Here is a step-by-step example of how bubble sort works:

1. Start at the beginning of the list.
2. Compare the first two items in the list. If they are in the wrong order, swap them.
3. Move to the next two items in the list and repeat step 2.
4. Continue this process until the end of the list is reached.
5. Repeat steps 1-4 until no more swaps are needed, which means the list is sorted.

Here is some sample code in Python to illustrate the bubble sort algorithm:

```
def bubble_sort(arr):
    n = len(arr)
    for i in range(n-1):
        for j in range(0, n-i-1):
            if arr[j] > arr[j+1]:
                arr[j], arr[j+1] = arr[j+1], arr[j]
```

112.75 tok/s 459 total tokens

This is synchronized input

Synced + :

qwen_qwq_32b QwQ-32B Synced + :

bubble sort algorithm

Okay, the user is asking about the Bubble Sort algorithm. Let me start by recalling what I know about it. Bubble Sort is a simple comparison-based algorithm, right? It works by repeatedly stepping through the list, comparing adjacent elements and swapping them if they're in the wrong order. This process repeats until the list is sorted.

Hmm, I should explain the basic idea first. Maybe start with a brief overview. Then, perhaps outline the steps involved. Let me think of the steps again. The algorithm passes through the list multiple times. Each pass moves the largest unsorted element to its correct position, which is why it's called "bubble" sort—like bubbles rising to the top.

Wait, I should mention that it's not the most efficient for large datasets because its average and worst-case time complexity is $O(n^2)$. But it's easy to understand and implement, so that's a plus. Maybe give an example to illustrate how it works step by step. Let's see, if I have an example array like [5, 3, 8, 4, 2], how would each pass look?

First pass: Compare 5 and 3, swap them to get [3, 5, 8, 4, 2]. Then 5 and 8 stay, next 8 and 4 swap to [3, 5, 4, 8, 2], then 8 and 2 swap to [3, 5, 4, 2, 8]. So after the first pass, the largest element 8 is at the end. Second pass would go through the first four elements. Compare 3 and 5 (no swap), 5 and 4 (swap to [3, 4, 5, 2, 8]), then 5 and 2 (swap to [3, 4, 2, 5, 8]). Now the second largest 5 is in place. Third pass would handle the first three elements 3 and 4 (no swap).

32.89 tok/s 1187 total tokens

This is synchronized input

CHAPTER 14

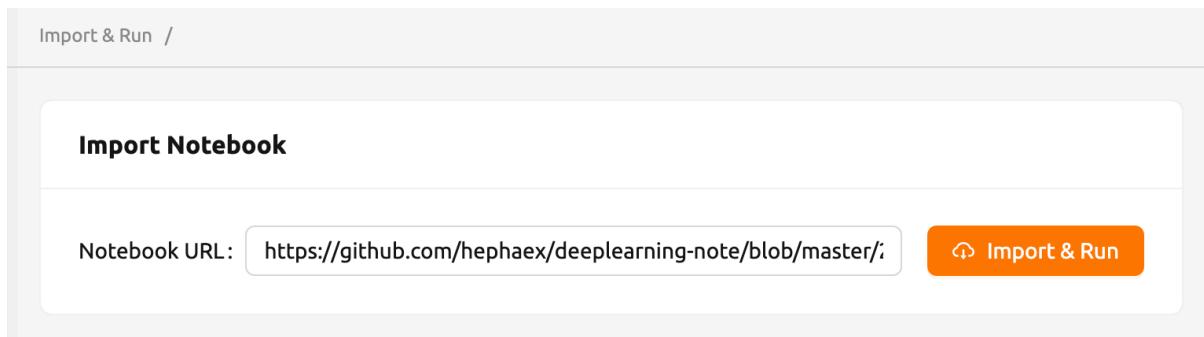
Jupyter notebook과 웹 기반 Git 저장소 불러오기 및 실행

'가져오기 & 실행' 페이지에서, 즉석으로 Jupyter notebook 파일을 실행하거나 GitHub, GitLab과 같은 웹 기반 Git 저장소를 가져올 수 있습니다. 파일을 받거나 실행하기 위해 로컬 저장소에서 직접 생성하거나 다운로드 받고 다시 업로드하지 않아도 됩니다. 유효한 URL을 입력하고 각 기능에 해당하는 패널의 우측에 있는 버튼을 클릭하기만 하면 됩니다.

14.1 Jupyter notebook 파일 가져오기 및 실행

Jupyter notebook 파일을 가져와서 실행하기 위해서는 notebook 파일에 대한 유효한 URL이 필요합니다. GitHub에 있는 Jupyter notebook을 실행하고자 한다면, 입력 필드에 해당 파일의 URL을 복사해서 붙여넣은 뒤, '가져와서 실행' 버튼을 클릭하면 됩니다.

참고: 로컬 주소의 Jupyter notebook 파일을 불러와서 실행할 경우 실행이 되지 않습니다. localhost로 시작되는 주소 이외의 URL을 입력해주십시오.



버튼을 클릭하면 다이얼로그가 띄워집니다. 이 다이얼로그는 세션 페이지나 요약 페이지에서 세션 시작시 나타나는 다이얼로그와 같습니다. 다만 notebook 불러오기에서는 세션을 시작하면서 불러온 jupyter notebook을 자동으로 실행하는 것이지만, 단순히 세션을 시작하는 것은 그렇지 않다는 점이 다릅니다. 나머지는 동일합니다. 실행 환경과 필요한 자원 할당을 완료한 뒤 '시작' 버튼을 클릭하여 세션을 생성합니다.

경고: 노트북 실행 화면을 즉시 확인하려면 실행 전 반드시 팝업 차단을 해제해 주시기 바랍니다. 또한 세션을 생성하기에 충분한 자원이 없을 경우, 불러온 Jupyter notebook 파일은 실행되지 않습니다.

Sessions / Start new session /

Import & Run / Start new session

Start new session

Session Type Edit

SessionType: interactive
Session name: imported-notebook-EXUAA

Environments Edit

Project: default
Image: Python 3.11 | aarch64 | Ubuntu 22.04 | Arm | 64 |

Resource allocation Edit

Resource Group: default
Resource Allocation Per Container: minimum-required | 1 Core | 0.313 GiB (SHM: 0.06GiB)
Agent: auto Number of container: 1
Cluster mode: Single Node

Total Allocation
1 Core | 0.313 GiB (SHM: 0.06GiB)

Data & Storage Edit

No storage folder is mounted

Network Edit

Preopen Ports: None

Reset ◀ Previous ⟳ Launch

1 Session Type
2 Environments & Resource allocation
3 Data & Storage
4 Network
▶ Confirm and Launch

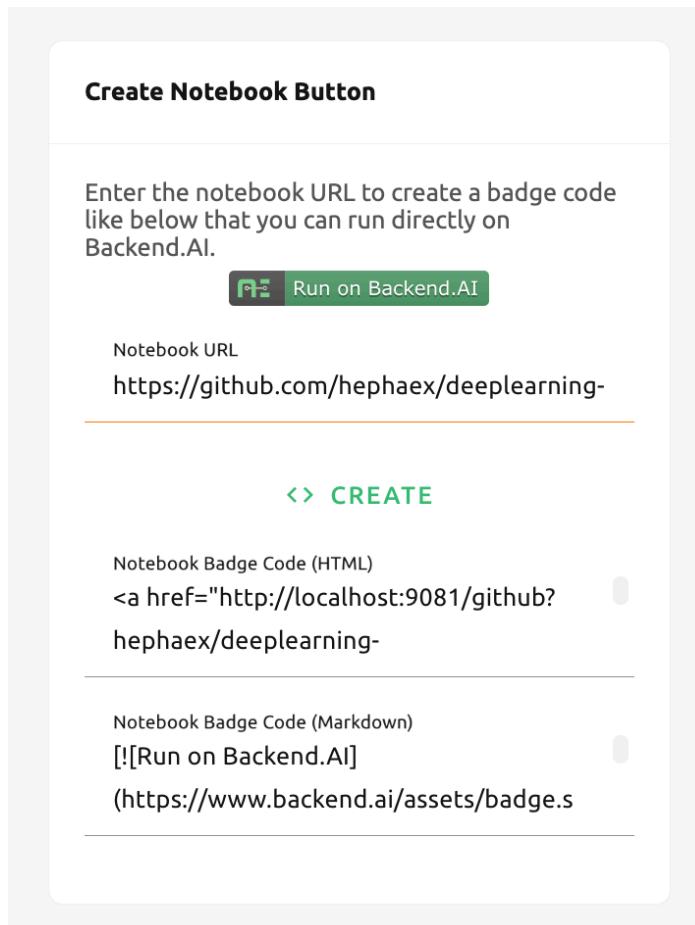
세션 페이지에서 불러오기 작업이 성공적으로 완료된 것을 확인할 수 있습니다.

RUNNING	INTERACTIVE	BATCH	INFERENCE	UPLOAD SESSIONS	FINISHED	...	START	
User ID	Session Info	Status	Control	Configuration	Usage	Reservation	Session Type	Agents
admin@lablup.com	imported-notebook-EXUAA 	RUNNING		No mount default 1 core 0.31GiB (SHM: 0.06GiB) -GPU	CPU 0.1 % RAM 0.13 / 0.31 GiB I/D R: 0.0 MB / W: 0.0 MB	4/10/2025, 5:46:40 PM Elapsed Time: 00:28:22	INTERACTIVE	i-Lablupui-MacBook

14.2 Jupyter notebook을 바로 실행할 수 있는 버튼 생성하기

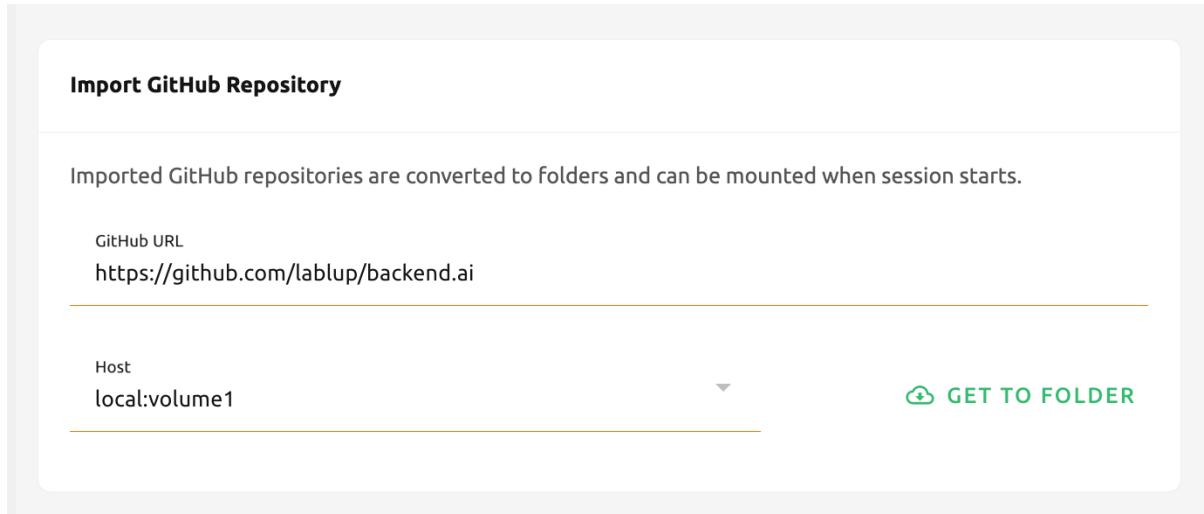
Jupyter notebook URL에 대해 HTML이나 Markdown에 삽입할 수 있는 버튼도 만들 수 있습니다. 유효한 Jupyter notebook URL을 입력 필드 란에 기입하고, '배지 코드 생성' 버튼을 클릭 하십시오. 노트북 파일을 바로 실행하는 세션을 생성해주는 코드 블록이 생성됩니다. 배지 코드가 잘 동작하는 것을 GitHub 저장소 또는 html이나 markdown 파일을 지원하는 그 어떤 곳에서도 사용이 가능합니다.

참고: 버튼 클릭 전 반드시 로그인이 되어있어야 합니다. 아닐 경우 실행 전 반드시 로그인이 필요합니다.



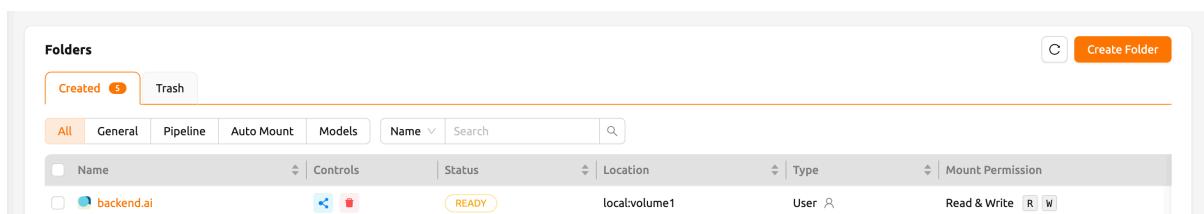
14.3 GitHub 저장소 불러오기

GitHub 저장소 불러오기는 Jupyter notebook 불러오기 및 실행하기와 비슷합니다. GitHub 저장소의 URL을 입력 필드에 적은 뒤 '가져오기' 버튼을 클릭하면 됩니다. 하나 이상의 스토리지 호스트를 사용하는 경우에는 Host 목록에서 하나를 선택하십시오.



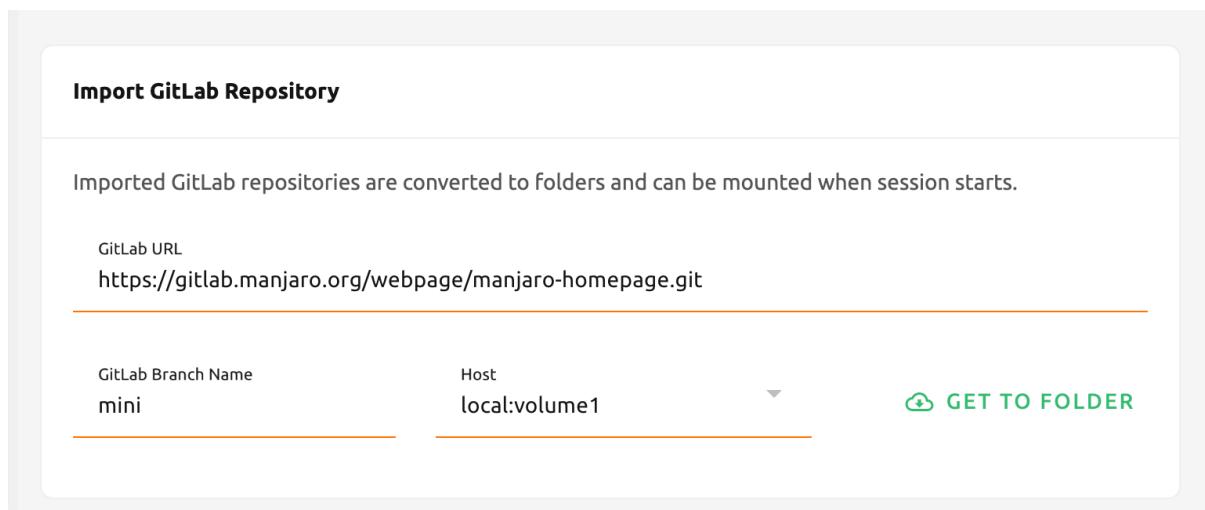
경고: 만약 세션을 생성할 수 있는 충분한 자원이 없거나, 폴더 생성 가능한 갯수가 최대 값에 도달했을 경우, 저장소 불러오기 작업은 불가능합니다. 저장소를 불러오기 전 반드시 Resource statistics 패널과 데이터 페이지를 확인하시기 바랍니다.

저장소가 하나의 폴더로써 성공적으로 불러와진 것을 확인할 수 있습니다.



14.4 GitLab 저장소 불러오기

Backend.AI 22.03 버전부터 GitLab 저장소 불러오기를 지원합니다. [GitHub 저장소 불러오기](#)와 거의 같은 방법으로 불러올 수 있지만, GitLab의 경우에는 브랜치 이름을 명시적으로 지정해야 합니다.



참고: 이미 동일한 이름의 데이터 폴더가 존재할 경우, 시스템에서는 자동으로 불러오는 저장소 명 뒤에 _ (밑줄)과 번호를 추가하게 됩니다.

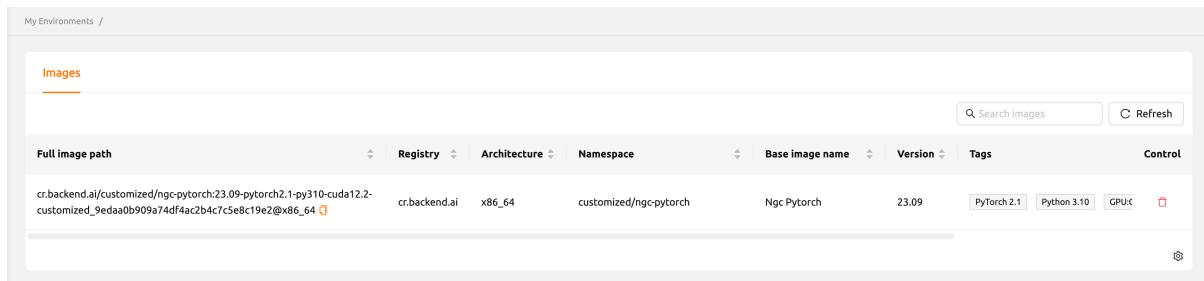
CHAPTER 15

나의 실행 환경

24.03 버전부터, 사용자를 위한 새로운 “나의 실행 환경” 페이지가 추가되었습니다. 이 페이지는 사용자가 [세션을 커밋해서 생성된 이미지](#)들의 리스트를 보여줍니다.

나의 실행 환경 페이지의 이미지 탭에서 사용자는 연산 세션 생성에 사용되는 사용자 정의 이미지를 관리할 수 있습니다. 이 탭은 연산 세션으로부터 변환된 이미지들의 메타 정보가 표시됩니다. 각 이미지에 대한 레지스트리, 아키텍쳐, 네임스페이스, 언어, 버전, 기반, 제약, digest 등의 정보를 확인할 수 있습니다.

이미지를 삭제하고 싶다면 제어 열의 빨간색 휴지통 버튼을 클릭합니다. 삭제 후에는 해당 이미지로 새로운 세션을 생성할 수 없습니다.



사용자 정의 이미지를 복사해서 세션을 생성할 수도 있습니다. 제어 열의 '복사' 버튼을 클릭한 뒤, 세션 페이지로 이동해서 세션을 생성합니다. 방금 복사한 이미지를 직접 입력하는 이미지의 입력으로 붙여넣기 하면 됩니다.

Environments

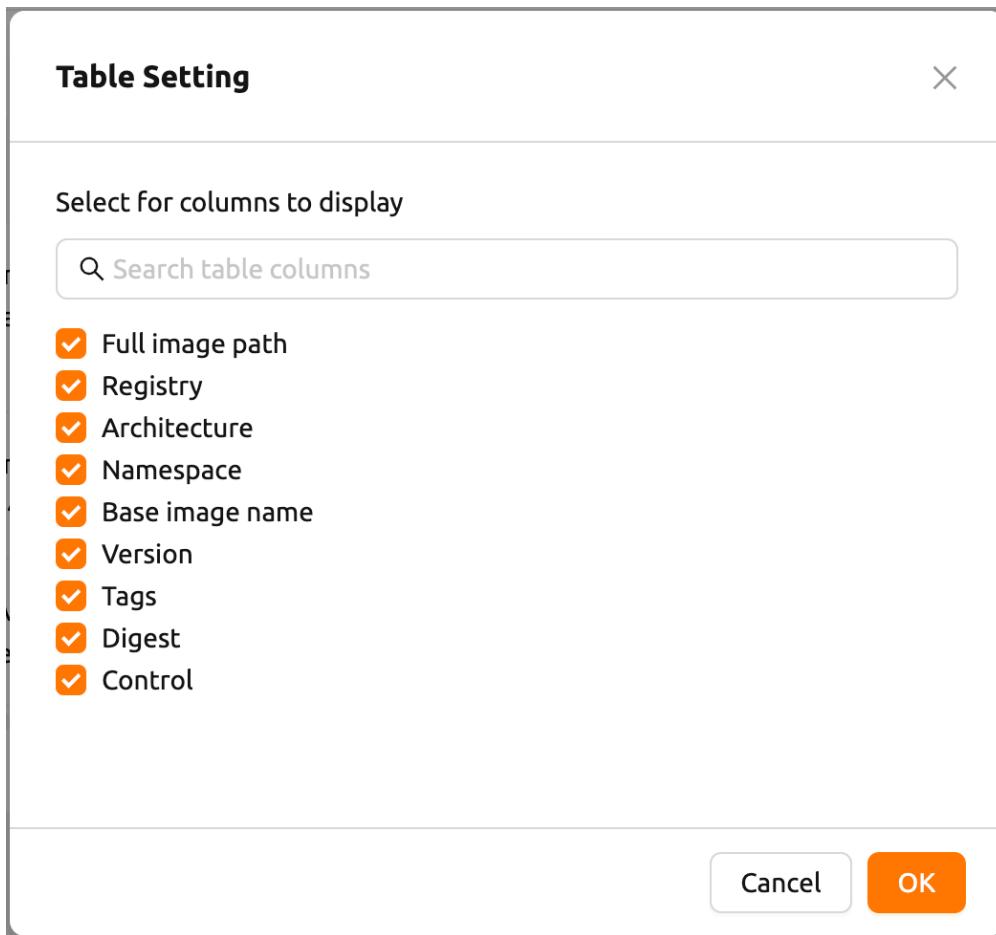
Environments / Version  (optional)

Image Name (Manual) (optional)
cr.backend.ai/customized/ngc-pytorch:23.09-pytorch2.1-py310-cuda12.2-customized_9edaa0b90 

Variable (optional)

+ Add environment variables

특정 열을 숨기거나 보이고 싶다면 테이블 우측 하단의 기어 아이콘을 클릭합니다. 그러면 아래 다이얼로그가 표시되어 보고 싶은 열을 선택할 수 있습니다.



CHAPTER 16

자원 요약

지금까지는 관리자 권한을 가진 사용자만 관리자 메뉴를 통해 에이전트 자원 정보를 확인할 수 있었습니다. 22.09 버전부터 Backend.AI WebUI에서는 설정 값에 따라 에이전트 노드의 부분적인 정보를 제공합니다. 자원 요약 메뉴에서는 에이전트 별 엔드포인트 주소, CPU 아키텍쳐, 자원 할당량, 그리고 에이전트의 스케줄링 가능 상태 여부와 같은 정보를 확인할 수 있습니다. 이 메뉴는 세션 생성 시 자원 할당을 하는 데에 유용합니다.

The screenshot shows the 'Agent Summary' page in the Backend.AI WebUI. At the top, there are tabs for 'Connected' (which is selected) and 'Terminated'. Below the tabs is a search bar with fields for 'ID' and 'Search' and a magnifying glass icon. To the right of the search bar is a clear button (C). The main table has columns: 'ID', 'Architecture', 'Allocation', and 'Schedulable'. A single row is displayed for the agent 'i-sj-bai-latest', which has an architecture of 'x86_64'. In the 'Allocation' column, there are two progress bars: one for 'Core' (4/7, 57.1%) and one for 'GB' (4/30, 13.2%). The 'Schedulable' column contains a green checkmark. At the bottom of the table, it says '1-1 of 1 items' with navigation arrows and a page size selector set to '20 / page'. There is also a gear icon for settings.

참고: 서버 설치 및 설정 환경에 따라, 자원 요약 기능이 비활성화되어 있을 수 있습니다. 자원 요약 기능 사용을 원하시는 경우, 시스템 관리자에게 문의하십시오.

CHAPTER 17

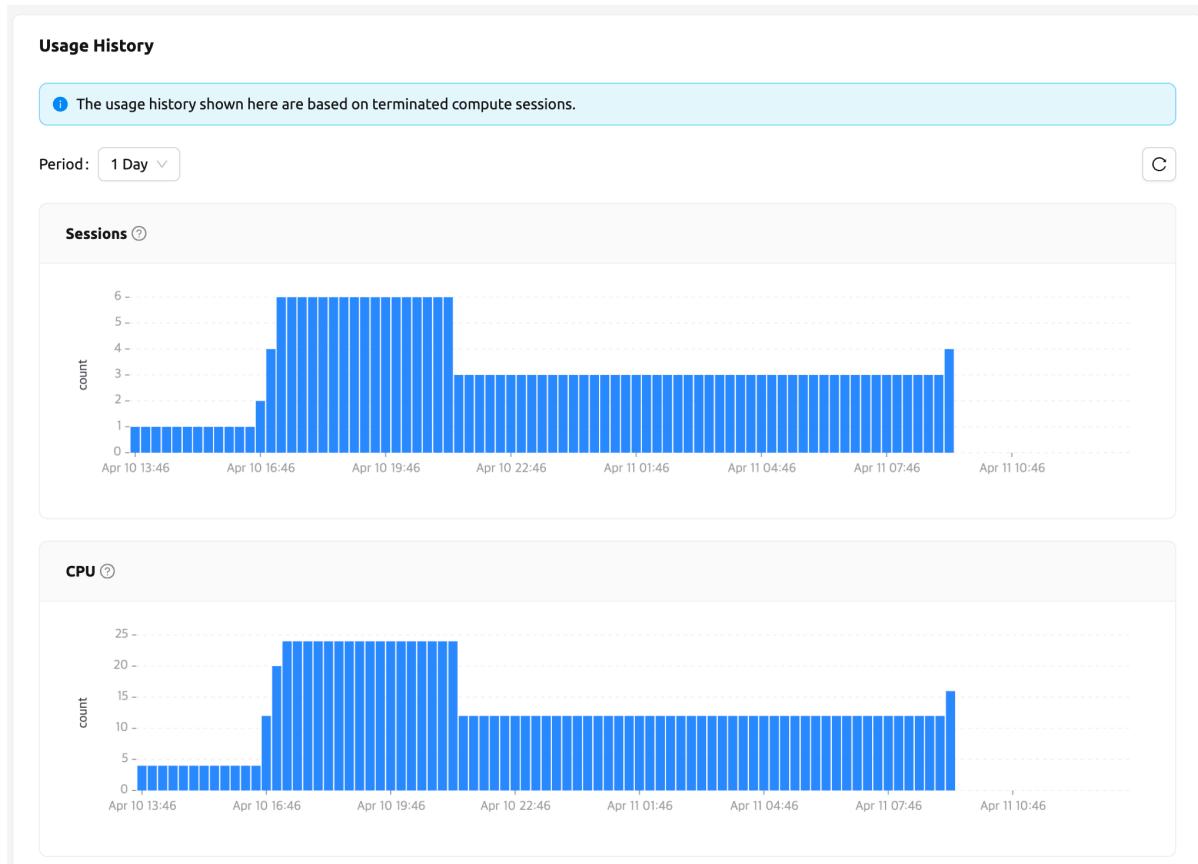
Statistics 페이지

통계 페이지에서는 연산 세션 사용과 관련된 간단한 통계를 그래프로 확인할 수 있습니다. 좌측 상단의 기간 선택 메뉴에서 이용 기간을 선택하여 하루 또는 일주일 동안의 통계를 확인할 수 있습니다. 표시되는 항목은 다음과 같습니다.

- Sessions: 생성된 연산 세션의 수입니다.
- CPU: 연산 세션이 생성되면서 할당한 CPU 코어의 수입니다.
- Memory: 연산 세션이 생성되면서 할당한 메모리의 양입니다.
- GPU: 연산 세션이 생성되면서 할당한 GPU 단위의 수입니다. Fractional GPU 기능이 작동하고 있을 경우, 물리 GPU 와 일치하지 않을 수 있습니다.
- IO-Read: 스토리지로부터 읽은 데이터의 양입니다.
- IO-Write: 스토리지에 기록된 데이터의 양입니다.

여기에 나타나는 통계는 종료된 연산 세션을 기준으로 합니다. 또한, 가입 일주일 미만의 사용 자에게는 주간 통계 조회 메뉴가 출력되지 않을 수 있습니다.

보다 자세한 통계 기능은 관리자 전용 Control-Panel에서 제공합니다.

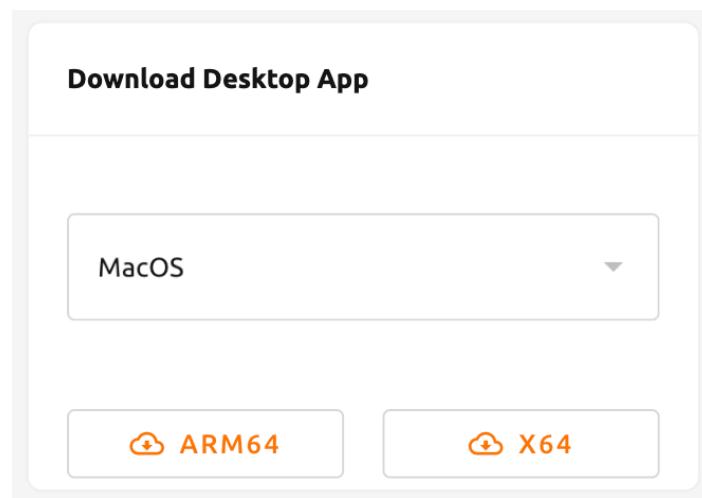


CHAPTER 18

연산 세션에 SSH/SFTP 접속하기 (Linux/Mac/Visual Studio Code)

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

참고: 24.03 버전부터 연산세션에 SSH/sFTP 접속하는 기능을 웹 기반 Web UI 와 데스크탑 앱에서 모두 지원합니다. 현재 버전이 23.09 이거나 그 이전 버전일 경우, 데스크탑 앱에서만 해당 기능을 지원합니다. 데스크탑 앱은 Summary 페이지의 별도 다운로드 패널에서 다운로드할 수 있습니다. 이 패널에서 다운로드하는 경우에는 호환 버전을 자동으로 다운로드 합니다.



데스크탑 앱은 <https://github.com/lablup/backend.ai-webui/releases>에서도 수동 다운로드 할 수 있습니다. 이 경우에는 Backend.AI 클러스터 서버 버전과 호환되는 Web-UI 앱을 받아야 최상의 호환성을 유지할 수 있습니다. Backend.AI 클러스터의 버전은 GUI 우측 상단에 있는 설정 아이콘을 클릭하면 나오는 “About Backend.AI” 메뉴를 클릭하여 확인할 수 있습니다.

18.1 Linux / Mac 환경

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 Web-UI 앱과 컨테이너 내부의 데몬이 연결됩니다.

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

이어 SSH/SFTP connection 정보를 담고 있는 디아일로그가 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.

The screenshot shows two panels side-by-side. The left panel, titled 'App', contains sections for Utilities (Console, SSH / SFTP), Development (Visual Studio Code, Visual Studio Code (Desktop), JupyterLab, Jupyter Notebook), Machine Learning Tools (TensorBoard), and Preopen Ports (mlflow-ui, nnboard). It also includes checkboxes for 'Open app to public' and 'Try preferred port'. The right panel, titled 'SSH / SFTP connection', displays connection information (User: work, Host: proxy.asia03t.app.test.com, Port: 10305), a connection example (sftp command), and a large orange 'DOWNLOAD SSH KEY' button.

다운로드 받은 SSH private key로 연산 세션에 SSH 접속을 하려면 쉘 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 work로 설정되는데, 만약 work가 아닌 다른 계정을 사용하는 경우라면 work@127.0.0.1에서 work 부분을 실제 접속되는 계정으로 바꾸어 주어야 합니다. 정상적으로 명령을 내렸다면 연산 세션에 SSH 접속이 이루어지고 컨테이너 내부의 쉘 환경에 접속되는 것을 확인할 수 있습니다.

```
$ ssh
> -i ~/.ssh/id_container -P 30722 \
> -o StrictHostKeyChecking=no \
```

(다음 페이지에 계속)

(이전 페이지에서 계속)

```
> -o UserKnownHostsFile=/dev/null \
> work@127.0.0.1
Warning: Permanently added '[127.0.0.1]:30722' (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 참고 하십시오.

참고: 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.

참고: 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.

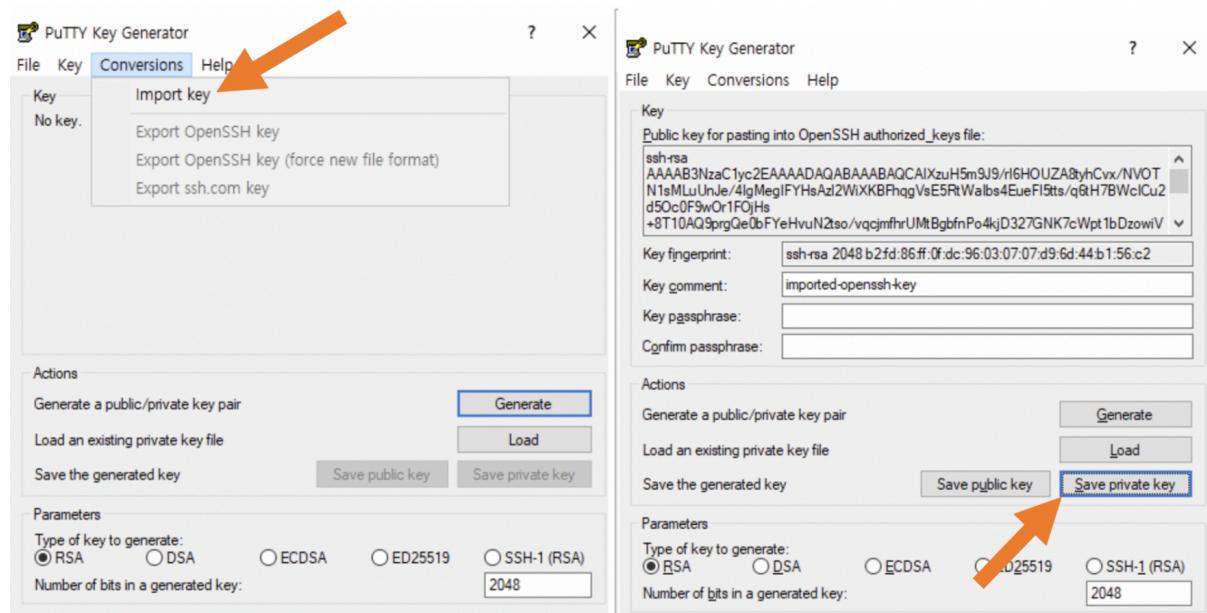
참고: 만약 다음과 같은 경고 메시지가 뜨면, `id_container` 의 권한을 600으로 설정한 후 재시도 합니다. (`chmod 600 <id_container 경로>`)

```
Warning: Permanently added '<SSH address>' (RSA) to the list of known hosts.
@WARNING: UNPROTECTED PRIVATE KEY FILE!@
Permissions 0755 for '<id_container path>' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
Load key "<id_container path>": bad permissions
work@localhost's password:
```

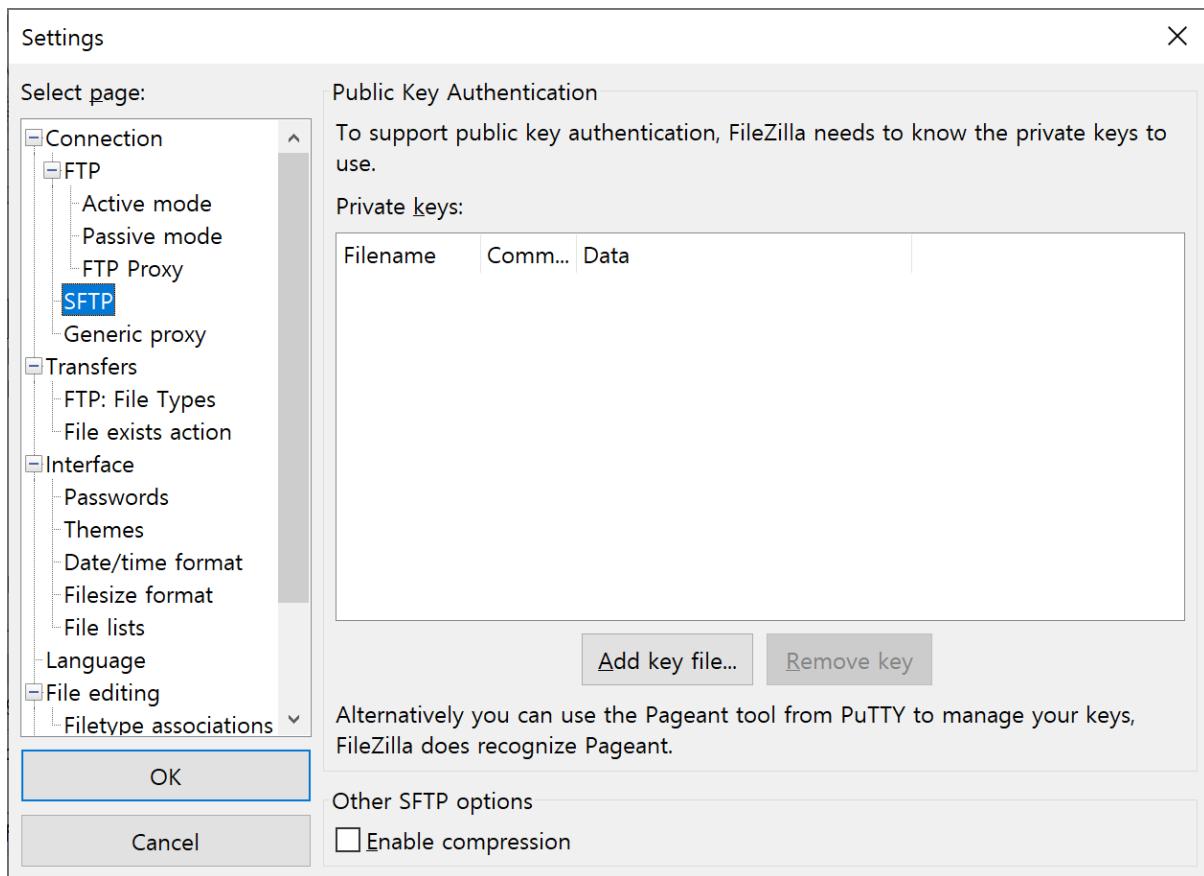
18.2 Windows / FileZilla 사용 환경

Backend.AI Web-UI 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.

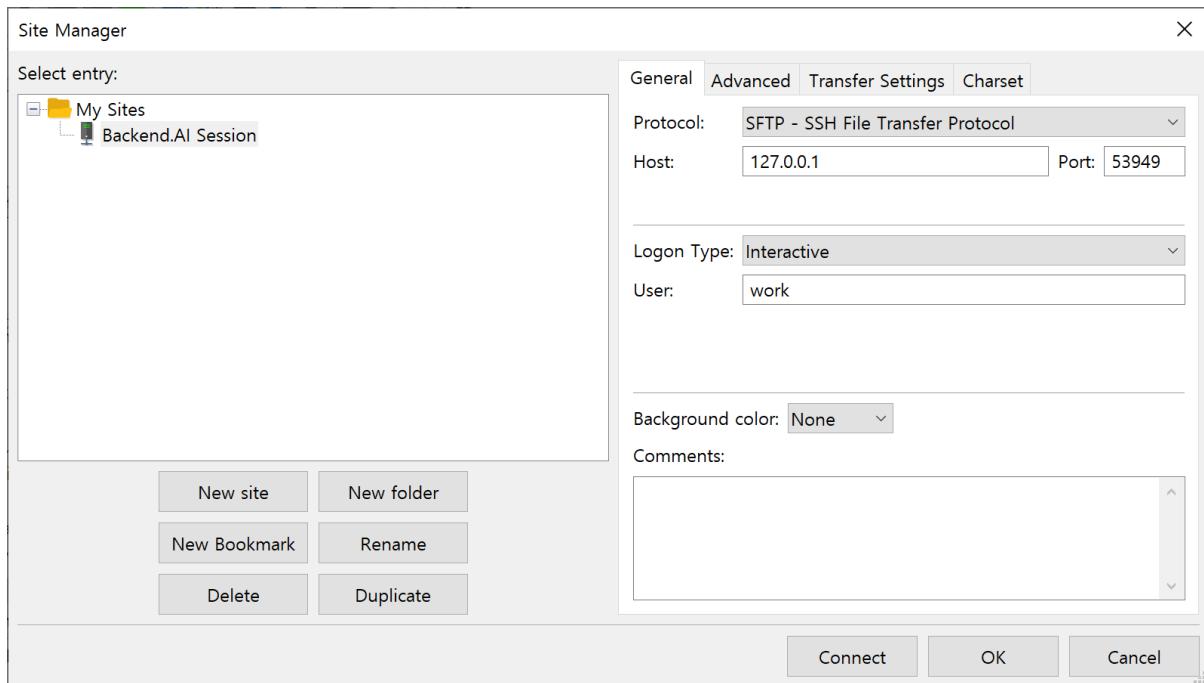
Linux/Mac에서의 접속 방법을 참고하여 연산 세션을 생성하고 접속 포트 확인과 id_container 다운로드를 수행합니다. id_container 는 OpenSSH 기반의 키이므로, 윈도우 또는 ppk 형태의 키만 지원하는 클라이언트를 사용할 경우 변환을 해야 합니다. 여기서는 PuTTY와 함께 설치되는 PuTTYgen 프로그램을 통해 변환을 해보겠습니다. PuTTYgen 프로그램을 실행한 후 Conversions 메뉴의 import key를 클릭합니다. 이어 또는 파일 선택 다이얼로그에서 다운로드 한 id_container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



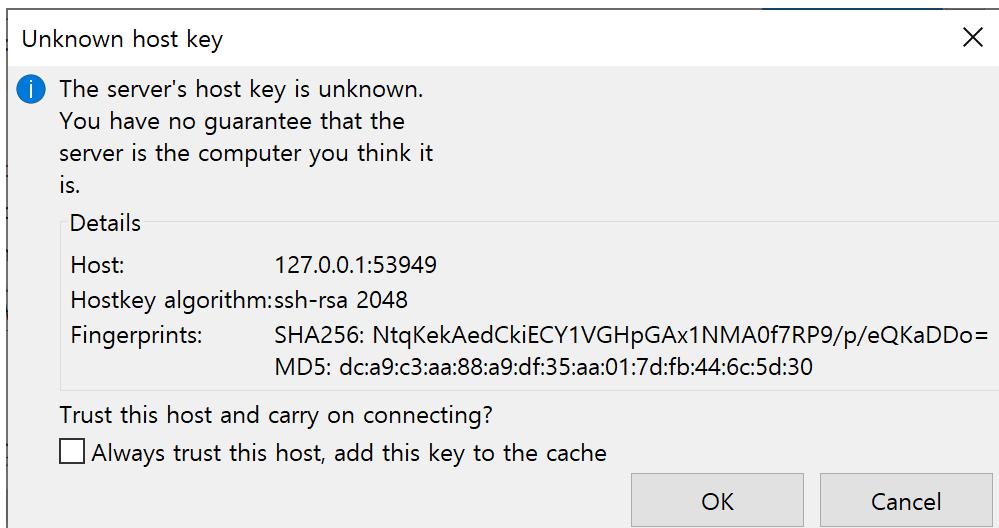
이제 FileZilla 클라이언트 앱을 실행한 후 설정-연결-SFTP 페이지로 들어가 생성한 id_container.ppk (OpenSSH를 지원하는 클라이언트의 경우 id_container) 키 파일을 등록해 줍니다.



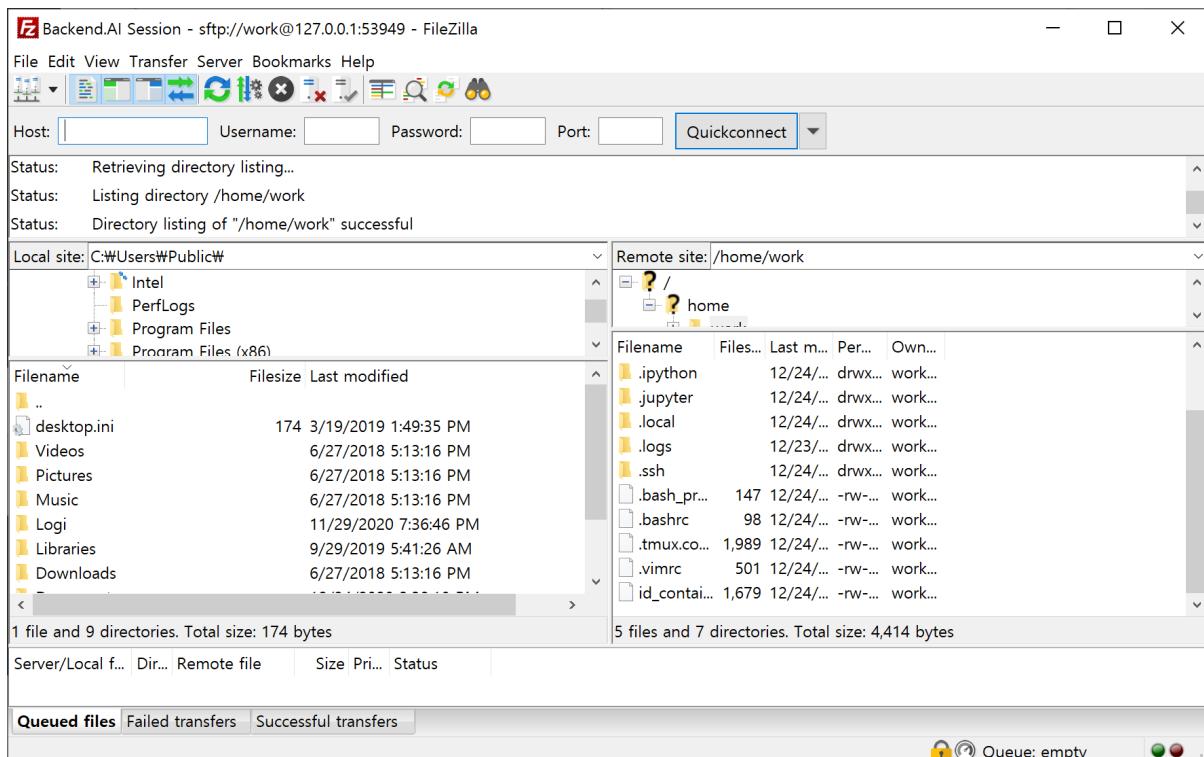
사이트 관리자를 열고 새 사이트를 하나 만든 후 다음과 같이 접속 정보를 적어 줍니다.



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



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.

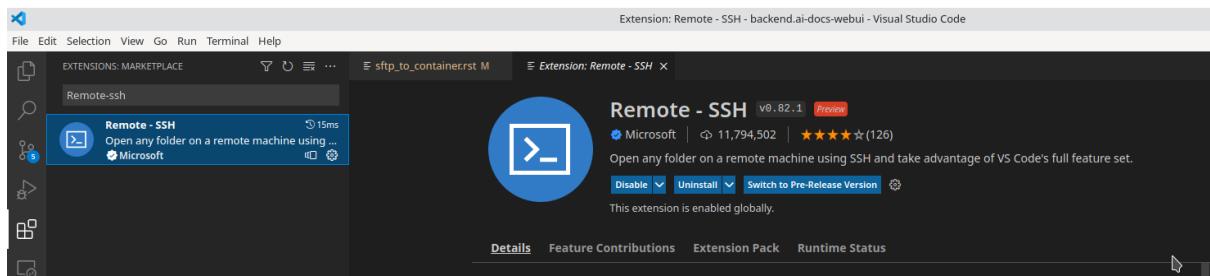


18.3 For Visual Studio Code

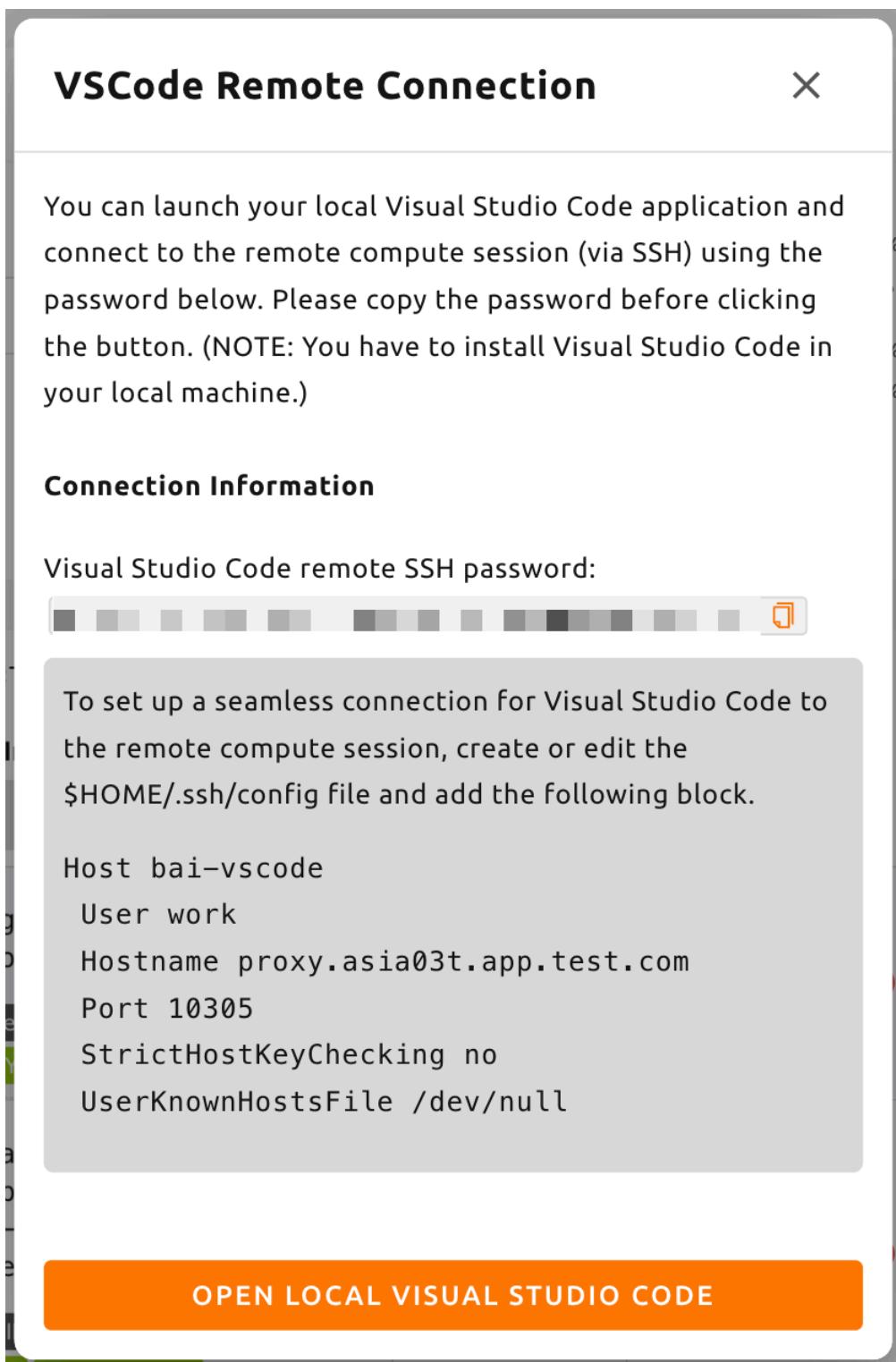
로컬 Visual Studio Code에서 SSH/SFTP를 통해 Backend.AI 연산 세션에 접속하여 원격 개발할 수도 있습니다. 한번 연결을 하면, 연산 세션 상에 존재하는 모든 파일과 폴더를 활용하여 작업할 수 있습니다. 이번 절에서는 그 방법에 대해 알아봅니다.

우선, Visual Studio Code와 원격 개발(Remote Development) 확장팩을 설치해야 합니다.

링크: <https://aka.ms/vscode-remote/download/extension>



확장팩 설치 후 연산 세션 접속을 위한 SSH 연결 설정을 진행합니다. VSCode Remote Connection 다이얼로그에서 복사 아이콘 버튼을 클릭하여 Visual Studio Code 원격 비밀번호를 복사합니다. 포트 번호도 기억해 두십시오.



다음으로 SSH 설정 파일을 수정합니다. `~/.ssh/config` 파일(Linux/Mac) 또는 `C:\Users\[user name]\.ssh\config` (Windows) 파일을 텍스트 편집기로 열고 다음 블록을 추가합니다. 편의를 위해 호스트 이름을 `bai-vscode`로 설정했지만, 이는 원하는 대로 바꾸셔도 됩니다.

```

Host bai-vscode
User work

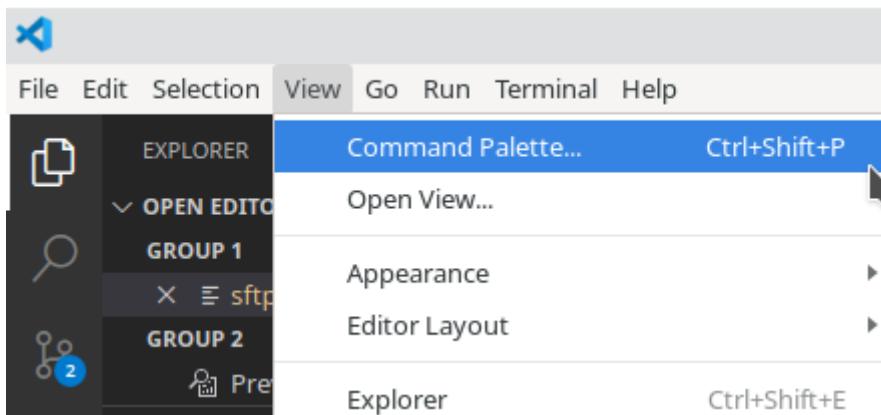
```

(다음 페이지에 계속)

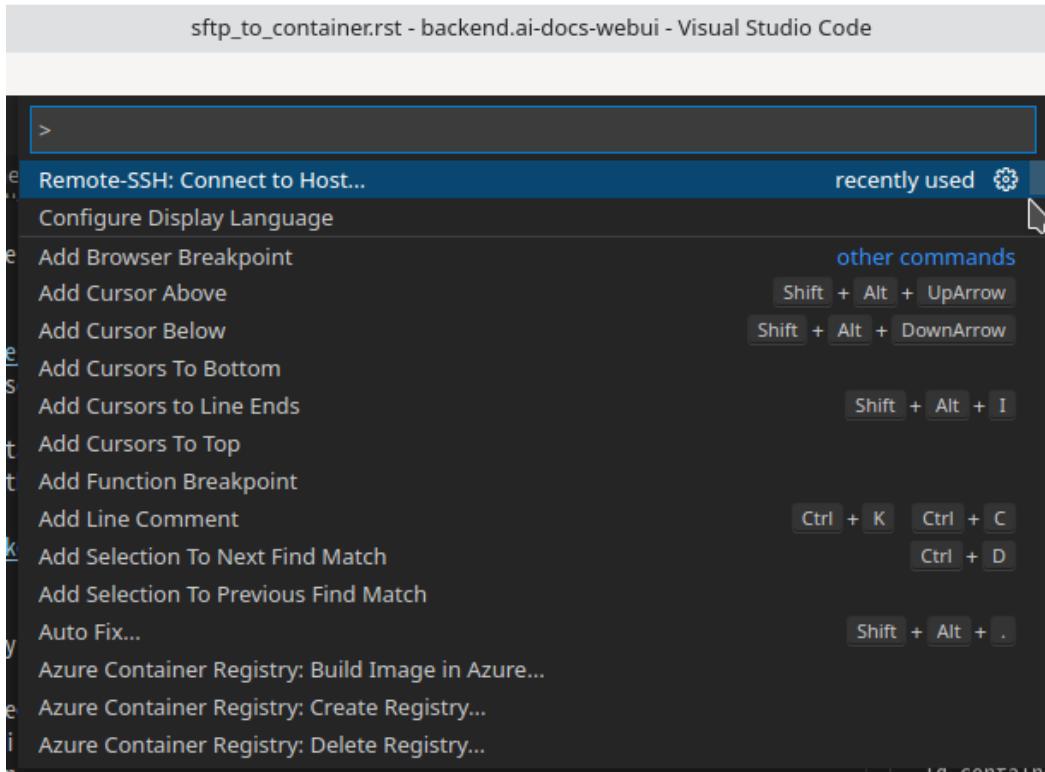
(이전 페이지에서 계속)

```
Hostname 127.0.0.1
# write down the port number that you remembered
Port 49335
StrictHostKeyChecking no
UserKnownHostsFile /dev/null
```

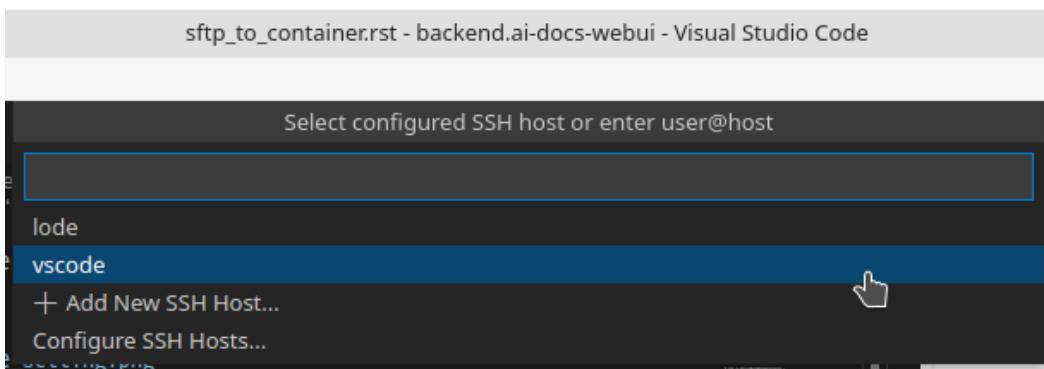
이제 Visual Studio Code의 View - Command Palette... 메뉴를 선택합니다.



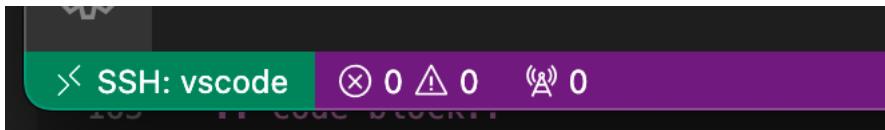
Visual Studio Code는 연결할 호스트의 태입을 자동으로 탐지합니다. Remote-SSH: Connect to Host...를 선택해봅시다.



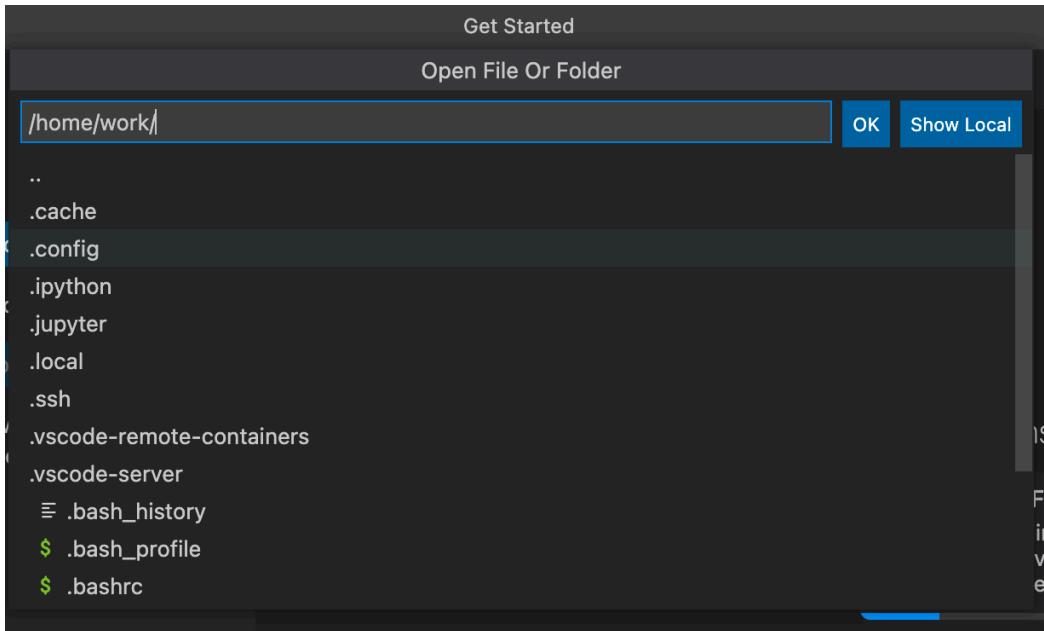
여기서는 .ssh/config의 호스트 목록을 볼 수 있습니다. 연결할 호스트(vscode)를 선택합니다.



연결이 된 후 빈 창이 표시됩니다. 상태 표시 줄을 참조하면 연결된 호스트를 확인할 수 있습니다.
연결이 된 후 빈 창이 표시됩니다. 상태 표시 줄을 참조하면 연결된 호스트를 확인할 수 있습니다.



로컬에서와 마찬가지로 파일 > 열기... 또는 파일 > 워크스페이스 열기... 메뉴를 통해 원격 호스트에서 폴더나 워크스페이스를 열 수 있습니다!



18.4 Backend.AI Client 패키지를 통해 SSH 접속하는 방법

이 문서에서는 GUI 환경 사용이 불가능한 곳에서 연산 세션으로 SSH를 접속하는 방법에 관해 설명합니다.

일반적으로 연산 세션이 생성된 GPU 노드는 외부에서 직접 접속이 불가능합니다. 따라서 연산 세션에 SSH 또는 sFTP 접속을 하기 위해서는 사용자 머신과 GPU 노드에 생성된 연산 세션 사이의

터널을 중계하는 로컬 프록시가 필요합니다. Backend.AI Client 패키지를 활용하면 이 과정을 비교적 간단하게 설정할 수 있습니다.

18.4.1 Backend.AI Client 패키지 사용 환경 구성

Docker image를 이용한 구성

Backend.AI Client 패키지를 도커 이미지로 다운로드할 수 있습니다. 다음 명령을 실행하여 Docker Hub에서 이미지를 다운로드할 수 있습니다:

```
$ docker pull lablup/backend.ai-client
$ # If you want to use the specific version, you can pull the image
  ↪with the following command:
$ docker pull lablup/backend.ai-client:<version>
```

Backend.AI 서버 버전은 Web UI의 우상단 사람 아이콘을 클릭하면 나오는 “Backend.AI에 대하여”에서 확인할 수 있습니다.

Backend.AI Web UI 23.09.11

Open Source Edition

Backend.AI Cluster 23.09.11

WebServer Build

다운 받은 이미지를 다음 명령으로 실행합니다.

```
$ docker run --rm -it lablup/backend.ai-client bash
```

backend.ai 명령이 존재하는지 확인합니다. 정상적으로 진행되었다면 도움말이 출력됩니다.

```
$ backend.ai
```

Python 가상환경을 통해 호스트에서 사용 환경 구성하기

Docker를 사용할 수 없거나 사용하고 싶지 않다면, 호스트 머신에 Backend.AI Client 패키지를 직접 설치할 수도 있습니다. 사전 요구 사항은 다음과 같습니다:

- Python 버전은 Backend.AI Client 버전에 따라 바뀝니다. <https://github.com/lablup/backend.ai#python-version-compatibility> 페이지에서 호환 매트릭스를 확인할 수 있습니다.
- clang 컴파일러 설치가 필요할 수 있습니다
- indygreg Python 바이너리를 사용하는 경우 zstd 패키지를 미리 설치해야합니다.

Client 패키지 설치를 위해 별도의 Python 가상 환경 사용을 권장합니다. 한 가지 방법은 `indygreg` 저장소에 있는 정적 빌드된 Python 바이너리를 사용하는 것입니다. 다음 사이트에서 Client 패키지를 설치할 머신 아키텍처에 맞는 Python 바이너리를 다운로드한 후 압축을 해제합니다.

- <https://github.com/indygreg/python-build-standalone/releases>
- 대중적으로 많이 사용되는 x86 기반 Ubuntu 환경이라면 다음과 같이 다운로드 할 수 있습니다:

```
$ wget https://github.com/indygreg/python-build-standalone/
  ↳releases/download/20240224/cpython-3.11.8+20240224-x86_64-
  ↳unknown-linux-gnu-pgo-full.tar.zst
$ tar -I unzstd -xvf *.tar.zst
```

압축을 풀면 현재 디렉토리 하위에 `python` 디렉토리가 생길 것입니다. 다음 명령을 실행하여 다운로드 한 파이썬의 버전을 확인할 수 있습니다.

```
$ ./python/install/bin/python3 -V
Python 3.11.8
```

시스템의 다른 파이썬 환경에 영향을 주지 않기 위해 사용자 머신에 독립적인 Python 3.10 가상 환경을 생성합니다. 다음 명령 실행 시 `.venv` 디렉토리에 Python 가상 환경이 생성됩니다.

```
$ ./python/install/bin/python3 -m venv .venv
```

생성한 Python 가상환경을 활성화 합니다. 새로운 가상 환경을 활성화한 것이므로, `pip list` 명령 실행 시 `pip`과 `setuptools` 패키지만 설치되어 있을 것입니다.

```
$ source .venv/bin/activate
(.venv) $ pip list
Package      Version
```

(다음 페이지에 계속)

(이전 페이지에서 계속)

```
-----
pip      24.0
setuptools 65.0
```

Backend.AI Client 패키지를 설치합니다. 서버팜에 설치된 서버 버전에 맞추어 클라이언트 패키지를 설치합니다. 여기서는 23.09 버전이라고 가정합니다. 만약, `netifaces`라는 패키지 설치 관련 오류가 발생하는 경우, `pip`과 `setuptools`의 버전을 낮추어야 할 수도 있습니다.

```
(.venv) $ pip install -U pip==24.0 && pip install -U setuptools==65.
      ↵ 5.0
(.venv) $ pip install -U backend.ai-client~=23.09
(.venv) $ backend.ai
```

18.4.2 CLI 클라이언트에서 서버 접속 설정

.env 파일을 만들고 다음과 같이 내용을 추가합니다. `webserver-url`에는 브라우저에서 Web UI 서비스로 접속할 때 사용하는 주소를 동일하게 씁니다.

```
BACKEND_ENDPOINT_TYPE=session
BACKEND_ENDPOINT=<webserver-url>
```

다음 CLI 명령을 실행하여 서버에 연결합니다. 브라우저에서 로그인할 때 사용하는 이메일과 비밀 번호를 입력합니다. 정상적으로 실행된 경우 `Login succeeded.` 메시지가 됩니다.

```
$ backend.ai login
User ID: myuser@test.com
Password:
✓ Login succeeded.
```

18.4.3 연산 세션에 SSH/SCP 접속

브라우저에서 데이터를 복사할 폴더를 마운트 한 후 연산 세션을 생성합니다. CLI로도 생성 가능 하지만 편의를 위해 브라우저에서 생성한다고 하겠습니다. 생성된 연산 세션의 이름을 알아둡니다. 여기서는 `ibnFmWim-session`라고 가정합니다.

단순히 SSH 접속을 하고자 하는 경우에는 다음과 같은 명령을 실행하면 됩니다.

```
$ backend.ai ssh ibnFmWim-session
• running a temporary sshd proxy at localhost:9922 ...
work@main1[ibnFmWim-session]:~$
```

SSH 키 파일을 받아 ssh 명령을 명시적으로 실행하고자 하는 경우에는 먼저 다음과 같은 명령을 실행하여 로컬 머신에서 연산 세션으로 접속을 중계하는 로컬 프록시 서비스를 띄워야 합니다. -b 옵션으로 로컬 머신에서 사용할 포트(9922)를 지정할 수 있습니다.

```
$ backend.ai app ibnFmWim-session sshd -b 9922
• A local proxy to the application "sshd" provided by the session
→ "ibnFmWim-session" is available at:
tcp://127.0.0.1:9922
```

로컬 머신에서 다른 터미널 창을 띄웁니다. .env 파일이 있는 작업 디렉토리로 이동한 후 연산 세션에서 자동 생성된 SSH 키를 다운로드 합니다.

```
$ source .venv/bin/activate # Reactivate the Python virtual environment as this is a different terminal
$ backend.ai session download ibnFmWim-session id_container
Downloading files: 3.58kbytes [00:00, 352kbytes/s]
✓ Downloaded to /*/client.
```

다운로드 한 키를 이용해서 SSH 접속을 다음과 같이 할 수 있습니다. 9922번에 로컬 프록시 포트를 띄웠으므로 접속 주소는 127.0.0.1, 포트는 9922 를 지정해야 합니다. 접속 사용자 계정은 work 로 사용하십시오.

```
$ ssh \
-o StrictHostKeyChecking=no \
-o UserKnownHostsFile=/dev/null \
-i ./id_container \
-p 9922 \
work@127.0.0.1
Warning: Permanently added '[127.0.0.1]:9922' (RSA) to the list of known hosts.
work@
```

마찬가지로 scp 명령을 사용하여 파일을 복사할 수 있습니다. 이 때, 연산 세션 내 마운트 된 폴더로 파일을 복사해야 연산 세션이 종료되었을 때에도 파일을 보존할 수 있습니다.

```
$ scp \
-o StrictHostKeyChecking=no \
```

(다음 페이지에 계속)

(이전 페이지에서 계속)

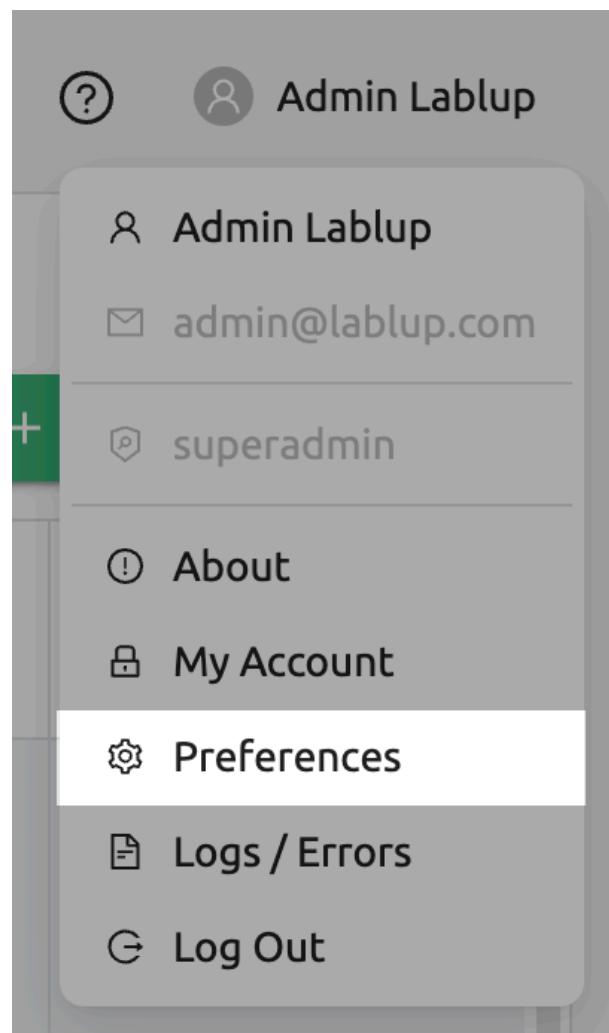
```
-o UserKnownHostsFile=/dev/null \
-i ./id_container \
-P 9922 \
test_file.xlsx work@127.0.0.1:/home/work/myfolder/
Warning: Permanently added '[127.0.0.1]:9922' (RSA) to the list of
known hosts.
test_file.xlsx
```

모든 작업이 끝난 경우에는 첫 번째 터미널에서 Ctrl-C 를 눌러 로컬 프록시 서비스를 종료합니다.

CHAPTER 19

사용자 설정 페이지

사용자 설정 페이지는 우측 상단 사람 아이콘을 클릭하면 나오는 Preferences 메뉴를 선택하여 접근할 수 있습니다. 여기서는, 사용자 설정 페이지의 각 항목 및 메뉴에 관해 간단히 설명합니다.



19.1 GENERAL 탭

The screenshot shows the 'General' tab in the Backend.AI WebUI settings. On the left, a sidebar lists sections: 'All (13)', 'Preferences (8)', 'Shell Environments (2)', and 'Experimental features (3)'. The main content area is titled 'Preferences' and contains several configuration groups:

- Desktop Notification**: An option to enable desktop notifications, with a note that it uses the OS built-in notification system.
- Use Compact Sidebar by default**: An option to compact the sidebar for more workspace, noting it becomes the initial UI at startup.
- Language**: Set the UI language to English.
- Automatic Update Check**: An option to notify when a new webui version is released, with a note about toggling the switch.
- Auto Logout**: An option to log out automatically if all logged-in pages are closed except session originated pages.
- My Keypair Information**: A section to see keypair information, with a 'Config' button.

At the top right of the main area are 'Display Only Changes' and 'Reset' buttons.

GENERAL 탭에는 많은 사용자 설정이 있습니다. 섹션 맨 위에 있는 검색 필드에서 검색하거나, 'Display Only Changes' 를 클릭해서 변경된 항목만 걸러낼 수 있습니다. 이전 설정으로 되돌리고 싶다면, 섹션 오른쪽 위에 있는 Reset 버튼을 클릭합니다.

19.1.1 데스크톱 알림

데스크톱 알림 기능을 사용 여부를 설정합니다. 브라우저 및 운영체제가 지원하는 경우, WebUI에서 또는 각종 하단 메시지가 데스크톱 알림 패널에도 같이 뜨게 됩니다. 첫 실행 시 운영체제 단에서 비활성화한 경우, 여기서 옵션을 켜더라도 데스크톱 메시지가 출력되지 않을 수 있습니다. 이 옵션 설정과 관계없이, WebUI 내부의 알림 기능은 그대로 동작합니다.

19.1.2 간결한 사이드바 기본 사용

이 옵션이 켜져 있으면 좌측 사이드바가 콤팩트 형태 (너비가 줄어든 형태)로 보이게 됩니다. 다만, 옵션을 적용한다고 해서 즉시 그 효과가 보이지는 않으며, 브라우저를 갱신할 때 적용됩니다. 페이지 갱신 없이 사이드바 형태를 즉시 변경하고 싶다면, 헤더 상단부의 가장 좌측 아이콘을 클릭하십시오.

19.1.3 언어 설정

UI에 출력되는 언어를 설정합니다. 현재 Backend.AI에서는 한국어, 영어를 포함해 다섯 개 이상의 언어를 지원하고 있습니다. 다만, 페이지 갱신 전에는 언어가 바뀌지 않는 사용자 인터페이스 항목이 있을 수 있습니다.

- OS Default: 운영체제 기본 설정된 언어를 사용합니다.
- English: 영어를 기본 언어로 설정합니다.
- Korean: 한국어를 기본 언어로 설정합니다.
- Brazilian Portuguese: 브라질 포르투갈어를 기본 언어로 설정합니다.
- Chinese (Simplified): 중국어(간체)를 기본 언어로 설정합니다.
- Chinese (Traditional): 중국어(번체)를 기본 언어로 설정합니다.
- French: 프랑스어를 기본 언어로 설정합니다.
- Finnish: 핀란드어를 기본 언어로 설정합니다.
- German: 독일어를 기본 언어로 설정합니다.
- Greek: 그리스어를 기본 언어로 설정합니다.
- Indonesian: 인도네시아어를 기본 언어로 설정합니다.
- Italian: 이탈리아어를 기본 언어로 설정합니다.
- Japanese: 일본어를 기본 언어로 설정합니다.
- Mongolian: 몽골어를 기본 언어로 설정합니다.
- Polish: 폴란드어를 기본 언어로 설정합니다.
- Portuguese: 포르투갈어를 기본 언어로 설정합니다.
- Russian: 러시아어를 기본 언어로 설정합니다.
- Spanish: 스페인어를 기본 언어로 설정합니다.
- Thai: 태국어를 기본 언어로 설정합니다.

- Turkish: 터키어를 기본 언어로 설정합니다.
- Vietnamese: 베트남어를 기본 언어로 설정합니다.

참고: 몇몇 언어에서는 __NOT_TRANSLATED__ 로 남아있는 항목이 있을 수 있습니다. Backend.AI는 오픈소스이므로 모국어 번역에 직접 참여하고 싶으신 경우에는 다음 저장소를 방문해주세요: <https://github.com/lablup/backend.ai-webui>.

19.1.4 자동 업데이트 체크

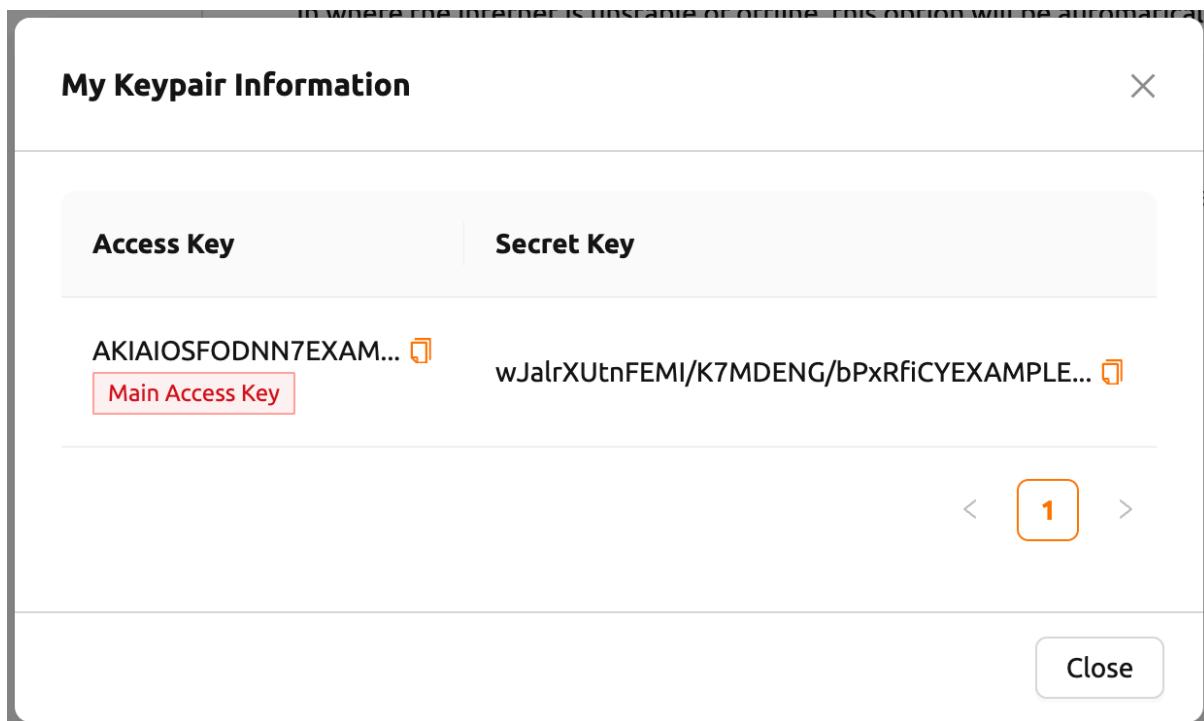
WebUI의 새 버전이 검색될 경우 알림 창을 띄웁니다. 이 기능은 인터넷 접속이 가능한 환경에서만 동작합니다.

19.1.5 자동 로그아웃

세션 내 앱을 실행하기 위해 생성된 페이지를 제외한 모든 Backend.AI WebUI 페이지가 닫힐 경우, 자동으로 로그아웃 됩니다. (Jupyter Notebook, Web Terminal 등의 앱을 접속하는 경우에는 로그아웃이 되지 않습니다.)

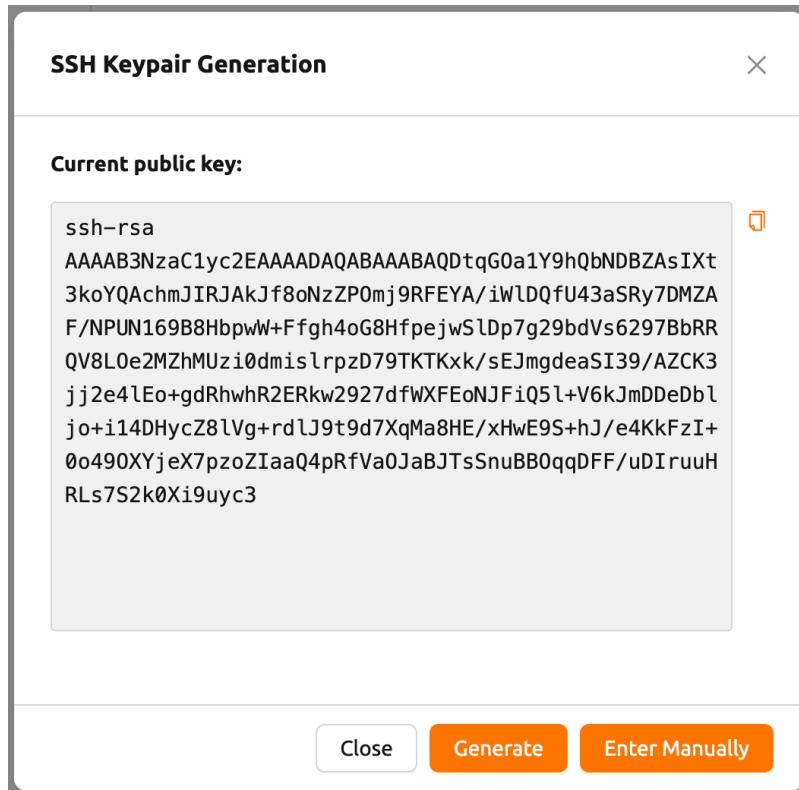
19.1.6 내 키페어 정보

모든 사용자는 하나 이상의 키페어를 가지고 있습니다. 아래의 구성 버튼을 클릭하면 액세스 및 비밀 키페어를 확인할 수 있습니다. 기본 액세스 키페어는 하나만 존재합니다.



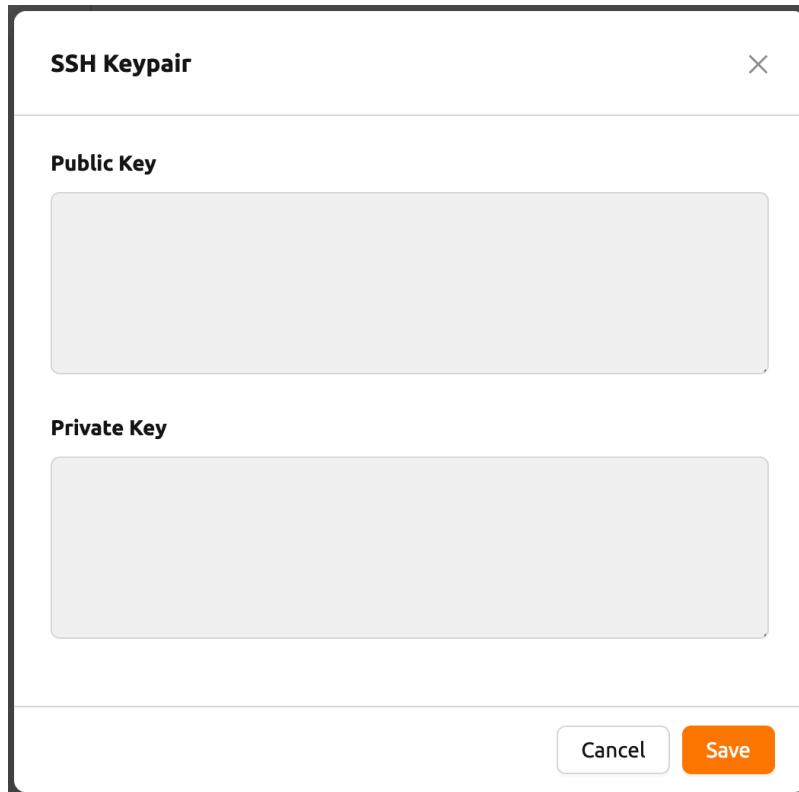
19.1.7 SSH 키페어 관리

연산 세션에 직접 SSH 로 접속할 때 필요한 SSH 키페어를 조회하고 생성하는 기능입니다. 우측 버튼을 클릭하면 다음과 같은ダイ얼로그가 뜹니다. 우측의 복사 버튼을 클릭하면 현재 존재하는 SSH 공개 키를 복사할 수 있습니다. 처음 Backend.AI 에 계정을 등록하게 되면, 자동으로 한 쌍의 키페어가 발급되게 됩니다. 갱신하고자 한다면 GENERATE 버튼을 클릭합니다. SSH 공개/비밀 키는 랜덤으로 생성되어 사용자 정보로 저장됩니다. 비밀 키는 생성 직후 따로 저장해두지 않으면 다시 확인할 수 없음에 주의하십시오.

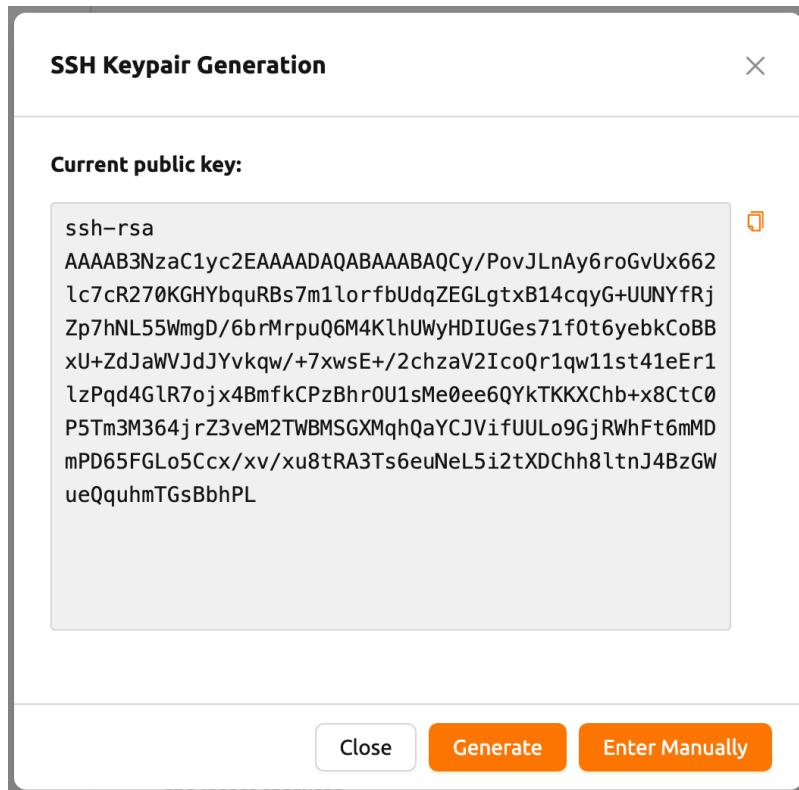


참고: Backend.AI는 OpenSSH에 기반한 SSH keypair를 사용합니다. Windows에서는 PPK 기반 키로 변환해야 할 수 있습니다.

22.09 버전부터 Backend.AI WebUI는 사설 저장소에 접근하는 작업과 같은 유연성을 제공하기 위해 사용자가 직접 SSH 키페어를 추가할 수 있도록 지원하고 있습니다. 직접 SSH 키페어를 추가 하려면, ENTER MANUALLY 버튼을 클릭하세요. 그다음, public, private 이라고 적혀있는 텍스트 입력 칸을 확인하실 수 있습니다.



대응하는 키를 입력하고, SAVE 버튼을 클릭하세요. 여러분이 등록한 키를 통해 손쉽게 Backend.AI 세션에 접근할 수 있습니다.



19.1.8 Classic 세션 런처

Backend.AI 24.90 버전 이후부터, 세션 런처(NEO)의 개선된 버전을 기본 세션 런처로 제공합니다. 이전의 대화식 세션 런처를 사용하고 싶다면, 이 옵션을 활성화하세요. Classic session launcher의 사용방법에 대해선 다음 [링크](#)를 참고하세요.

19.1.9 부트스트랩 스크립트 수정

연산 세션 시작 후 한 번만 스크립트를 실행하고자 할 경우, 여기에 그 내용을 작성해주십시오.

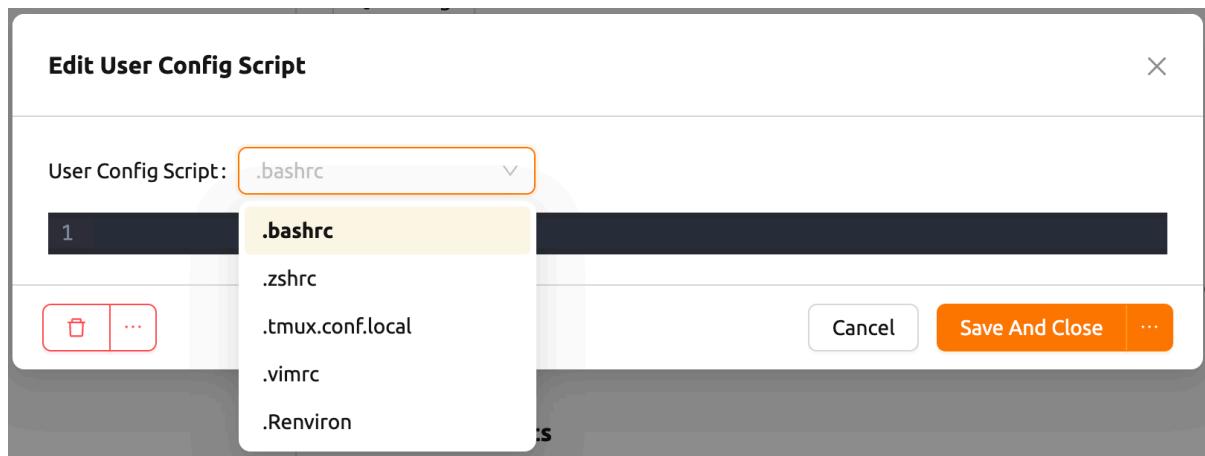


경고: 부트스트랩 스크립트 실행이 끝나기 전까지 세션은 PREPARING 상태에 머무르게 되고, 이 상태에서는 사용자가 연산 세션을 사용할 수 없습니다. 따라서, 오랫동안 실행해야 하는 작업이 있다면, 부트스트랩 스크립트에 작성하는 대신 연산 세션이 RUNNING 상태가 된 후 터미널 앱을 열고 실행하는 것이 나을 수 있습니다.

19.1.10 사용자 환경 스크립트 수정

연산 세션 생성 시 몇 가지 사용자 설정 스크립트를 작성하여 기본 설정을 대체할 수 있습니다. `.bashrc`, `.tmux.conf.local`, `.vimrc` 등과 같은 파일을 변경하여 사용자화할 수 있습니다. 이 스크립트는 사용자 별로 저장되며, 일정한 자동화 작업이 필요할 때 활용할 수 있습니다. 예를 들어, `.bashrc` 스크립트를 수정하여 원하는 명령 alias를 등록하거나 특정 파일을 특정 위치에 항상 다운로드하도록 지정해둘 수 있습니다.

상단의 드롭다운 메뉴를 활용해서 작성할 스크립트의 종류를 선택한 후 내용을 작성하십시오. 작성이 완료되면 SAVE 또는 SAVE AND CLOSE 버튼을 클릭해서 스크립트를 저장할 수 있습니다. DELETE 버튼을 클릭하면 해당 스크립트를 삭제할 수 있습니다.



19.1.11 실험적 기능

실험적 기능들이 정식으로 출시되기 전에 활성화하거나 비활성화할 수 있습니다.

Experimental features

Try out new and experimental features before they are officially released. These features may change or be removed in future updates.

NEO Session list

Enabled

AI Agents

Enabled

Dashboard

Use Dashboard page instead of Summary page.

19.2 LOGS 탭

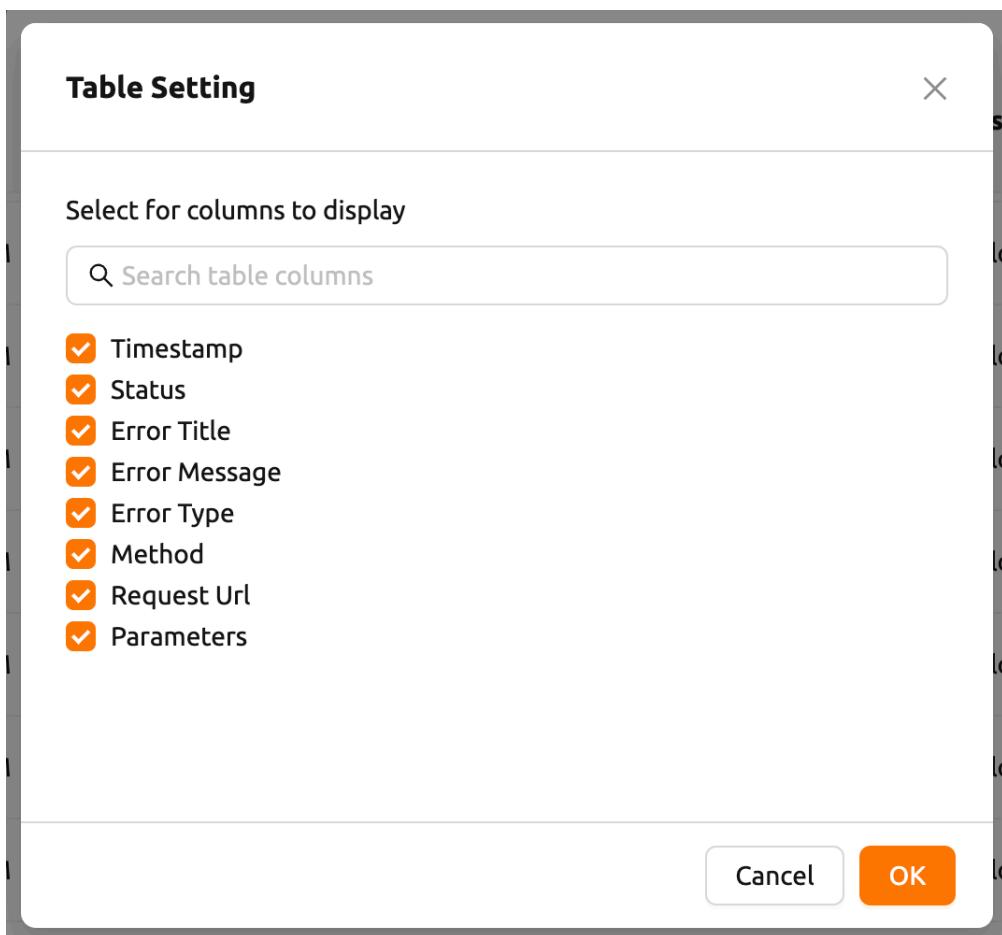
클라이언트 측에서 기록된 각종 로그의 상세 정보를 출력합니다. 요청 오류가 발생했을 때 자세한 내용을 확인하고 싶을 때 이 페이지에 방문할 수 있습니다. 우측 상단의 버튼을 이용해서 로그를 검색하거나 여러 메시지를 필터링하고, 로그를 새로고침하거나 지울 수 있습니다.

The screenshot shows the 'Logs' tab of the Backend.AI WebUI. At the top, there are tabs for 'General' and 'Logs', with 'Logs' being active. Below the tabs is a search bar labeled 'Search logs' and several buttons: 'Display only error' (unchecked), 'Refresh' (with a circular arrow icon), and 'Clear Logs' (with a trash bin icon). The main area is titled 'Log Messages' and indicates '(Up to 3000 recent logs)'. A table follows, with columns: Timestamp, Status, Error Title, Error Message, Error Type, Method, Request Url, and Parameters. The table contains 10 log entries from April 10, 2025, at various times between 6:37:35 PM and 6:39:06 PM. Most entries show a POST method to 'http://local.backend.ai:8090/func/admin/gql' with parameters containing a query. One entry shows a GET method to 'http://local.backend.ai:8090/func/user-config/dotfiles'. The last column, 'Parameters', shows a truncated JSON object for each entry. At the bottom right of the table is a pagination control with page numbers 1 through 201, where page 1 is highlighted with an orange border.

Timestamp	Status	Error Title	Error Message	Error Type	Method	Request Url	Parameters
Apr 10, 2025 6:39:06 PM	200 OK	-	-	-	POST	http://local.backend.ai:8090/func/admin/gql	{"query": "qu...}
Apr 10, 2025 6:39:06 PM	200 OK	-	-	-	POST	http://local.backend.ai:8090/func/admin/gql	{"query": "qu...}
Apr 10, 2025 6:38:51 PM	200 OK	-	-	-	POST	http://local.backend.ai:8090/func/admin/gql	{"query": "qu...}
Apr 10, 2025 6:38:44 PM	200 OK	-	-	-	GET	http://local.backend.ai:8090/func/user-config/dotfiles	-
Apr 10, 2025 6:38:36 PM	200 OK	-	-	-	POST	http://local.backend.ai:8090/func/admin/gql	{"query": "qu...}
Apr 10, 2025 6:38:21 PM	200 OK	-	-	-	POST	http://local.backend.ai:8090/func/admin/gql	{"query": "qu...}
Apr 10, 2025 6:38:06 PM	200 OK	-	-	-	POST	http://local.backend.ai:8090/func/admin/gql	{"query": "qu...}
Apr 10, 2025 6:37:51 PM	200 OK	-	-	-	POST	http://local.backend.ai:8090/func/admin/gql	{"query": "qu...}
Apr 10, 2025 6:37:35 PM	200 OK	-	-	-	POST	http://local.backend.ai:8090/func/admin/gql	{"query": "qu...}

참고: 로그인된 페이지가 하나만 존재할 경우, REFRESH 버튼을 클릭하면 제대로 작동하지 않는 것처럼 보일 수 있습니다. 로그 페이지는 서버에 대한 요청과 서버의 응답을 모아둔 것이며, 현재 페이지가 로그 페이지인 경우 명시적으로 페이지를 새로 고침하는 것 외에는 서버에 요청을 보내지 않습니다. 로그가 제대로 쌓이는지 확인하려면 다른 페이지를 열고 REFRESH 버튼을 클릭하십시오.

특정 열을 숨기거나 보이게 하려면, 테이블 우측 하단의 기어 아이콘을 클릭하십시오. 그러면 아래와 같은 다이얼로그가 나타나며, 보고 싶은 열을 선택할 수 있습니다.



CHAPTER 20

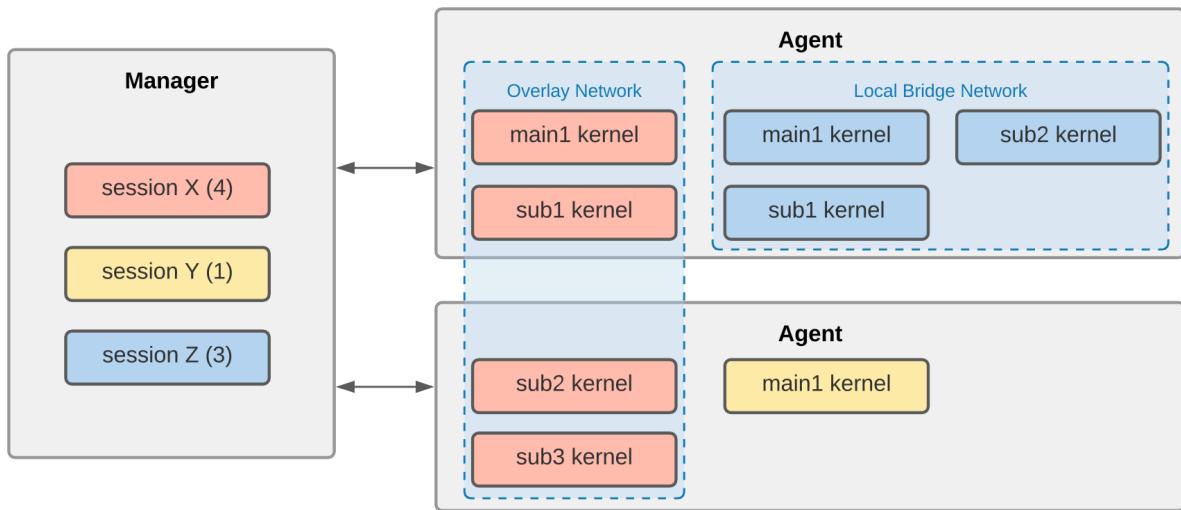
Backend.AI 클러스터 연산 세션

참고: 클러스터 연산 세션 기능은 Backend.AI 서버 20.09 버전 이상부터 지원합니다.

20.1 Backend.AI 클러스터 연산 세션 개괄

Backend.AI에서는 분산 연산 및 학습 작업을 지원하기 위해 클러스터 연산 세션을 지원하고 있습니다. 클러스터 세션은 여러 개의 컨테이너로 구성되며, 각 컨테이너는 여러 Agent 노드에 걸쳐 생성됩니다. 클러스터 세션으로 묶인 컨테이너는 동적으로 생성된 사설 네트워크를 통해 자동으로 연결되며, 임시 도메인 이름(`main1`, `sub1`, `sub2` 등)이 부여되어 간단하게 SSH 접속 등의 네트워킹 작업이 가능합니다. 컨테이너 간 SSH 접속을 위해 필요한 비밀 키 및 각종 설정은 모두 자동으로 생성됩니다.

Backend.AI 클러스터 세션의 상세 특징은 다음을 참고 하십시오.



- 클러스터 세션을 구성하는 컨테이너는 한 자원 그룹에 속한 하나 이상의 Agent 노드에 걸쳐 생성 됩니다.
- 클러스터 세션은 하나의 메인 컨테이너(main1)와 하나 이상의 서브 컨테이너(subX)로 구성됩니다.
- 모든 구성 컨테이너는 동일한 자원을 할당하여 생성 됩니다. 위 그림에서 session X의 네 개 컨테이너는 모두 동일한 양의 자원으로 생성 됩니다.
- 모든 구성 컨테이너는 연산 세션 생성 시 지정한 데이터 폴더를 동일하게 마운트 합니다.
- 모든 구성 컨테이너는 사설 네트워크로 묶입니다.
 - 메인 컨테이너의 이름은 main1 이 됩니다.
 - 서브 컨테이너의 이름은 sub1, sub2, ... 와 같은 순서로 지어집니다.
 - 한 클러스터 세션을 구성하는 컨테이너 사이에는 별도의 방화벽이 없습니다.
 - 사용자는 메인 컨테이너에 직접 접속할 수 있으며, 서브 컨테이너는 메인 컨테이너에 먼저 접속한 후 메인 컨테이너 내에서 접속할 수 있습니다.
- 두 가지 모드/형태의 클러스터 세션이 있습니다.
 - 단일 노드 클러스터 세션: 하나의 Agent 노드에서 두 개 이상의 컨테이너를 묶어 구성한 클러스터 세션입니다. 위 그림에서는 session Z가 이에 해당하며 로컬 브릿지 네트워크로 묶입니다.
 - 멀티 노드 클러스터 세션: 두 개 이상의 컨테이너를 서로 다른 Agent 노드에 배치해서 구성한 클러스터 세션입니다. 위 그림에서는 session X가 이에 해당하며, 오버레이 네트워크로 묶입니다.
 - 하나의 컨테이너를 가지는 연산 세션은 클러스터 세션이 아닌 일반 연산 세션으로 분류 합니다. 위 그림에서는 session Y가 이에 해당합니다.

- 다음과 같은 경우에는 단일 노드 클러스터 세션이 생성됩니다.
 - 연산 세션 생성 시 Cluster mode에 “Single Node”를 선택한 경우. 모든 컨테이너를 동시에 생성할 수 있는 자원을 가진 단일 Agent가 없으면, 해당 세션 생성 작업은 대기 (PENDING) 상태로 머무르게 됩니다.
 - 연산 세션 생성 시 Cluster mode에 “Multi Node”를 선택했으나, 모든 컨테이너를 동시에 생성할 수 있는 자원을 가진 단일 Agent가 있을 경우, 그 Agent에 모든 컨테이너를 배치합니다. 이는 가능한 외부 네트워크를 배제하여 연산 세션을 구성하여 네트워크 지연시간을 줄기 위함입니다.

클러스터 세션의 각 컨테이너에는 다음과 같은 환경 변수가 지정됩니다. 클러스터 및 현재 접속된 컨테이너 정보를 확인하기 위해서 참고할 수 있습니다.

- BACKENDAI_CLUSTER_HOST: 현재 컨테이너의 이름 (ex. main1)
- BACKENDAI_CLUSTER_HOSTS: 현재 클러스터 세션에 속한 모든 컨테이너의 이름 (ex. main1, sub1, sub2)
- BACKENDAI_CLUSTER_IDX: 현재 컨테이너의 숫자 인덱스 (ex. 1)
- BACKENDAI_CLUSTER_MODE: 클러스터 세션의 모드/형태 (ex. single-node)
- BACKENDAI_CLUSTER_ROLE: 현재 컨테이너의 종류 (ex. main)
- BACKENDAI_CLUSTER_SIZE: 현재 클러스터 세션에 속한 컨테이너의 총 수 (ex. 4)
- BACKENDAI_KERNEL_ID: 현재 컨테이너의 ID (ex. 3614fdf3-0e04-…)
- BACKENDAI_SESSION_ID: 현재 컨테이너가 속한 클러스터 세션의 ID (ex. 3614fdf3-0e04-…). 메인 컨테이너의 BACKENDAI_KERNEL_ID 는 BACKENDAI_SESSION_ID 와 같습니다.

20.2 Backend.AI 클러스터 연산 세션의 사용

이번 절에서는 실제로 사용자 GUI를 통해 클러스터 연산 세션을 생성하고 사용하는 법에 관해 알아보겠습니다.

Sessions 페이지에서 연산 세션 생성 디아일로그를 띄운 후 일반 연산 세션을 생성하는 것과 같은 방식으로 설정합니다. 이 때 설정한 자원의 양은 하나의 컨테이너에 할당되는 양입니다. 예를 들어, CPU를 4개로 설정한 경우 클러스터 세션을 구성하는 각 컨테이너에 4 코어 CPU가 할당됩니다. 클러스터 연산 세션 전체에 할당되는 자원 양이 아니라는 점에 유의 하십시오. 클러스터 연산 세션을 생성하기 위해서는 여기서 설정한 자원양의 N배에 해당하는 서버 자원이 필요합니다 (N은 클러스터 크기). 데이터 보존을 위해 데이터 폴더를 마운트 하는 것도 잊지 마십시오.

Resource allocation

Resource Group
default

Resource Presets
Custom allocation

CPU ⓘ
4 Core 1 256

Memory ⓘ
MEM 8 GiB 576m 64g
Set minimum shared memory

AI Accelerator ⓘ (optional)
1.0 0 8

Sessions ⓘ
1 # 1 3

Select Agent ⓘ
Auto Select

Cluster mode ⓘ
 Single Node Multi Node

Cluster size
3 Container 1 10

아래쪽에 있는 “Cluster mode” 필드에서 원하는 클러스터 생성 형태를 결정할 수 있습니다.

- Single Node: 모든 구성 컨테이너는 하나의 Agent 노드에 생성 됩니다.
- Multi Node: 구성 컨테이너는 자원 그룹 내에 있는 여러 Agent 노드에 걸쳐 생성됩니다. 다만, 모든 컨테이너가 하나의 Agent 노드에 생성될 수 있는 경우에는, 하나의 Agent 노드에 우선 생성합니다. 이는 컨테이너 간 통신 과정에서 네트워크 지연 시간을 최대한 줄이기 위함입니다.

그 아래에 있는 “Cluster size”를 설정합니다. 3으로 설정할 경우, 메인 컨테이너를 포함해서 총 세 개의 컨테이너가 생성됩니다. 이 세 개의 컨테이너는 사실 네트워크로 묶여 하나의 연산 세션을 구성합니다.

LAUNCH 버튼을 클릭하여 연산 세션 생성 요청을 보내고 잠시 기다리면 클러스터 세션이 생성됩니다. 세션 생성 이후에, 세션 상세 페이지에서 생성된 3개의 컨테이너를 확인할 수 있습니다.

Kernels

Kernel ID	Role	Status	Container ID
66a3f808-fa92-47ef-a31e-a1201cd71e2b	sub	RUNNING	4fd8aca423f8cc4823da917e09...
c426e322-cfa0-45cd-9927-16fe7bebe26b	sub	RUNNING	9e65acac295e5d3076c9b8c6...
caa0ab7c-e2f3-4337-b555-52b90fadd67a	main	RUNNING	19cae171dcf9ba58a63d8bfb...

생성한 연산 세션에서 터미널 앱을 열어 보겠습니다. 환경 변수를 조회해보면 위 절에서 설명한 BACKENDAI_CLUSTER_* 변수가 설정된 것을 확인할 수 있습니다. 각 환경 변수의 의미와 값을 위 설명과 대조해 보십시오.

```
work@main1[5VIedkFc-session]:~$ export
declare -x BACKENDAI_CLUSTER_HOST="main1"
declare -x BACKENDAI_CLUSTER_HOSTS="sub2,sub1,main1"
declare -x BACKENDAI_CLUSTER_IDX="1"
declare -x BACKENDAI_CLUSTER_LOCAL_RANK="0"
declare -x BACKENDAI_CLUSTER_REPLICAS="main:1,sub:2"
declare -x BACKENDAI_CLUSTER_ROLE="main"
declare -x BACKENDAI_CLUSTER_SIZE="3"
```

sub1 컨테이너로 SSH 접속을 할 수도 있습니다. 별도의 SSH 설정 없이 ssh sub1이라는 명령만 내리면 끝입니다. work@ 뒤의 호스트 이름이 변한 것을 볼 수 있습니다. sub1 컨테이너의 쉘로 접속한 것입니다.

```

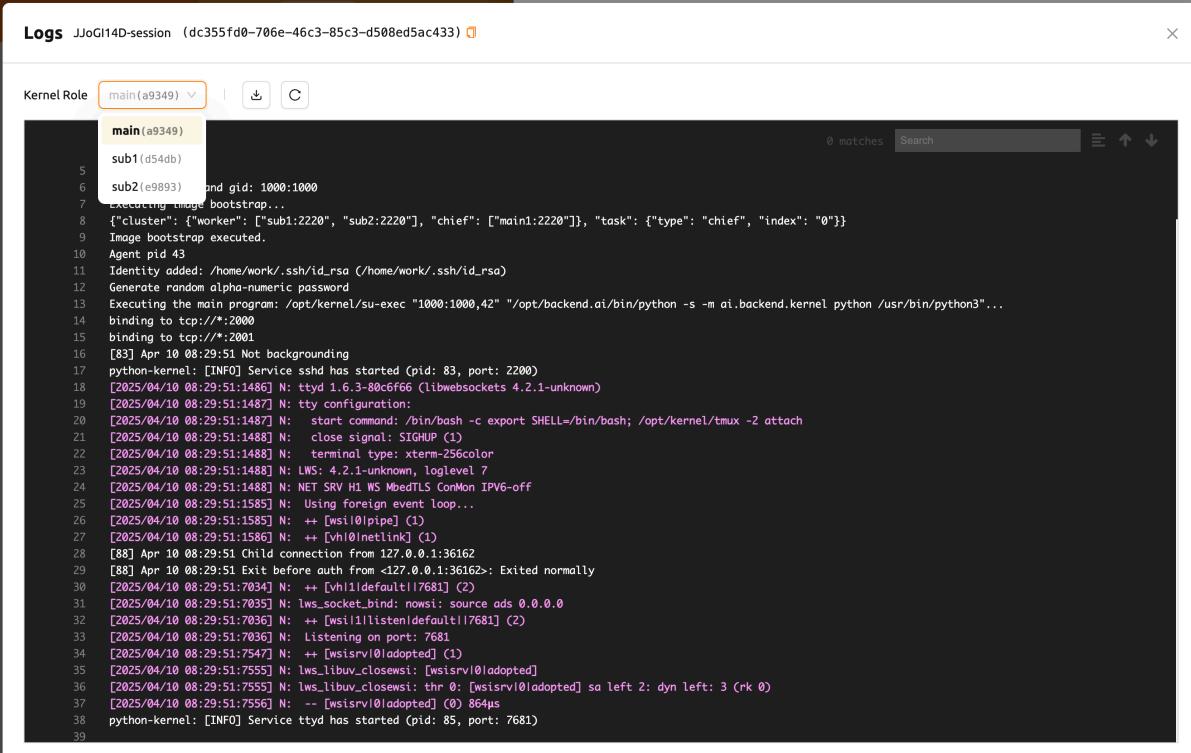
work@main1[5VIedkFc-session]:~$ ssh sub1
Warning: Permanently added '[sub1]:2200, [172.27.0.4]:2200' (RSA) to the list of known hosts.
work@sub1[5VIedkFc-session]:~$ ls -al
total 64
drwxr-xr-x 8 work work 4096 Mar 28 09:42 .
drwxr-xr-x 1 root root 4096 Mar 28 09:42 ..
-rw-rw-r-- 1 work work 147 Mar 28 09:42 .bash_profile
-rw-rw-r-- 1 work work 249 Mar 28 09:42 .bashrc
drwxr-xr-x 3 work work 4096 Mar 28 09:42 .cache
drwxr-xr-x 3 work work 4096 Mar 28 09:42 .config
drwxr-xr-x 5 work work 4096 Mar 28 09:42 .ipython
drwxrwxr-x 3 work work 4096 Mar 28 09:42 .jupyter
drwxr-xr-x 3 work work 4096 Mar 28 09:42 .local
-rw-r--r-- 1 root root 56 Mar 28 09:42 .password
drwx----- 2 work work 4096 Mar 28 09:42 .ssh
-rw-rw-r-- 1 work work 2195 Mar 28 09:42 .tmux.conf
-rw-rw-r-- 1 work work 501 Mar 28 09:42 .vimrc
-rw-rw-r-- 1 work work 472 Mar 28 09:42 .zshrc
-rw-rw-r-- 1 work work 540 Oct 26 14:44 DO_NOT_STORE_PERF
-rw----- 1 work work 1675 Mar 28 09:42 id_container
work@sub1[5VIedkFc-session]:~$ exit
logout
Connection to sub1 closed.

```

이런 방식으로 Backend.AI 에서는 클러스터 연산 세션을 손쉽게 생성할 수 있습니다. 클러스터 연산 세션을 통해 분산 학습 및 연산을 실행하기 위해서는, TensorFlow/PyTorch 등 ML 라이브러리에서 제공하는 분산 학습용 모듈이나 Horovod, NNI, MLFlow 등과 같은 별도의 지원 소프트웨어가 필요하고, 해당 소프트웨어를 활용할 수 있는 방식으로 코드를 주의깊게 작성해야 합니다. Backend.AI에서는 분산 학습에 필요한 소프트웨어를 포함하는 커널 이미지를 제공하고 있으므로, 해당 이미지를 사용해서 멋진 분산 학습 알고리즘을 만들어 보실 수 있습니다.

20.3 컨테이너 별로그 확인하기

24.03 부터는 컨테이너 로그 다이얼로그에서 각 컨테이너별로그를 확인할 수 있습니다. 'main' 컨테이너 뿐만 아니라 'sub' 컨테이너에서 어떤 일이 일어나고 있는지 더 쉽게 확인할 수 있습니다.



The screenshot shows the 'Logs' section of the Backend.AI WebUI. The title bar indicates it's for session (dc355fd0-706e-46c3-85c3-d508ed5ac433). The main area displays log entries from a container named 'main(a9349)'. The log output is as follows:

```

Logs JJoGI14D-session (dc355fd0-706e-46c3-85c3-d508ed5ac433) X

Kernel Role: main(a9349) | ↻ C
main(a9349)
sub1(d54db)
sub2(e9893) and gid: 1000:1000
Executing image bootstrap...
{"cluster": ["worker": ["sub1:2220", "sub2:2220"], "chief": ["main1:2220"]], "task": {"type": "chief", "index": "0"}}
Image bootstrap executed.
Agent pid 43
Identity added: /home/work/.ssh/id_rsa (/home/work/.ssh/id_rsa)
Generate random alpha-numeric password
Executing the main program: /opt/kernel/su-exec "1000:1000,42" "/opt/backend.ai/bin/python -s -m ai.backend.kernel python /usr/bin/python3"...
binding to tcp://*:2000
binding to tcp://*:2001
[83] Apr 10 08:29:51 Not backgrounding
python-kernel: [INFO] Service sshd has started (pid: 83, port: 2200)
[2025/04/10 08:29:51:1486] N: tttyd 1.6.3-80c6f66 (libwebsockets 4.2.1-unknown)
[2025/04/10 08:29:51:1487] N: tty configuration:
[2025/04/10 08:29:51:1487] N: start command: /bin/bash -c export SHELL=/bin/bash; /opt/kernel/tmux -2 attach
[2025/04/10 08:29:51:1488] N: close signal: SIGHUP (1)
[2025/04/10 08:29:51:1488] N: terminal type: xterm-256color
[2025/04/10 08:29:51:1488] N: LWS: 4.2.1-unknown, loglevel 7
[2025/04/10 08:29:51:1488] N: NET SRV H1 WS MbedTLS ConMon IPV6-off
[2025/04/10 08:29:51:1585] N: Using foreign event loop...
[2025/04/10 08:29:51:1585] N: ++ [wsl0|pipe] (1)
[2025/04/10 08:29:51:1586] N: ++ [vh10|netlink] (1)
[88] Apr 10 08:29:51 Child connection from 127.0.0.1:36162
[88] Apr 10 08:29:51 Exit before auth from <127.0.0.1:36162>: Exited normally
[2025/04/10 08:29:51:7034] N: ++ [vh10|default||7681] (2)
[2025/04/10 08:29:51:7035] N: lws_socket_bind: nowstl: source ads 0.0.0.0
[2025/04/10 08:29:51:7036] N: ++ [wsl1|listen|default||7681] (2)
[33] [2025/04/10 08:29:51:7036] N: Listening on port: 7681
[34] [2025/04/10 08:29:51:7547] N: ++ [wsisrv|0|adopted] (1)
[35] [2025/04/10 08:29:51:7555] N: lws_libuv_closews: [wsisrv|0|adopted]
[36] [2025/04/10 08:29:51:7555] N: lws_libuv_closews: th 0: [wsisrv|0|adopted] sa left 2: dyn left: 3 (rk 0)
[37] [2025/04/10 08:29:51:7556] N: -- [wsisrv|0|adopted] (0) 864μs
python-kernel: [INFO] Service tttyd has started (pid: 85, port: 7681)

```

CHAPTER 21

관리자 기능

관리자 계정으로 로그인 하면 왼쪽 사이드바에 Administration 메뉴가 추가로 보입니다. Backend.AI 에 등록된 사용자 정보는 Users 탭에서 볼 수 있습니다. Domain admin 의 경우 도메인에 속한 사용자 정보만 확인할 수 있고, super admin 은 전체 사용자 정보를 조회할 수 있습니다. 사용자의 생성 및 비활성화는 super admin 만 할 수 있습니다.

사용자 ID(이메일), username, 메인 access key 는 컬럼 헤더의 검색창에 텍스트를 입력하여 조회 결과를 필터링할 수도 있습니다.

The screenshot shows the 'Users' section of the Backend.AI WebUI. At the top, there are tabs for 'Users' (selected) and 'Credentials'. Below the tabs, there are buttons for 'Active' (selected), 'Inactive', and a dropdown for 'User ID'. A search bar and a clear button ('C') are also present. On the right, there is a '+ Create User' button. The main area is a table with columns: 'User ID', 'Name', 'Role', 'Description', 'Created At', and 'Control'. The 'Control' column contains three icons: a person icon, a gear icon, and a delete icon. The table lists eight users:

User ID	Name	Role	Description	Created At	Control
test-user@lblup.com	testuser	user	testuser's account	Feb 20, 2025 6:11 PM	
test2@lblup.com	test2	user	test2's Account	Jan 14, 2025 11:28 AM	
test1@lblup.com	test1	user	test1's Account	Jan 14, 2025 11:28 AM	
user2@lblup.com	user2	user	Lablup's User Account, Number 2	Dec 5, 2024 2:38 PM	
monitor@lblup.com	monitor	monitor	Lablup's Monitor Account	Dec 5, 2024 2:38 PM	
domain-admin@lblup.com	domain-admin	admin	Lablup's Default Domain Admin Account	Dec 5, 2024 2:38 PM	
user@lblup.com	user	user	Lablup's User Account	Dec 5, 2024 2:38 PM	
admin@lblup.com	admin	superadmin	Lablup's Admin Account	Dec 5, 2024 2:38 PM	

At the bottom, there is a pagination indicator '1-8 of 8 users' with arrows, a page number '1' in a highlighted box, and a '20 / page' dropdown.

참고: 플러그인 설정에 따라 이중 인증 사용(2FA Enabled) 열이 안보일 수 있습니다. 이 경우에는 시스템 관리자에게 문의 하십시오.

21.1 새로운 사용자 생성 및 정보 갱신

사용자는 CREATE USER 버튼을 클릭하여 생성할 수 있습니다. 이 때 비밀번호는 8 자 이상, 알파벳/특수문자/숫자를 1 개 이상 포함해야 합니다. E-Mail 과 Username 필드에는 최대 64 자까지 입력할 수 있습니다.

참고: 이미 같은 이메일이나 사용자 이름을 가지는 사용자가 존재한다면 사용자를 생성할 수 없습니다. 다른 이메일과 사용자 이름을 사용해 보십시오.

Create User

* User ID: test-user@lablup.com

User Name: testuser

Full Name: testuser

* Password:  At least 1 alphabet, 1 number and 1 special character is required with at least 8 chars.

* Confirm Password: 

Description: testuser's account

User Status: Active 

Require password change?: 

Enable sudo session: 

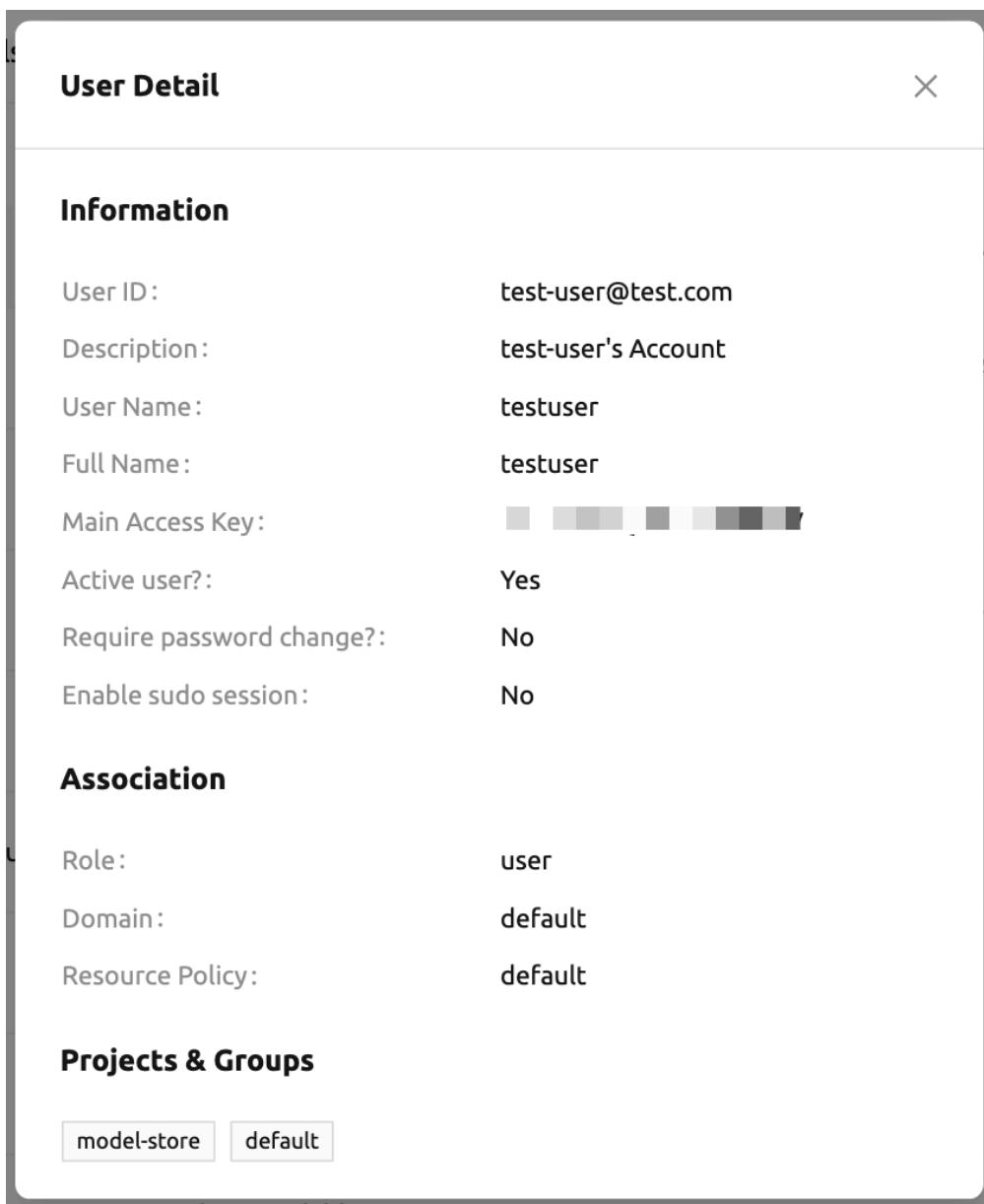
* Resource Policy: default 

Cancel **OK**

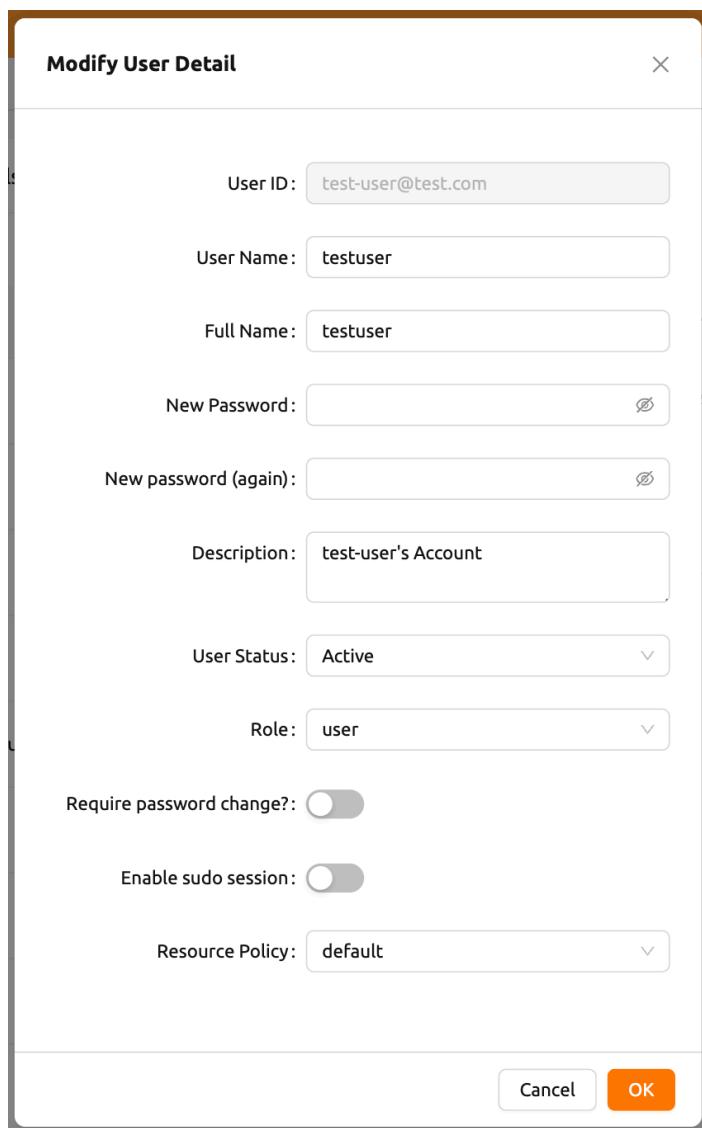
사용자가 생성된 것을 확인합니다.

User ID	Name	Role	Description	Created At	Control
test-user@lablup.com	testuser	user	testuser's account	Feb 20, 2025 6:11 PM	  

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

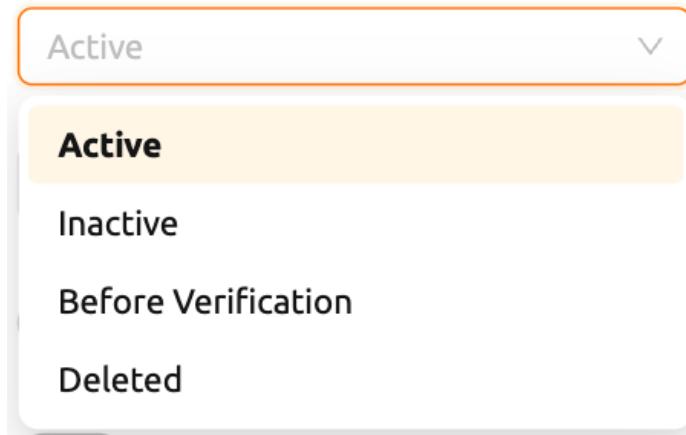


Controls 열의 톱니바퀴로 된 '설정' 버튼을 클릭하면 이미 존재하는 사용자의 정보를 업데이트 할 수 있습니다. 사용자의 이름, 비밀번호, 활성/비활성 여부 등을 변경하여 저장할 수 있습니다. User ID는 변경되지 않습니다.



이ダイアログ ハンドルの各項目は以下の機能を持っています。

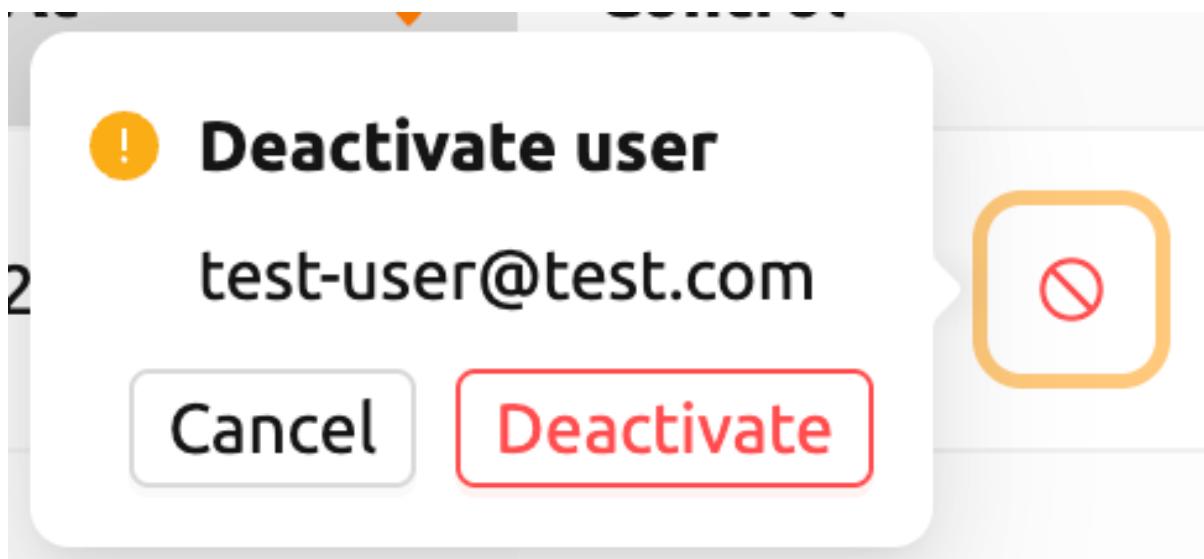
- User Status: ユーザーの状態を示します。非活動状態(Inactive)のユーザーはログインが不可能です。Before Verification 状態の場合、該当アカウントを有効化するため追加手順が必要な場合があります。これは、ユーザーのメールアドレスによる検証または管理者による承認手順などです。非活動ユーザーは Inactive タブで別途表示されます。



- **Require password change?**: 관리자가 일괄적으로 사용자를 생성하면서 비밀번호를 임의 지정했을 경우, 이 필드를 On 으로 지정하여 구분할 수 있습니다. 비밀번호 변경이 필요하다는 일종의 기록 플래그로, 사용자 화면 상단에 비밀번호 변경을 안내하는 메시지가 출력되긴 하지만 실 사용에는 아무런 영향을 미치지 않습니다.
- **Enable sudo session**: 사용자가 연산 세션에서 sudo 를 사용할 수 있도록 허용합니다. 이는 사용자가 패키지를 설치하거나 루트 권한이 필요한 명령을 실행할 때 유용합니다. 그러나 모든 사용자에게 이 옵션을 활성화 하는 것은 보안 문제를 일으킬 수 있기 때문에 권장되지 않습니다.
- **2FA Enabled**: 사용자가 이중 인증을 사용하는지를 나타내는 플래그입니다. 이중 인증을 사용할 경우, 사용자는 로그인 시 OTP 코드를 추가로 입력해야 합니다. 관리자는 다른 사용자의 이중 인증 해제만 가능합니다.
- **Resource Policy**: Backend.AI 24.09 버전 이후부터 사용자 자원 정책을 선택할 수 있습니다. 사용자 자원 정책에 대한 자세한 설명은 [사용자 자원 정책](#) 을 참고하세요.

21.2 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 삭제 아이콘을 클릭합니다. 확인을 위한 팝오버가 뜨는데, Deactivate 버튼을 클릭하면 사용자를 비활성화 할 수 있습니다.



사용자를 다시 활성화 하려면 Users - Inactive 탭에 방문하여 사용자 편집 버튼(톱니 아이콘)을 클릭한 후 사용자의 상태를 Active로 변경하십시오.

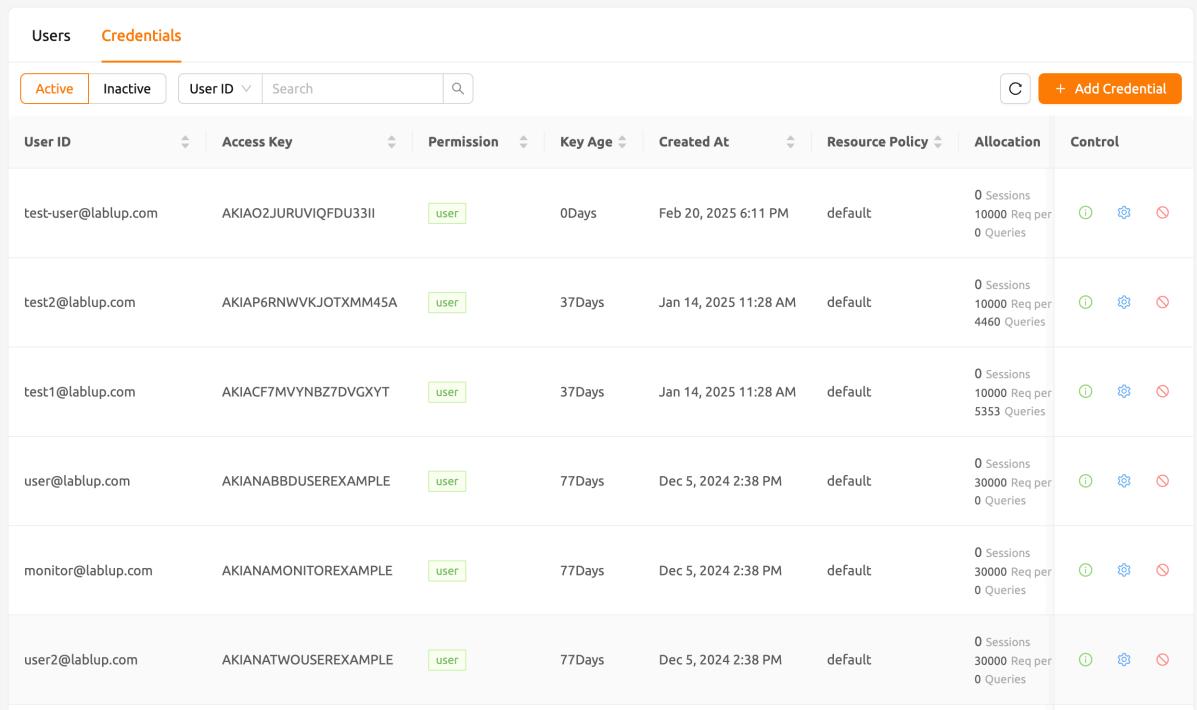
경고: 사용자 계정을 비활성화 하면 그 사용자의 모든 자격증명도 따라서 비활성 상태로 바뀌게 됩니다. 하지만, 사용자를 다시 활성화 해도 비활성 상태의 자격증명을 다시 활성화 하지는 않습니다. 사용자는 여러 개의 자격증명을 가질 수 있어 어떤 키페어를 활성화 할지 일관된 정책을 정하기 어렵기 때문입니다.

21.3 사용자의 키페어 관리

사용자 계정에는 보통 하나 이상의 키페어가 할당되어 있습니다. 키페어는 사용자 로그인 후 Backend.AI 서버로 요청을 보낼 때 인증을 위해 사용됩니다. 로그인을 위해서는 사용자 이메일 및 비밀번호를 통한 인증이 필요하지만, 사용자가 서버로 보내는 매 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. 새 사용자를 생성하면 자동으로 키페어가 하나 만들어집니다. 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.

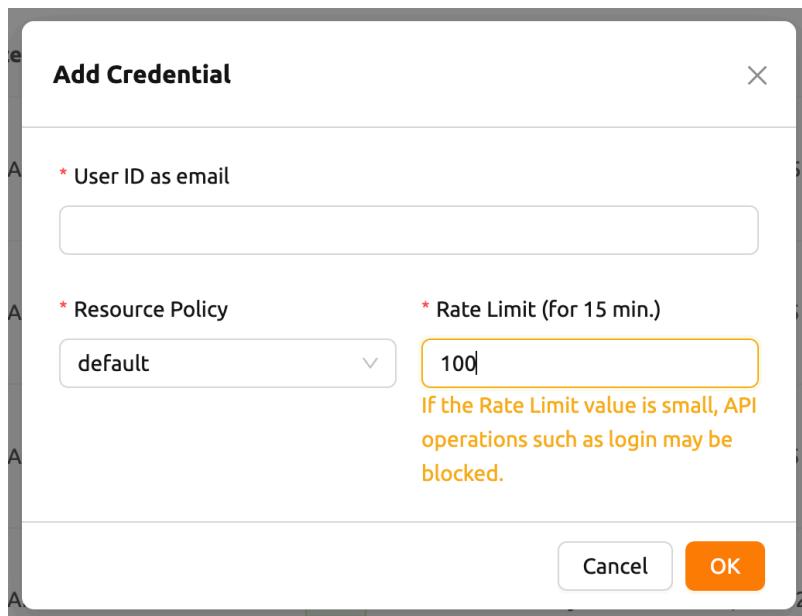


The screenshot shows the 'Credentials' tab of the Backend.AI WebUI. At the top, there are tabs for 'Users' and 'Credentials', with 'Credentials' being active. Below the tabs is a toolbar with buttons for 'Active' (selected), 'Inactive', a dropdown for 'User ID', a search bar, and a 'C' icon. To the right of the search bar is a '+ Add Credential' button. The main area is a table with columns: 'User ID', 'Access Key', 'Permission', 'Key Age', 'Created At', 'Resource Policy', 'Allocation', and 'Control'. The table lists six entries:

User ID	Access Key	Permission	Key Age	Created At	Resource Policy	Allocation	Control
test-user@lblup.com	AKIAO2JURUVIQFDU33II	user	0Days	Feb 20, 2025 6:11 PM	default	0 Sessions 10000 Req per 0 Queries	Info Settings Delete
test2@lblup.com	AKIAP6RNWVKJOTXMM45A	user	37Days	Jan 14, 2025 11:28 AM	default	0 Sessions 10000 Req per 4460 Queries	Info Settings Delete
test1@lblup.com	AKIACF7MVYNBZ7DVGXYT	user	37Days	Jan 14, 2025 11:28 AM	default	0 Sessions 10000 Req per 5353 Queries	Info Settings Delete
user@lblup.com	AKIANABBDUSEREXAMPLE	user	77Days	Dec 5, 2024 2:38 PM	default	0 Sessions 30000 Req per 0 Queries	Info Settings Delete
monitor@lblup.com	AKIANAMONITOREEXAMPLE	user	77Days	Dec 5, 2024 2:38 PM	default	0 Sessions 30000 Req per 0 Queries	Info Settings Delete
user2@lblup.com	AKIANATWouserexample	user	77Days	Dec 5, 2024 2:38 PM	default	0 Sessions 30000 Req per 0 Queries	Info Settings Delete

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 key 와 secret key 를 직접 명시적으로 적어줄 수도 있습니다.

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



21.4 프로젝트 Storage 폴더를 다른 사용자와 공유하기

Backend.AI 는 개인용 Storage 폴더 외에 프로젝트 전용 Storage 폴더를 제공합니다. 프로젝트 Storage 폴더는 특정 사용자가 아닌 특정 프로젝트에 속하는 폴더이며, 해당 프로젝트에 속한 모든 사용자가 접근할 수 있습니다.

참고: 프로젝트 폴더는 오직 관리자만 생성할 수 있습니다. 일반 사용자는 관리자가 생성한 프로젝트 폴더 내용을 공유받아 접근할 수 있을 뿐입니다. 단, 시스템 설정에 따라 프로젝트 폴더 생성이 허용되지 않는 경우도 있을 수 있습니다.

먼저 관리자 계정으로 로그인 한 뒤 프로젝트 폴더를 만들어보겠습니다. Data & Storage 페이지로 이동 후 NEW FOLDER 를 클릭하여 폴더 생성ダイ얼로그를 엽니다. 폴더 이름을 입력하고 Type 을 Project 로 설정한 후 목표 Project를 선택합니다. 이 때, 목표 프로젝트는 사용자 B 가 속한 프로젝트로 설정합니다. Permission 은 Read-Only 로 설정했습니다.

Create a new storage folder ×

Folder name *

Location ▼

Usage Mode General Model

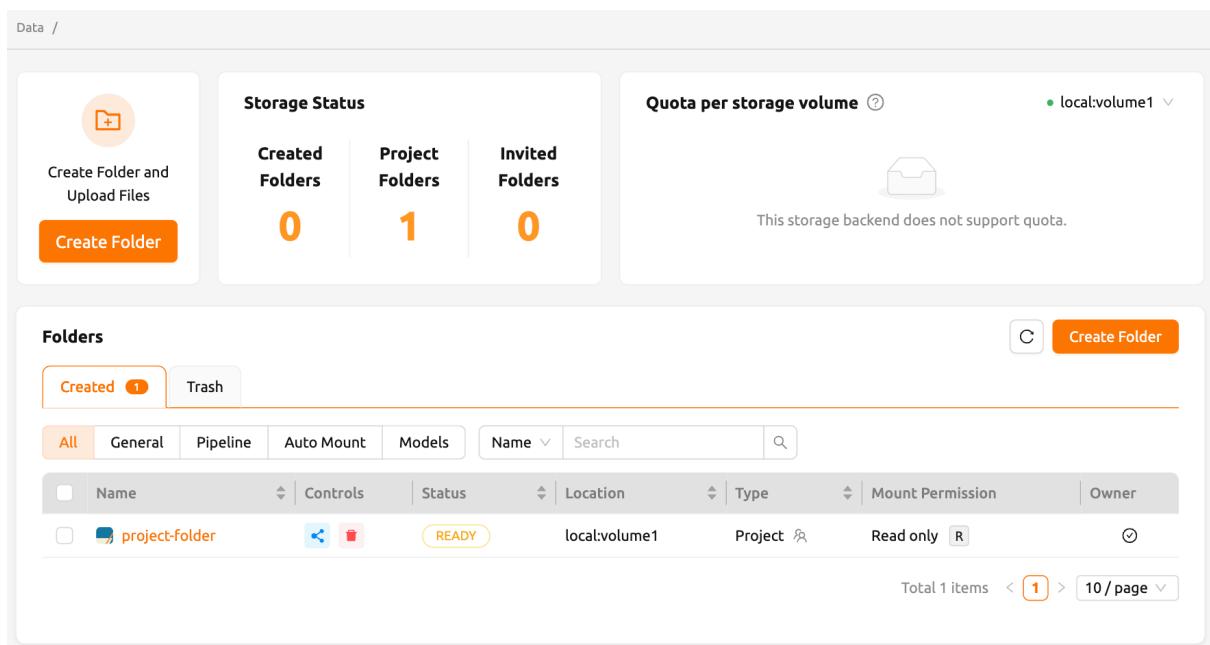
Type User Project

Project ▼

Permission Read & Write Read Only

Reset Cancel Create

폴더가 생성된 것을 확인한 후 사용자 B 계정으로 로그인 하여 Data & Storage 페이지에 방금 생성한 프로젝트 폴더가 별도의 초대 절차 없이 조회되는 것을 확인합니다. Permission 열에 R(읽기전용) 표시가 나타난 것을 확인할 수 있습니다.



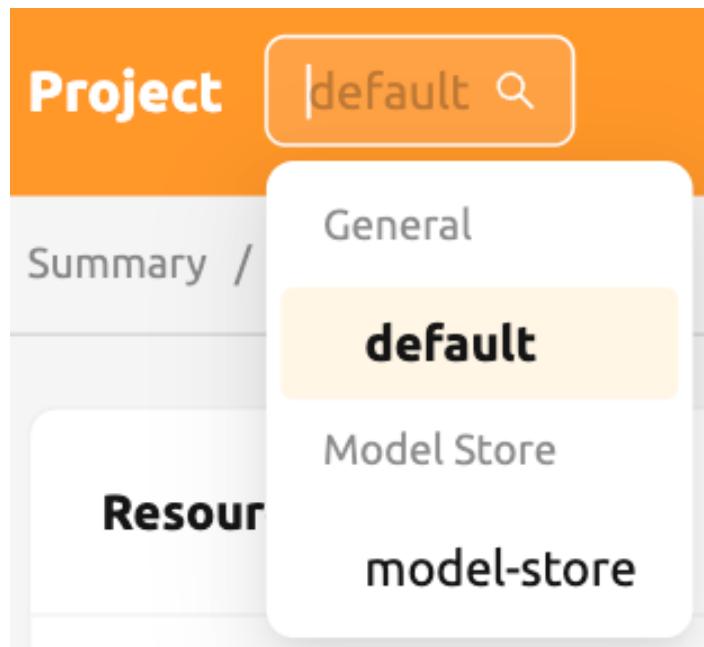
21.5 모델 카드 관리

모델 스토어에 있는 모든 모델 카드는 프로젝트 관리자에 의해 관리됩니다. 모델 스토어에 모델 정의 파일과 함께 모델 폴더를 업로드하면, 프로젝트에 있는 그 어떤 사용자도 모델 카드에 접근할 수 있고 필요하다면 복제할 수도 있습니다.

Hugging Face에 있는 모델을 모델 카드로 추가해봅시다.

참고: 모델 카드를 만들기 전에 Hugging Face에서 복제하고자 하는 모델에 대한 접근 권한을 얻어야 합니다. 더 자세한 내용은 [Gated models](#)를 참고하시기 바랍니다.

'model-store'로 프로젝트를 변경합니다.



오른쪽의 'Add' 버튼을 클릭합니다. 폴더명을 입력하고, 나머지 폴더 설정값을 다음과 같이 설정합니다.

- Type: project
- Project: 'model-store'
- Usage Mode: Model
- Permission: Read-Write
- Cloneable: True

Create a new storage folder

Folder name *

Location

Usage Mode General Model

Type User Project

Project

Permission Read & Write Read Only

Cloneable

Reset **Cancel** **Create**

폴더 생성이 끝나면, model-definition.yaml 파일을 만들고, 방금 생성한 폴더에 업로드해야 합니다. 여기 모델 정의 파일 예시가 있습니다. 모델 정의 파일 작성법이 궁금하시다면, [모델 정의 파일 형식](#)을 참고하시기 바랍니다.

```
models:
- metadata:
  architecture: LlamaForCausalLM
  author: meta-llama
  category: huggingface
  created_at: '2024-04-17 09:35:12'
  description: Meta's Llama 3 by AI@Meta are dialogue-optimized, ↴
```

(다음 페이지에 계속)

(이전 페이지에서 계속)

```

→safe large language
models in 8B and 70B sizes.

framework:
- transformers

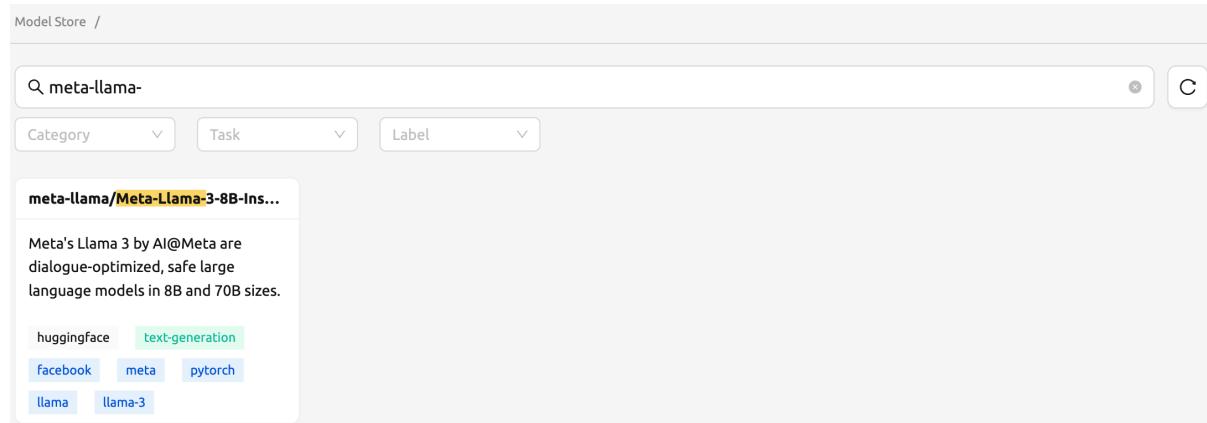
label:
- facebook
- meta
- pytorch
- llama
- llama-3

license: llama3

min_resource:
  cuda.shares: 2.4305981636047362
modified_at: '2024-05-29 12:27:16'
task: text-generation
title: meta-llama/Meta-Llama-3-8B-Instruct
model_path: /models
name: Meta-Llama-3-8B-Instruct

```

model-definition 파일 업로드가 끝나면, 모델 스토어 페이지에 모델 카드가 추가된 것을 확인할 수 있습니다.



참고: 모델 정의 파일 작업이 완료된 뒤, 모델 폴더에 모델을 직접 다운로드해야 합니다. 모델 파일을 폴더에 다운로드 하기 위해서는, 모델 폴더를 세션 생성시 마운트 한 뒤, [Downloading models](#) 를 참고하여 파일을 다운로드 받을 수 있습니다.

방금 생성한 모델 카드를 클릭하면 방금 업로드한 모델 정의 파일의 상세 정보를 확인할 수 있습니다. 이제 프로젝트 구성원 모두가 모델 카드에 접근 및 복제할 수 있게 되었습니다.

Description

Meta's Llama 3 by AI@Meta are dialogue-optimized, safe large language models in 8B and 70B sizes.

Model Details

Meta developed and released the Meta Llama 3 family of large language models (LLMs), a collection of pretrained and instruction tuned generative text models in 8 and 70B sizes. The Llama 3 instruction tuned models are optimized for dialogue use cases and outperform many of the available open source chat models on common industry benchmarks. Further, in developing these models, we took great care to optimize helpfulness and safety.

Model developers Meta

Variations Llama 3 comes in two sizes — 8B and 70B parameters — in pre-trained and instruction tuned variants.

Input Models input text only.

Output Models generate text and code only.

Model Architecture Llama 3 is an auto-regressive language model that uses an optimized transformer architecture. The tuned versions use supervised fine-tuning (SFT) and reinforcement learning with human feedback (RLHF) to align with human preferences for helpfulness and safety.

Training Data	Params	Context length	GQA	Token count	Knowledge cutoff
Llama 3 A new mix of publicly available online data.	8B 70B	8k 8k	Yes Yes	15T+	March, 2023 December, 2023

21.6 자원 정책 관리

21.6.1 키페어 자원 정책

Backend.AI에서 관리자는 각 키페어, 사용자 및 프로젝트에 대해 사용 가능한 자원의 총량에 대한 제한을 설정할 수 있습니다. 자원 정책을 사용하면 최대 허용 자원 및 기타 계산 세션 관련 설정을 정의할 수 있습니다. 필요한 경우 사용자 또는 연구 목적과 같이 여러 자원 정책을 만들고 개별적으로 적용할 수 있습니다.

자원 정책 페이지에서 관리자는 현재 등록된 자원 정책 목록을 확인할 수 있습니다. 자원 정책 페이지에서는 키페어, 사용자 및 프로젝트에 설정된 자원 정책 목록을 확인할 수 있습니다. 먼저 키페어에 대한 자원 정책을 살펴봅시다. 아래 그림에서는 총 세 가지 정책 (gardener, student, default)이 있습니다. 무한대 기호 (∞)는 해당 리소스에 제한이 없음을 나타냅니다.

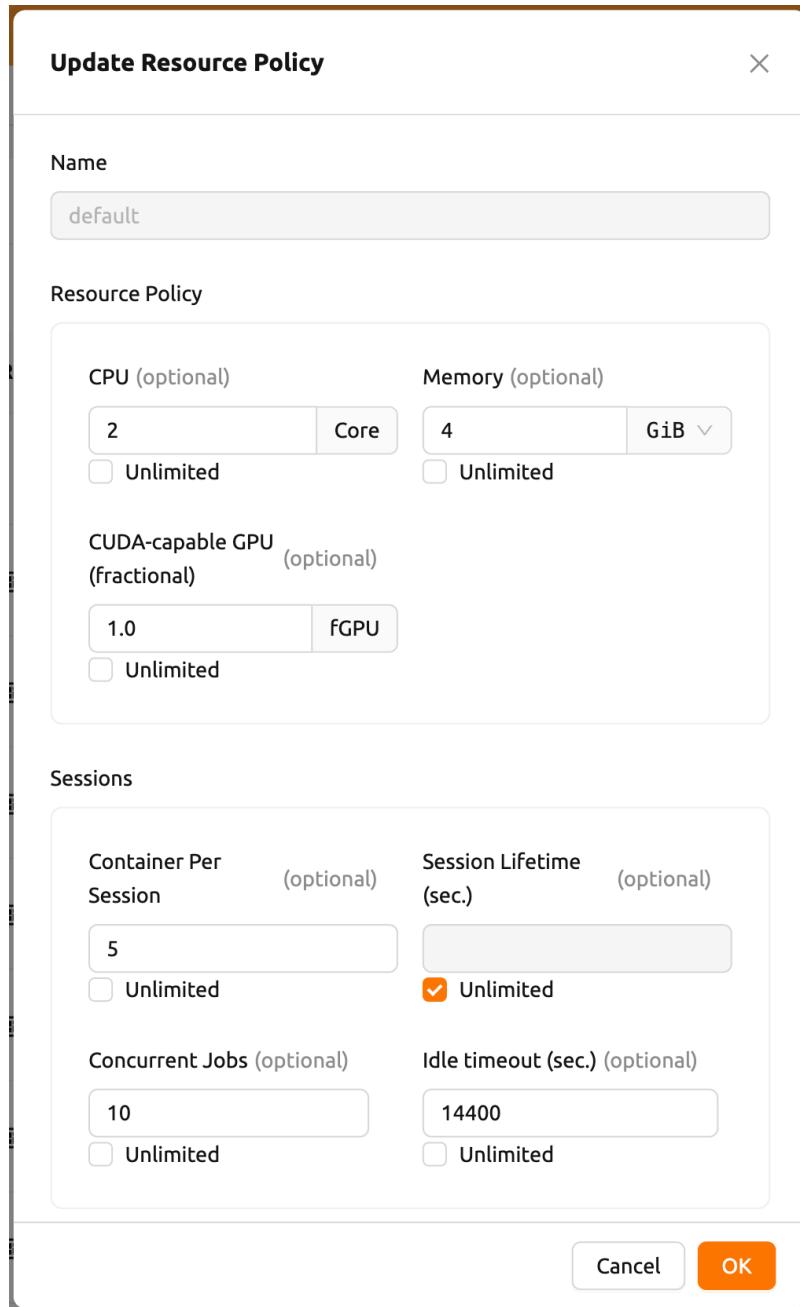
Keypair	User	Project					
Tools							
Name	Resources	Concurrency	Cluster Size	Idle Timeout	Max Session Lifetime	Storage Nodes	Control
default	-	5	1	3600	∞	local:volume1	
gardener	-	∞	∞	∞	∞	local:volume1	
student	-	∞	∞	∞	∞	local:volume1	

이 가이드에서 사용 중인 사용자 계정은 현재 default 자원 정책에 할당되어 있습니다. 이 정보는 사용자 페이지의 자격 증명 탭에서 확인할 수 있습니다. 자원 정책 패널에서는 모든 자원이 기본 정책으로 설정되어있는 것을 확인할 수 있습니다.

Users	Credentials							
Active	Inactive	User ID	Permission	Key Age	Created At	Resource Policy	Allocation	Control
domain-admin@lablup.com	AMPLE		admin user	5Days	Apr 4, 2025 6:54 PM	default	0 Sessions 30000 Req per 15 min 0 Queries	
admin@lablup.com	AMPLE		admin user	5Days	Apr 4, 2025 6:54 PM	default	2 Sessions 30000 Req per 15 min 16226 Queries	
user@lablup.com	AMPLE		user	5Days	Apr 4, 2025 6:54 PM	default	0 Sessions 30000 Req per 15 min 1100 Queries	
monitor@lablup.com	AMPLE		user	5Days	Apr 4, 2025 6:54 PM	default	0 Sessions 30000 Req per 15 min 0 Queries	
user2@lablup.com	AMPLE		user	5Days	Apr 4, 2025 6:54 PM	default	0 Sessions 30000 Req per 15 min 0 Queries	

1-5 of 5 items < 1 > 20 / page

자원 정책을 수정하려면 기본 정책 그룹의 제어 열에 있는 '설정' 버튼을 클릭합니다. 자원 정책 업데이트 다이얼로그에서 정책 이름을 제외한 모든 옵션을 편집할 수 있으며, 이는 목록에서 자원 정책을 구별하는 주요 키 역할을 합니다. 여기에서는 CPU, RAM 및 fGPU 하단의 무제한 확인란을 해제하고 원하는 값으로 리소스 제한을 설정하겠습니다. 할당된 리소스는 총 하드웨어 용량보다 작아야 합니다. 이 경우 CPU, RAM 및 fGPU를 각각 2, 4, 1로 설정하겠습니다. 업데이트된 자원 정책을 적용하려면 OK 버튼을 클릭합니다.



자원 정책 창의 각 옵션에 대한 자세한 내용은 아래 설명을 참조하세요.

• 자원 정책

- CPU : CPU 코어 최대 할당 가능량 설정. (최대 입력 값: 512)
- Memory : GB 단위로 최대 할당 가능한 메모리 양 설정. 최대 할당 가능한 GPU 메모리 양의 두 배 이상으로 설정하는 것이 권장됨. (최대 입력 값: 1024)
- CUDA-capable GPU : 할당 가능한 최대 물리 GPU 개수 설정. 서버에서 GPU 분할 가상화 기능이 켜져 있는 경우에는 적용되지 않음. (최대 입력 값: 64) * 대부분의 엔터프라이즈 사이트에는 GPU 분할 가상화 기능이 켜져 있음.
- CUDA-capable GPU (fractional) : Backend.AI 의 Fractional GPU (fGPU)

기능이 켜져 있으면, 물리 GPU 를 분할하여 여러 사용자 세션에 나뉘어 배치하고 효율적으로 사용할 수 있음. 할당 가능한 최대 분할 GPU 양은 여기서 설정합니다. 만약 서버에 GPU 분할 가상화 기능이 꺼져 있다면, 이 항목은 적용되지 않습니다. (최대 입력 값: 256)

• 연산 세션

- Container Per Session : 한 세션이 가질 수 있는 최대 컨테이너의 수. 사용자가 클러스터 세션을 생성할 수 있게 하려면, 이 값이 1보다 커야 함. (최대 입력 값: 100)
- Session Lifetime (sec.): 세션이 예약 시간 이후 활성 상태로 존재할 수 있는 최대 시간. 활성 상태는 PENDING 및 RUNNING 상태를 포함함. 예약 후 설정된 시간이 지나면 세션이 사용중이라 할지라도 강제로 종료됨. 사용중이라 할지라도 강제로 종료됩니다. 연산 세션이 무한히 실행되는 것을 방지할 때 유용한 옵션.
- Concurrent Jobs: 키페어를 통해 동시에 생성할 수 있는 최대 연산 세션의 개수입니다. 예를 들어, 이 값이 3 으로 지정되어 있을 경우, 해당 정책에 영향을 받는 사용자는 동시에 3개가 넘는 연산 세션을 생성할 수 없음. (최대 입력 값: 100)
- Idle timeout (sec.): 사용자가 세션에 영향을 주지 않고 비활성 상태일 수 있는 최대 시간. 유휴 제한 시간 동안 연산 세션에 활동이 감지되지 않으면, 해당 세션이 자동으로 삭제됨. 활동을 측정하는 기준은 다양하며, 관리자가 설정할 수 있음. (최대 입력 값: 15552000(초) (약 180 일))

• 폴더

- Allowed hosts : Backend.AI 는 여러 NFS 마운트포인트를 인식할 수 있음. 이 항목을 통해 사용자가 접근할 수 있는 마운트 위치를 지정할 수 있음. 예를 들어, "data-1" 이라는 NFS 가 Backend.AI 에서 인식되어 있는 상태일지라도, Allowed hosts 정책으로 허용되지 않은 경우 사용자는 해당 NFS 에 접근할 수 없음.
- (23.09.4 이후로 사용 중단) Max. #: 생성 또는 공유 초대 받을 수 있는 저장 폴더의 최대 개수. (최대 입력 값: 50)

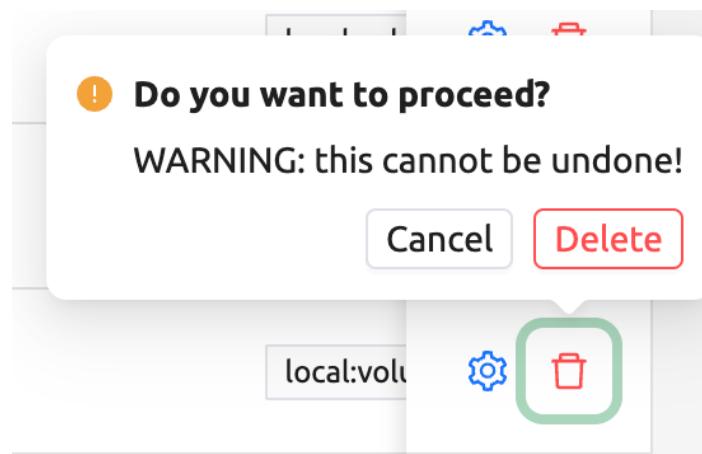
키페어 자원 정책 리스트에서 default 정책의 Resources 값이 업데이트 된 것을 확인합니다.

Name	Resources
default	2 Core 4 GiB 1.00 fGPU

Create 버튼을 클릭하여 새로운 자원 정책을 생성할 수 있습니다. 각 설정값의 의미는 상기한 설명과 동일합니다.

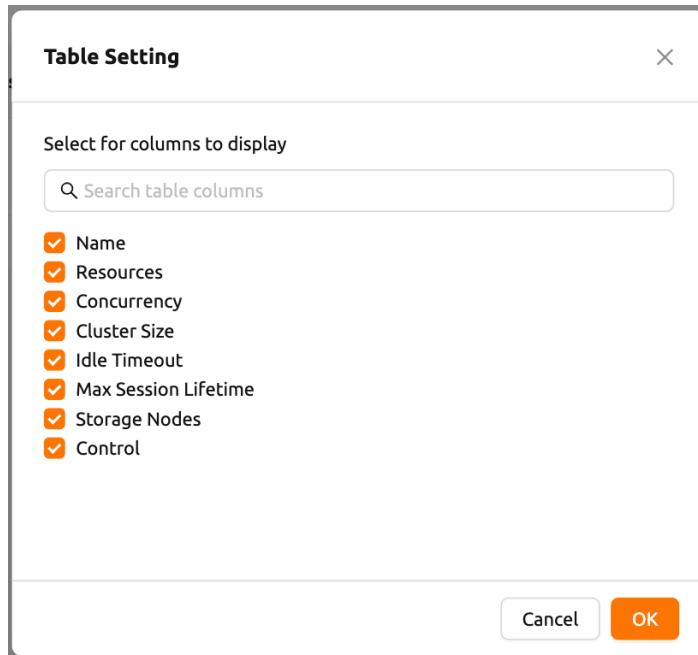
키퍼어 자원 정책을 생성한 후 키페어에 연결하기 위해서는 Users 페이지의 Credentials 탭으로 가서 원하는 키페어의 Controls 열에 위치한 '설정' 버튼을 누른 후, Select Policy 필드를 클릭하여 선택합니다.

Control 열의 '휴지통' 아이콘을 클릭하여 각 리소스 키페어를 삭제할 수 있습니다. 아이콘을 클릭하면 확인 팝업이 나타납니다. 삭제하려면 'Delete' 버튼을 클릭하세요.



참고: 비활성 사용자를 포함한 어떤 유저가 삭제할 자원 정책을 따르고 있다면, 삭제가 되지 않을 수 있습니다. 자원 정책을 삭제하기 위해서는 반드시 해당 자원 정책을 선택한 사용자가 남아 있지 않아야 합니다.

특정 열을 숨기거나 보고 싶다면 테이블 우측 하단의 '설정' 버튼을 클릭합니다. 이후에는 아래에 다이얼로그가 표시되어 보고 싶은 열을 선택할 수 있습니다.

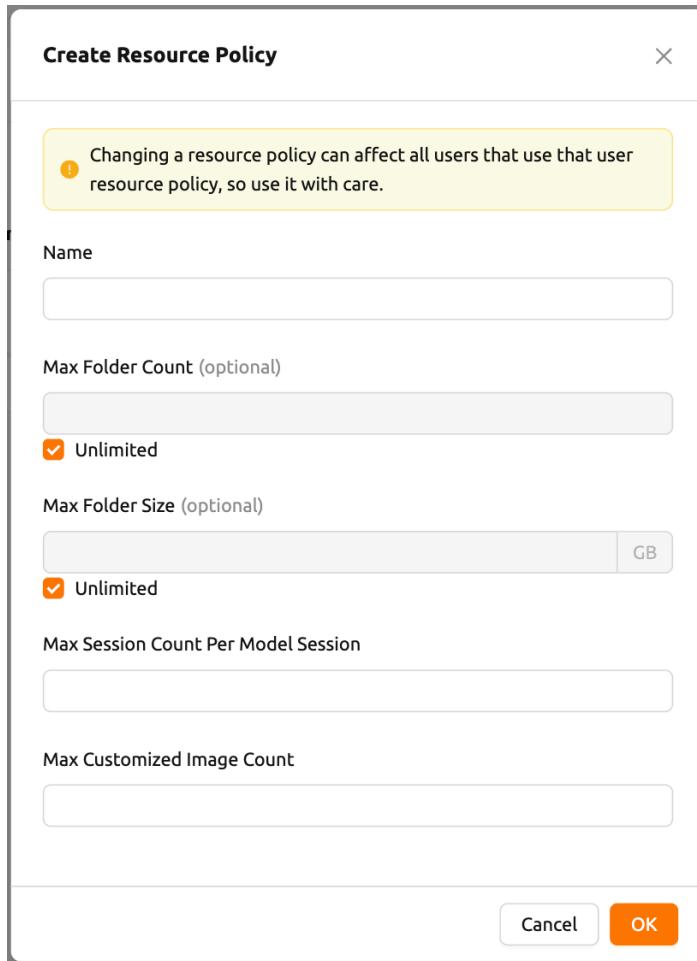


21.6.2 사용자 자원 정책

24.03 버전부터 Backend.AI는 사용자 자원 정책 관리를 지원합니다. 각 사용자는 여러 키페어를 가질 수 있지만 사용자 당 사용자 자원 정책은 하나만 가질 수 있습니다. 사용자 자원 정책 페이지에서는 Max Folder Count 및 Max Folder Size와 같은 폴더 관련 다양한 설정을 제한 할 수 있으며, Max Session Count Per Model Session 및 Max Customized Image Count와 같은 개별 리소스를 제한 할 수 있습니다.

Name	Max Folder Count	Max Session Count Per Model Session	Max Quota Scope Size (GB)	Max Customized Image Count	Control
default	∞	10	∞	3	

테이블 오른쪽 위에 있는 Create 버튼을 클릭하여 새로운 사용자 자원 정책을 생성할 수 있습니다.



- **Name:** 사용자 자원 정책의 이름.
- **Max Folder Count:** 사용자가 생성할 수 있는 최대 폴더 수. 사용자의 폴더 수가 해당 값을 초과하면 새 폴더를 생성할 수 없게 되며, 무제한으로 설정된 경우에는 “∞”로 표시됨.
- **Max Folder Size:** 사용자의 저장 공간의 최대 크기. 사용자의 저장 공간이 해당 값을 초과하면 사용자는 새 데이터 폴더를 만들 수 없게 되며, 무제한으로 설정된 경우에는 “∞”로 표시됨.
- **Max Session Count Per Model Session:** 각 사용자가 생성한 모델 서비스 당 사용 가능한 세션의 최대 수. 해당 값을 증가시키면 세션 스케줄러에 부하가 가해질 수 있고, 잠재적인 시스템 다운타임으로 이어질 수 있습니다. 따라서, 이 설정을 조정할 때는 주의가 필요합니다.
- **Max Customized Image Count:** 사용자가 생성할 수 있는 최대 사용자 정의 이미지 개수. 사용자의 사용자 정의 이미지 수가 이 값을 초과하면, 사용자는 새로운 사용자 정의 이미지를 만들 수 없음. 사용자 정의 이미지에 대해 더 알고 싶다면, [나의 실행 환경](#) 섹션을 참조 할 수 있음.

Control 열의 '설정' 버튼을 클릭합니다. 삭제하려면 '휴지통' 버튼을 클릭하세요.

참고: 자원 정책을 변경하면 해당 사용자 자원 정책을 사용하는 모든 사용자에게 영향을 줄 수

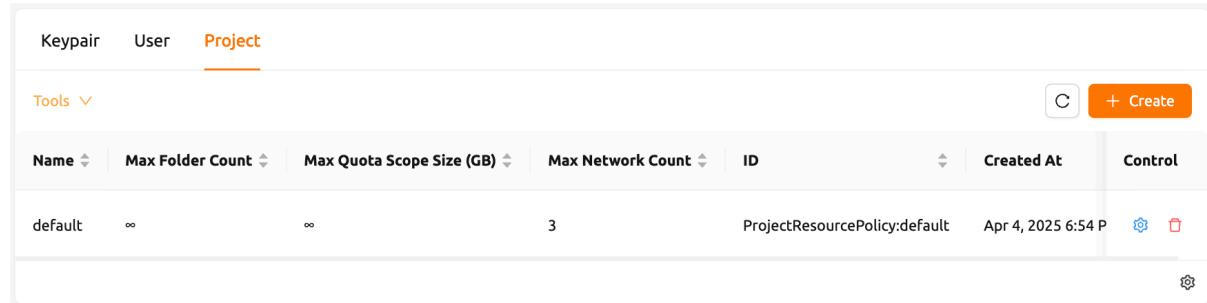
있습니다. 자원 정책을 변경할 때는 사용자의 주의가 필요합니다.

키페어 자원 정책과 마찬가지로, 원하는 열만 선택하여 표시하려면 테이블 오른쪽 하단의 '설정' 버튼을 클릭하세요.

21.6.3 프로젝트 자원 정책

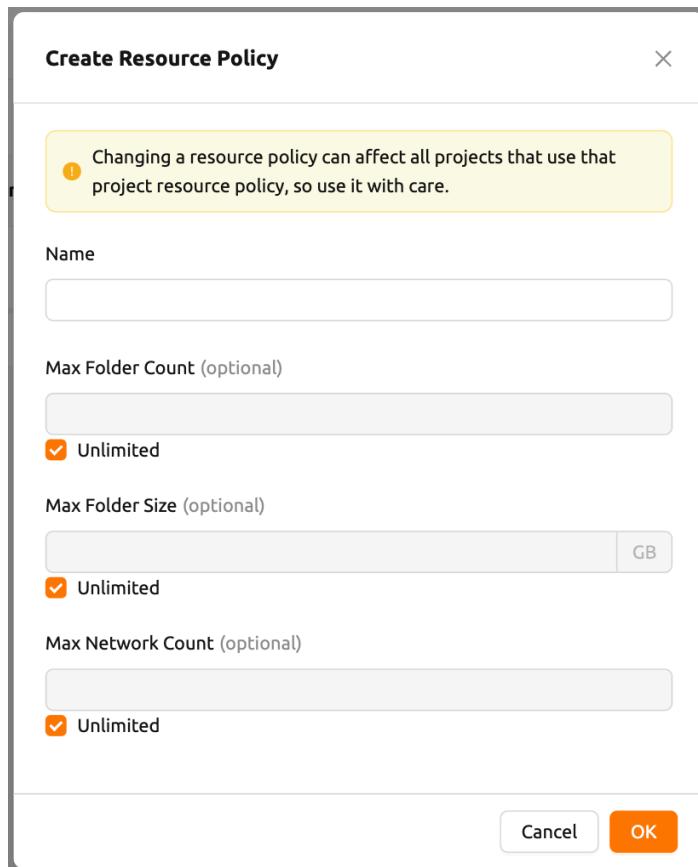
24.03 버전부터 Backend.AI는 프로젝트 자원 정책 관리를 지원합니다. 프로젝트 자원 정책은 프로젝트의 저장 공간(할당량) 및 폴더 관련 제한 사항을 관리합니다.

Resource Policy 페이지의 'Project' 탭을 클릭하면 프로젝트 자원 정책 목록을 볼 수 있습니다.



A screenshot of the Backend.AI Resource Policy interface. The top navigation bar has tabs for 'Keypair', 'User', and 'Project', with 'Project' being the active tab. Below the tabs is a toolbar with a 'Tools' dropdown, a search bar, and a large orange '+ Create' button. The main area is a table with columns: Name, Max Folder Count, Max Quota Scope Size (GB), Max Network Count, ID, Created At, and Control. There is one row visible with the name 'default', max folder count as infinity, max quota scope size as infinity, max network count as 3, ID as 'ProjectResourcePolicy:default', created at 'Apr 4, 2025 6:54 P', and control icons.

테이블 오른쪽 위에 있는 'Create' 버튼을 클릭하여 새로운 프로젝트 자원 정책을 생성할 수 있습니다.



A screenshot of the 'Create Resource Policy' dialog box. It contains a warning message: 'Changing a resource policy can affect all projects that use that project resource policy, so use it with care.' Below the message are input fields for 'Name', 'Max Folder Count (optional)', and 'Max Folder Size (optional)'. Both fields have an 'Unlimited' checkbox checked. Below these are 'Max Network Count (optional)' and another 'Unlimited' checkbox. At the bottom right are 'Cancel' and 'OK' buttons.

- Name: 프로젝트 자원 정책의 이름.
- Max Folder Count: 관리자가 생성할 수 있는 프로젝트 폴더의 최대 개수. 프로젝트 폴더 수가 해당 값을 초과하면 관리자가 새 프로젝트 폴더를 생성할 수 없게 되며, 무제한으로 설정된 경우에는 “∞”로 표시됨.
- Max Folder Size: 프로젝트의 저장 공간의 최대 크기. 프로젝트의 저장 공간이 이 값을 초과하면 관리자가 새 프로젝트 폴더를 생성할 수 없게 되며, 무제한으로 설정된 경우에는 “∞”로 표시됨.
- Max Network Count: Backend.AI 버전 24.12 부터 가능한 프로젝트 별 네트워크 생성 최대 갯수. 무제한으로 설정된 경우에는 “∞”로 표시됨.

각 필드의 의미는 사용자 자원 정책과 유사합니다. 차이점은 프로젝트 자원 정책이 프로젝트 폴더에 적용되는 반면 사용자 자원 정책은 사용자 폴더에 적용된다는 것입니다.

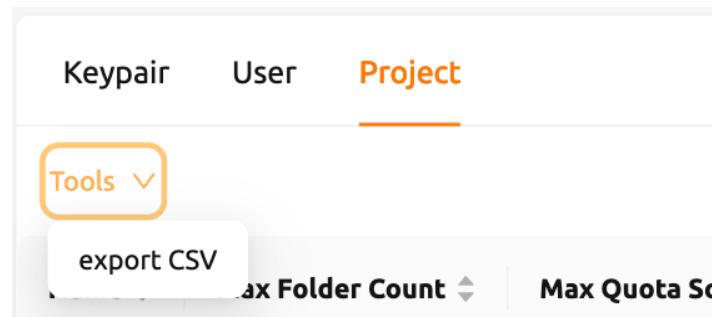
변경을 원하면 제어 열의 '설정' 버튼을 클릭하세요. 자원 정책 이름은 편집할 수 없습니다. 삭제는 '휴지통' 버튼을 클릭하여 수행할 수 있습니다.

참고: 자원 정책을 변경하면 해당 사용자 자원 정책을 사용하는 모든 사용자에게 영향을 줄 수 있습니다. 자원 정책을 변경할 때는 사용자의 주의가 필요합니다.

원하는 열만 선택하여 표시하려면 테이블 오른쪽 하단의 '설정' 버튼을 클릭하세요.

21.6.4 프로젝트 자원 정책 수정 다이얼로그

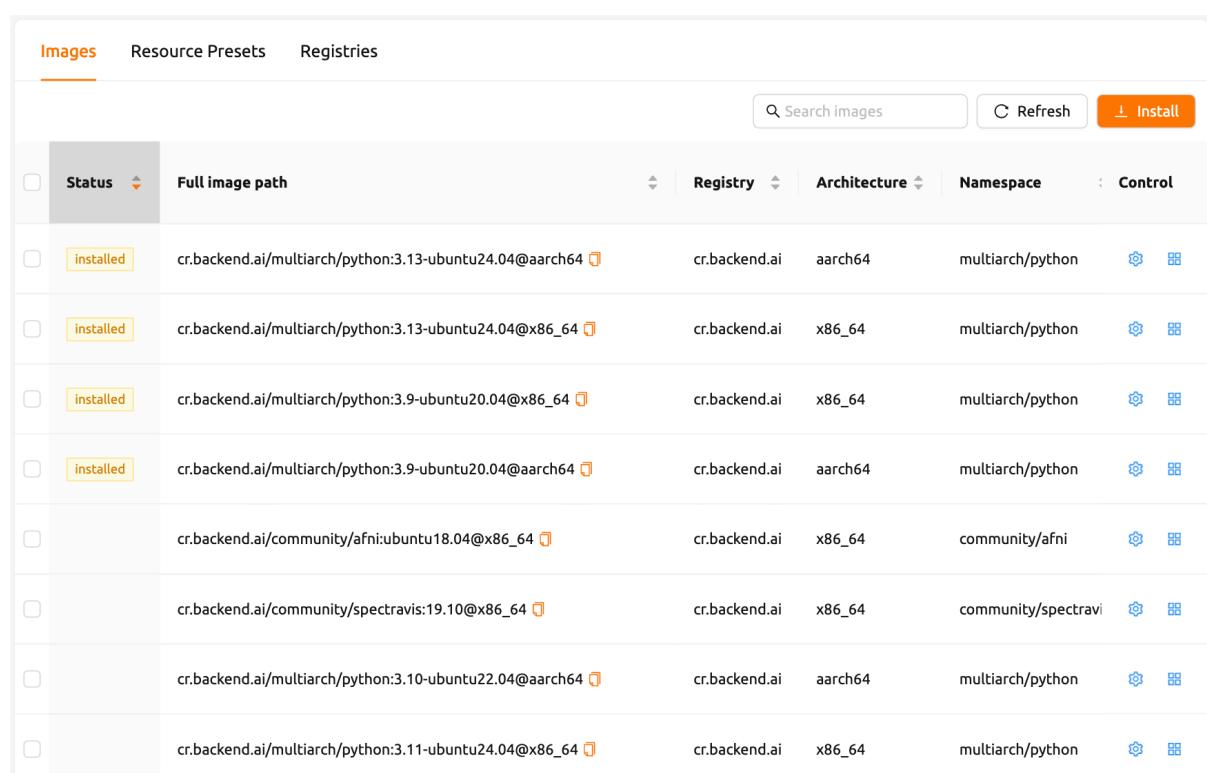
현재 자원 정책을 파일로 저장하고 싶다면, 각 탭의 왼쪽 위에 위치한 'Tools' 메뉴를 클릭합니다. 메뉴를 클릭하면, 파일 다운로드 모달이 띄워지게 됩니다.



21.7 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 `installed` 태그가 각 이미지의 Status 컬럼에 표시됩니다.

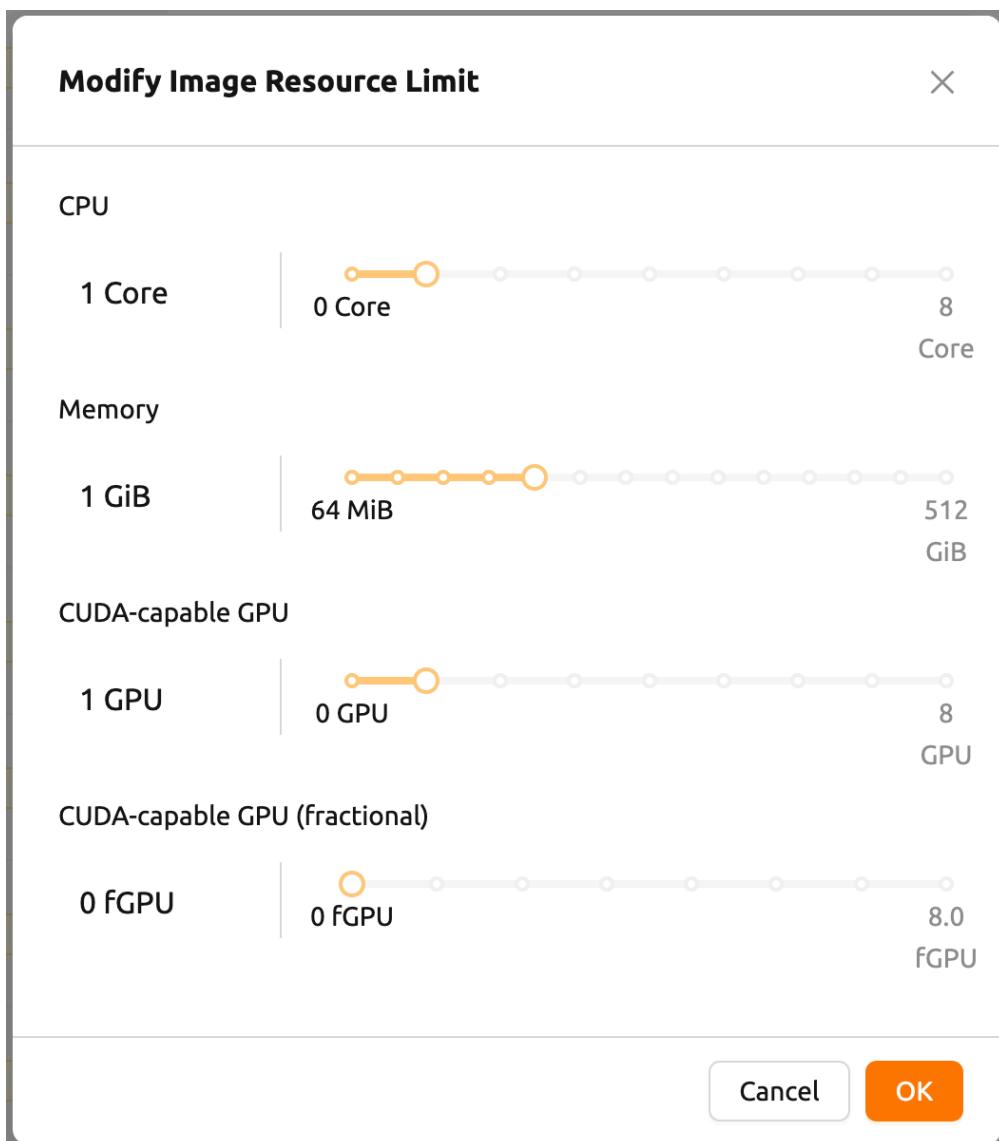
참고: 특정 agent를 선택하여 이미지를 설치하는 기능은 현재 개발 중에 있습니다.



The screenshot shows the 'Images' tab of the Backend.AI WebUI. At the top, there are three tabs: 'Images' (highlighted in orange), 'Resource Presets', and 'Registries'. Below the tabs is a search bar with placeholder text 'Search images' and a 'Refresh' button. An 'Install' button is also visible. The main area is a table with the following columns: 'Status' (with a dropdown arrow), 'Full image path', 'Registry', 'Architecture', 'Namespace', and 'Control'. There are eight rows in the table, each representing a different Docker image. Each row includes a checkbox, the status 'installed' in a yellow box, the full image path (e.g., cr.backend.ai/multiarch/python:3.13-ubuntu24.04@aarch64), the registry (cr.backend.ai), architecture (aarch64 or x86_64), namespace (multiarch/python), and two control icons (gear and cloud).

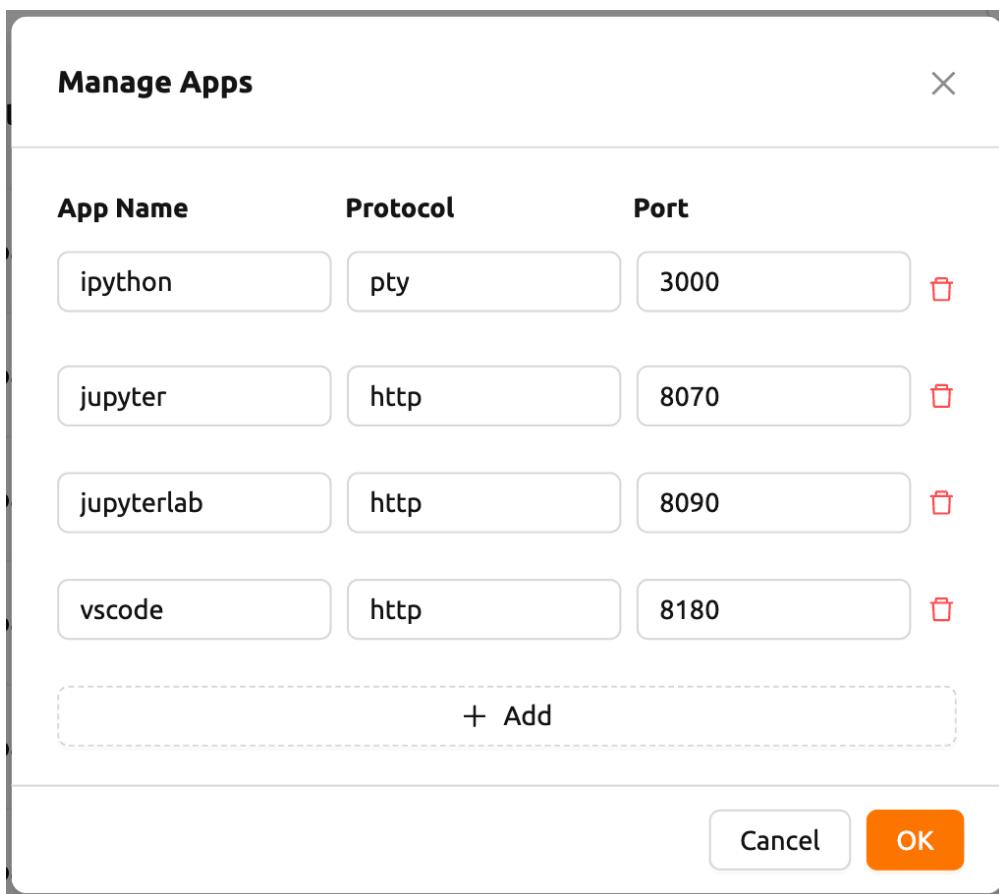
Status	Full image path	Registry	Architecture	Namespace	Control
installed	cr.backend.ai/multiarch/python:3.13-ubuntu24.04@aarch64	cr.backend.ai	aarch64	multiarch/python	 
installed	cr.backend.ai/multiarch/python:3.13-ubuntu24.04@x86_64	cr.backend.ai	x86_64	multiarch/python	 
installed	cr.backend.ai/multiarch/python:3.9-ubuntu20.04@x86_64	cr.backend.ai	x86_64	multiarch/python	 
installed	cr.backend.ai/multiarch/python:3.9-ubuntu20.04@aarch64	cr.backend.ai	aarch64	multiarch/python	 
	cr.backend.ai/community/afni:ubuntu18.04@x86_64	cr.backend.ai	x86_64	community/afni	 
	cr.backend.ai/community/spectravis:19.10@x86_64	cr.backend.ai	x86_64	community/spectravis	 
	cr.backend.ai/multiarch/python:3.10-ubuntu22.04@aarch64	cr.backend.ai	aarch64	multiarch/python	 
	cr.backend.ai/multiarch/python:3.11-ubuntu24.04@x86_64	cr.backend.ai	x86_64	multiarch/python	 

Controls 열의 '설정' 버튼을 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 지정 된 자원의 양보다 작은 자원으로 세션을 생성하려고 할 경우, 해당 요청은 자동으로 이미지 최소 자원 요구량으로 조정된 후 생성됩니다.



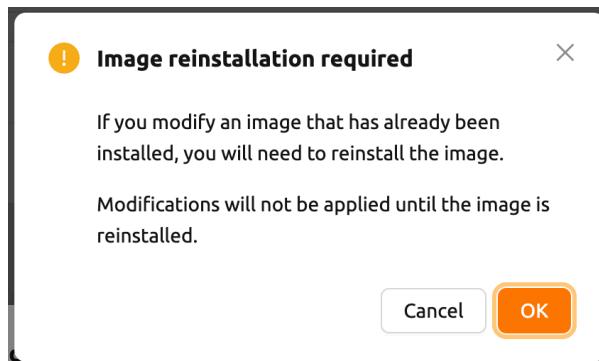
경고: 미리 지정된 값보다 작은 양으로 최소 자원 요구량을 변경하지 마세요! 이미지 메타데이터에 포함된 최소 자원 요구량은 테스트를 거쳐 결정된 값입니다. 변경하려는 최소 자원량에 대하여 정말 잘 알고 있는 것이 아니라면, 기본값으로 남겨두시기 바랍니다.

또한 Controls 열의 '앱' 버튼을 클릭하여 이미지 별로 지원하는 앱을 추가하거나 수정할 수 있습니다. 앱 버튼을 클릭하면, 이미지에서 지원하는 앱과 대응하는 포트넘버를 확인하실 수 있습니다.



이 모달에서는 'Add' 버튼을 클릭하여 지원되는 커스텀 앱을 추가할 수 있습니다. 앱을 삭제하고 싶다면, 각 행에 오른쪽에 있는 '빨간색 휴지통 버튼'을 클릭하면 됩니다.

참고: 이미지에서 지원하는 앱을 수정할 경우에는 이미지 재설치가 필요합니다.

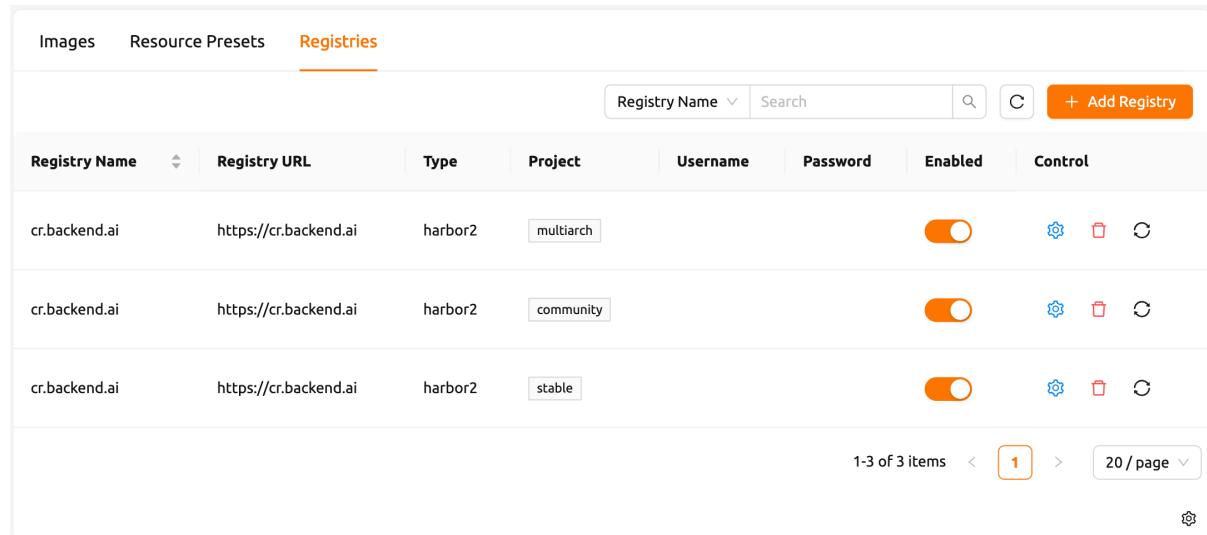


21.8 도커 레지스트리 관리

Environments의 'Registries' 탭을 클릭하여 현재 연결되어 있는 도커 레지스트리의 정보를 확인할 수 있습니다. cr.backend.ai는 Harbor에서 서비스 하는 레지스트리로 기본적으로 등록되어 있습니다.

참고: 오프라인 환경일 경우 기본 도커 레지스트리에 접근이 불가능하므로, 우측 '휴지통' 버튼을 클릭하여 삭제하면 됩니다.

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



The screenshot shows the 'Registries' tab in the Backend.AI WebUI. It lists three registry entries:

Registry Name	Registry URL	Type	Project	Username	Password	Enabled	Control
cr.backend.ai	https://cr.backend.ai	harbor2	multiarch			<input checked="" type="checkbox"/>	
cr.backend.ai	https://cr.backend.ai	harbor2	community			<input checked="" type="checkbox"/>	
cr.backend.ai	https://cr.backend.ai	harbor2	stable			<input checked="" type="checkbox"/>	

At the bottom, there is a pagination indicator showing '1-3 of 3 items' and a page size selector set to '20 / page'.

ADD REGISTRY 버튼을 클릭하여 운영하고 있는 사설 도커 레지스트리를 추가할 수 있습니다. 이 때, Registry Name과 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 Name을 접두어로 한 이름을 가져야 합니다. Username과 Password는 선택 사항으로, 레지스트리에서 별도 인증 설정을 한 경우에는 채워주시면 됩니다. Extra Information에서, 각 레지스트리 타입 별로 필요한 추가 정보를 JSON 문자열로 넘겨줄 수 있습니다.

The screenshot shows a modal dialog titled "Add Registry". It contains the following fields:

- Registry Name: An input field.
- Registry URL: An input field.
- Username (optional): An input field.
- Password (optional): An input field with a visibility toggle icon.
- Registry Type: A dropdown menu.
- Project Name: An input field.
- Extra Information (optional): A text area containing the number "1".

At the bottom right are two buttons: "Cancel" and "Add".

Registry Name을 제외하고 이미 존재하는 레지스트리 정보를 수정할 수 있습니다.

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 프로젝트 수준에서 allowed docker registries 필드에 해당 레지스트리를 등록해야 도메인 또는 프로젝트 소속 사용자가 레지스트리 이미지에 접근할 수 있습니다. Allowed docker registries 등록은 도메인과 프로젝트 관리 기능이 있는 Control-Panel 을 이용해서 할 수 있습니다. 키페어의 자원 정책에서 allowed docker registries를 설정하는 기능은 아직 제공하지 않고 있습니다.

21.9 자원 프리셋 설정

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

The screenshot shows the 'Resource Presets' section of the Backend.AI WebUI. At the top, there is a search bar containing 'cpu01-small'. Below the search bar, there are two tabs: 'Custom allocation' and 'Minimum requirements'. Under 'Custom allocation', there is a list of predefined presets:

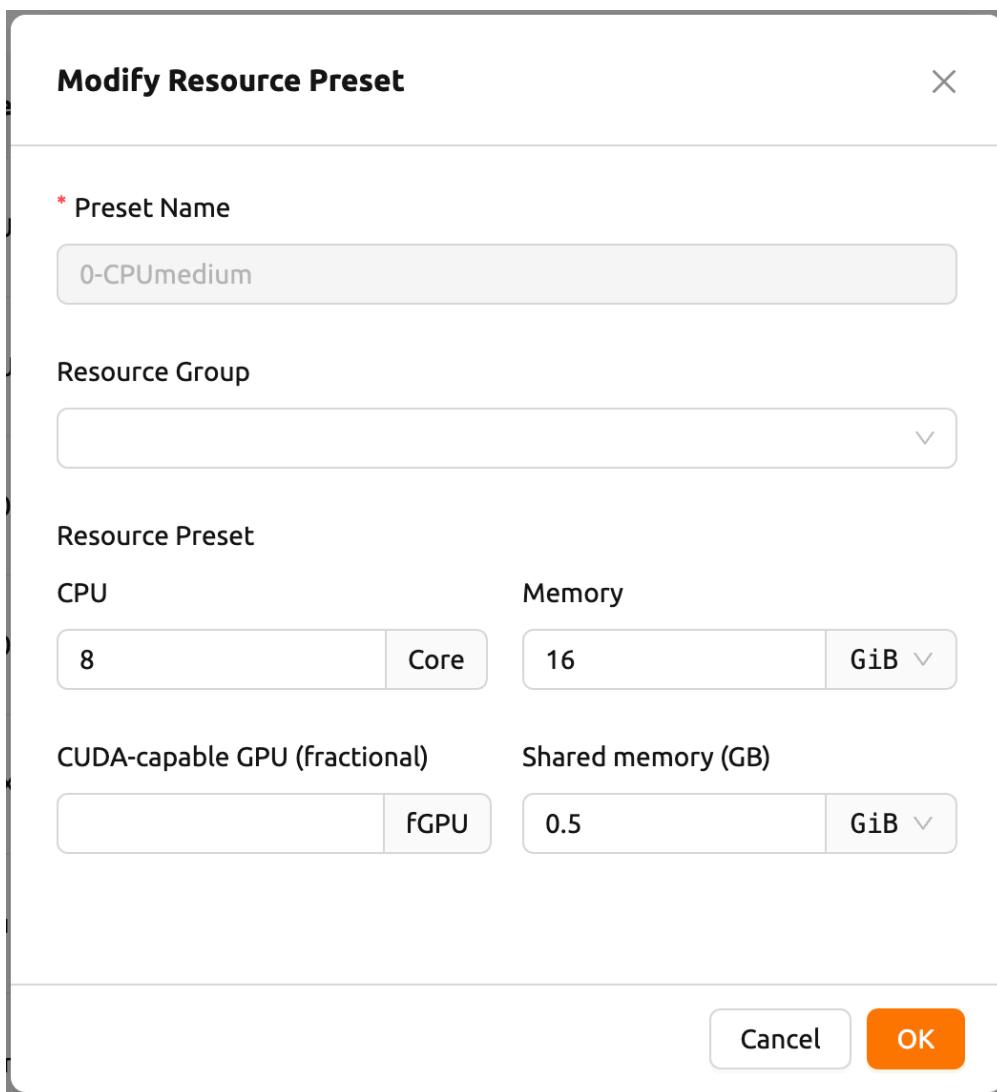
Preset	Resources
cpu01-small	1 Core 1 GiB
cpu02-medium	4 Core 8 GiB
cpu03-large	8 Core 16 GiB
cuda01-small	1 Core 4 GiB
cuda02-medium	4 Core 16 GiB

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

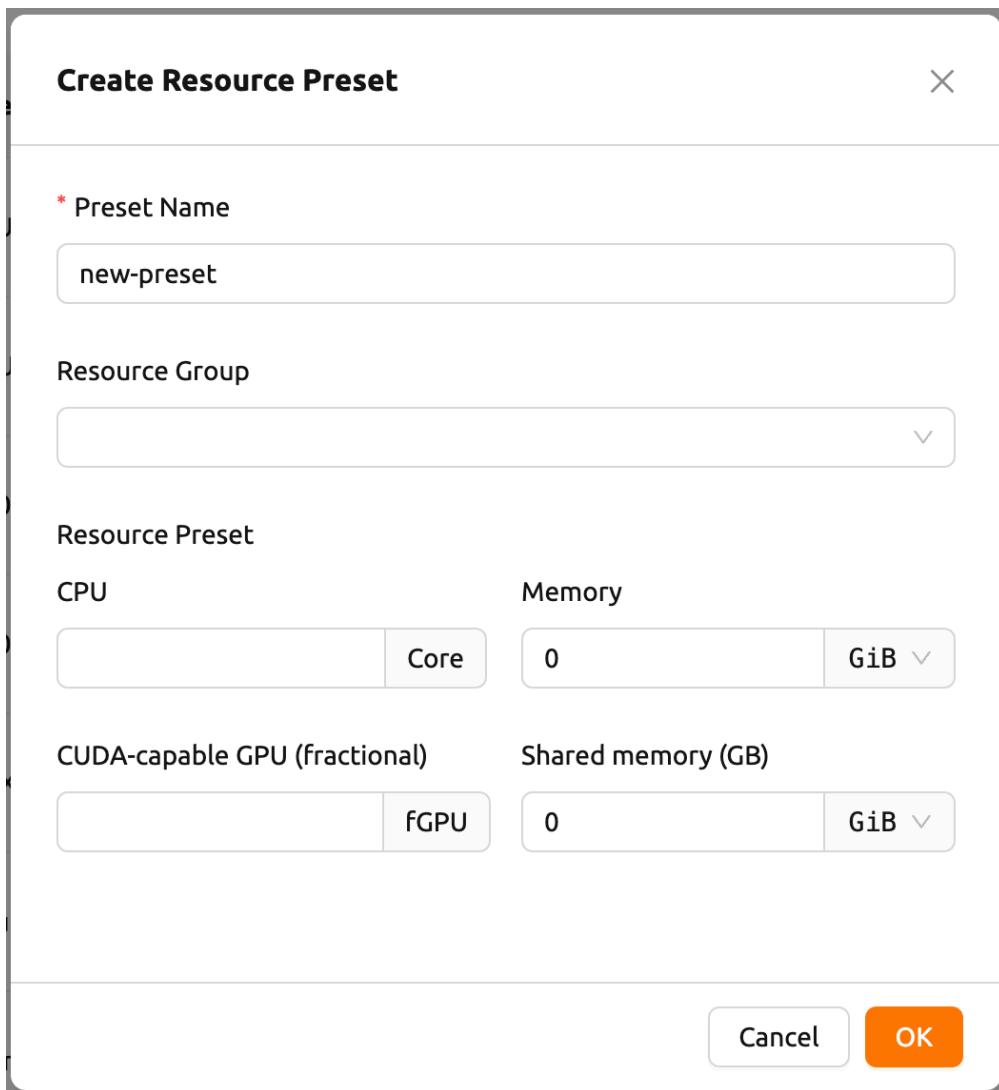
The screenshot shows the 'Resource Presets' table in the Backend.AI WebUI. The table has columns for Name, Resources, Shared memory (GB), Resource Group, and Control. There are six rows of data:

Name	Resources	Shared memory (GB)	Resource Group	Control
cpu01-small	1 Core 1 GiB	-	-	
cpu02-medium	4 Core 8 GiB	-	-	
cpu03-large	8 Core 16 GiB	-	-	
cuda01-small	1 Core 4 GiB 1 GPU 0.50 fGPU	-	-	
cuda02-medium	4 Core 16 GiB 2 GPU 2.00 fGPU	-	-	
cuda03-large	8 Core 32 GiB 4 GPU 4.00 fGPU	-	-	

Control 패널에서 '설정' 버튼을 클릭하여 자원 프리셋에서 제공할 CPU, RAM, FGPU 등의 자원을 설정할 수 있습니다. 자원 프리셋 생성 또는 수정 모달에서는 현재 사용 가능한 자원(서버 설정에 따라 특정 자원이 표시되지 않을 수 있습니다)을 기반으로 프리셋을 수정할 수 있습니다. 자원 프리셋 수정 후, 연산 세션을 생성할 때 해당 프리셋이 표시되는지 확인하세요. 사용 가능한 자원이 사전 프리셋에 설정된 자원보다 적은 경우 해당 프리셋이 표시되지 않습니다.



또한 Resource Presets 탭의 우측 상단의 CREATE PRESETS 버튼을 클릭하여 자원 프리셋을 생성 할 수도 있습니다. 이미 존재하는 자원 프리셋 이름으로는 생성이 불가능한데, 이는 프리셋 이름이 각 자원 프리셋을 구분하는 키 값이기 때문입니다.



21.10 Agent 노드 관리

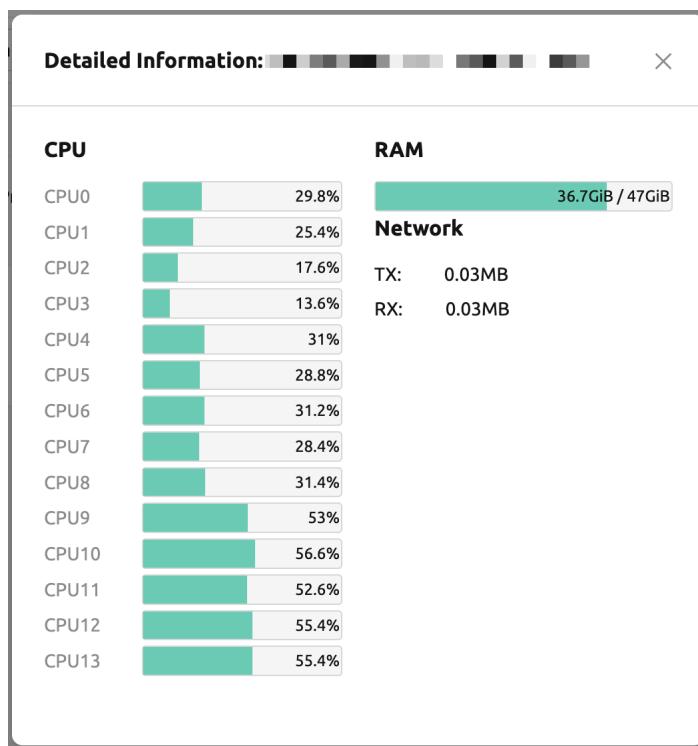
Superadmin의 경우 Resources 페이지의 Connected 탭에서는 현재 Backend.AI에 연결된 agent 워커 노드를 조회할 수 있습니다. 노드의 IP와 연결된 시간, 현재 사용중인 실제 자원 등을 조회할 수 있습니다. WebUI 앱에서는 별도로 agent 노드를 조작하는 기능은 제공하지 않습니다.

21.10.1 Agent 노드 조회

ID / Endpoint	Region	Architecture	Starts	Allocation	Utilization	Control
[REDACTED]	local	aarch64	Apr 7, 2025 10:59:41 AM Running 2d14:28:13	2/13 Core 2/47 GiB	15.4 % 4.3 %	CPU RAM

1-1 of 1 items < 1 > 20 / page

또한 agent 워커 노드의 자원에 대한 정확한 사용량을 Control 패널의 노트 아이콘을 클릭하여 확인할 수 있습니다.

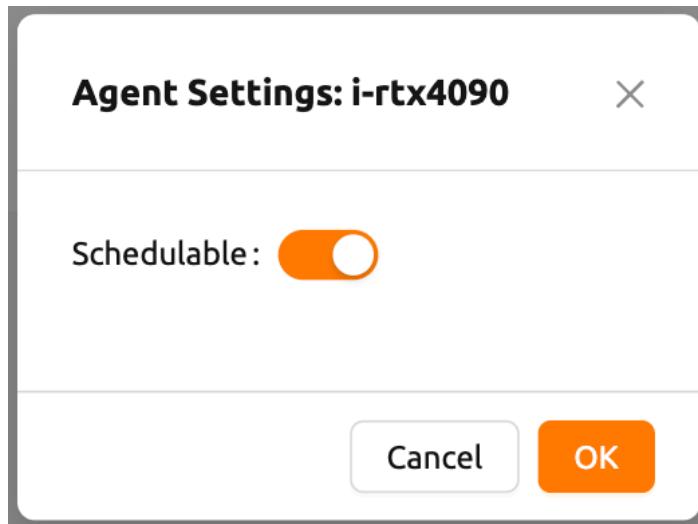


Terminated 탭으로 이동하면 한 번 연결되었다가 종료되거나 연결이 끊긴 에이전트의 정보를 확인할 수 있습니다. 노드 관리에 참고 자료로 활용할 수 있습니다.

ID / Endpoint	Region	Architecture	Starts	Allocation	Utilization	Disk %	Resource Group	Status	Schedulable	Control
No data										

21.10.2 Agent 노드의 스케줄링 가능 상태 설정하기

특정 Agent 서비스를 중단하지 않고 신규 세션이 스케줄링되는 것을 막고 싶을 수 있습니다. 이 경우, Agent의 Schedulable 상태를 비활성화할 수 있습니다. 기존에 해당 Agent에서 실행 중이던 연산 세션은 그대로 보존하면서 신규 세션 생성만 차단할 수 있습니다.



21.11 자원 그룹 관리

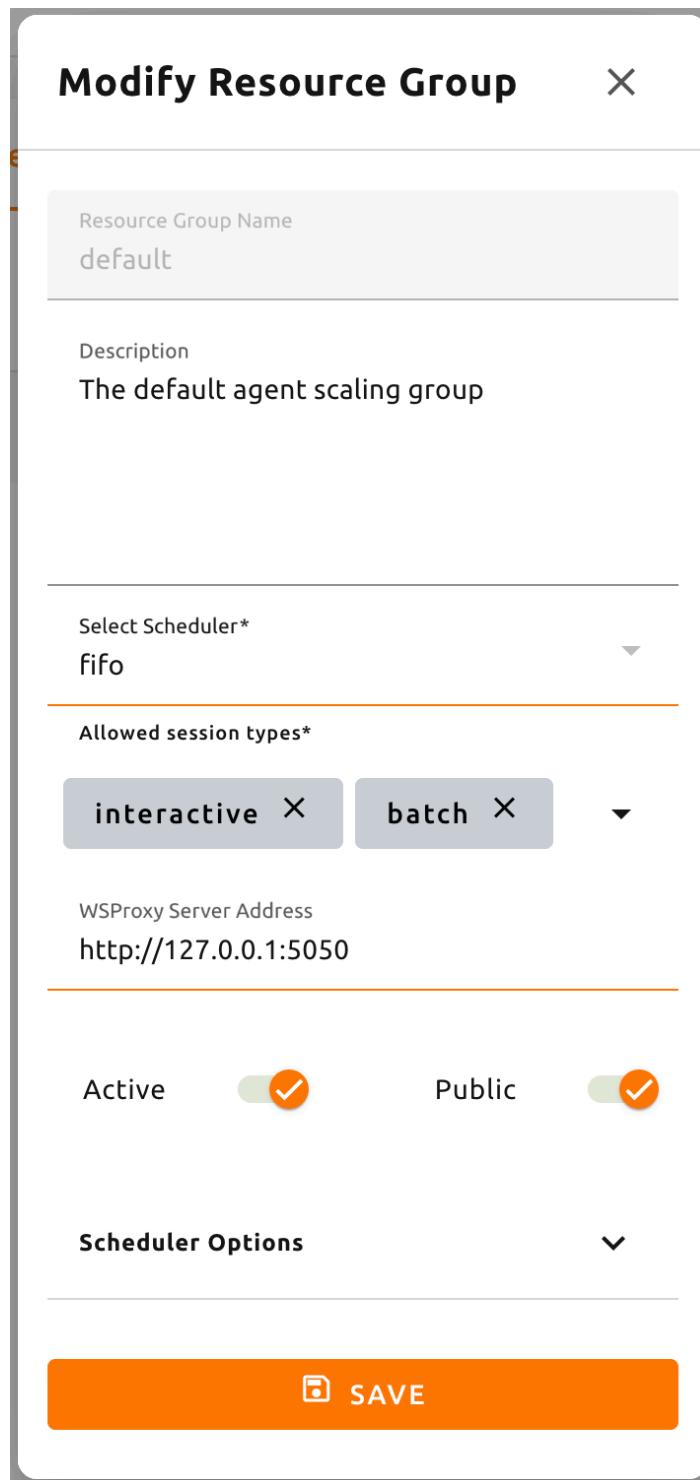
Agent는 자원 그룹이라는 단위로 묶일 수 있습니다. 예를 들어, V100 GPU를 탑재한 agent가 3대, P100 GPU를 탑재한 agent가 2대 있는데, 사용자에게 두 GPU 자원을 별도로 노출하고 싶을 경우 V100 agent 3대를 하나의 자원 그룹으로 묶고, 나머지 P100 agent 2대를 다른 자원 그룹으로 묶어서 관리할 수 있습니다.

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

Resource Group								
Resource Groups								
Name	Description	Active Status	Public Status	Driver	Scheduler	WSProxy Serve...	Control	+ ADD
default	The default age...	active	public	static	fifo	http://127.0.0.1...		

Control 열에서 '설정' 버튼을 클릭하여 자원 그룹을 편집할 수 있습니다. Select scheduler 필드에서 연산 세션 생성 스케줄링 방식을 선택할 수 있는데, 현재 지원하는 방식은 FIFO, LIFO, DRF 세 가지입니다. FIFO 와 LIFO 는 가장 처음 또는 가장 마지막에 작업 큐에 들어 온 연산 세션을 먼저 생성하는 방식으로 스케줄링 방식이고, DRF 는 Dominant Resource Fairness 의 약자로 사용자

별로 최대한 공평하게 자원 할당이 가능하도록 조절하여 스케줄링 하는 방식입니다. Active Status 를 꺼서 해당 자원 정책을 비활성화 할 수 있습니다.

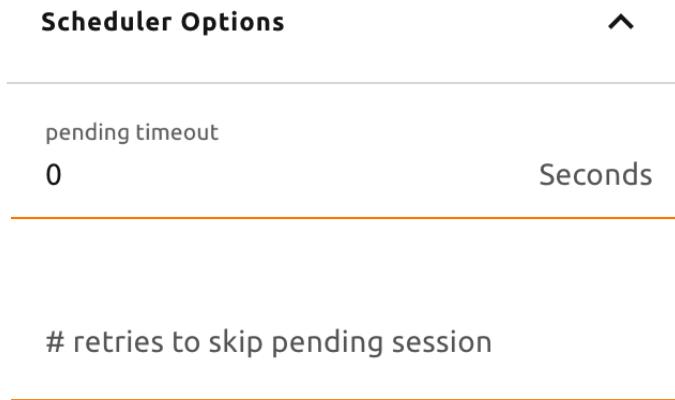


WSProxy Server Address에는 리소스 그룹에 속한 Agent 에서 사용할 WSProxy 서비스 주소를 설정할 수 있습니다. 이 필드에 URL 을 설정하면 WSProxy 에서 Jupyter 등의 앱 트래픽을 중계할 때 Manager 를 거치지 않고 Agent 를 통해 바로 사용자의 컨테이너에 접속하게 됩니다(v2 API). v2 API 를 사용하는 경우, 앱 서비스 사용 시 Manager 의 부하를 줄일 수 있습니다. 서비스를 배포할 때 효율성과 확장성도 증가합니다. 다만, WSProxy 에서 Agent 가 설치된 노드로 직접적인

네트워크 연결이 불가능한 경우에는, 이 필드를 빈 값으로 설정하여 Manager 를 거쳐 컨테이너로 트래픽을 중계하는 v1 API 를 사용할 수 있습니다.

리소스 그룹에서 스케줄러 관련 옵션(Scheduler Options)을 추가 설정할 수도 있습니다. 각 항목별 설정은 아래 내용을 참고하세요.

- Allowed session types: 사용자가 세션 타입을 설정할 수 있기 때문에, 자원 그룹에서도 특정 타입의 세션만 허용할 수 있음. 하나 이상의 세션 타입을 허용해야 하며, 허용되는 타입은 Interactive, Batch, Inference 임.
- Pending timeout: PENDING 상태에 머무는 시간이 Pending timeout 보다 긴 경우, 해당 세션을 취소함. 무한히 PENDING 상태에 머무르는 세션을 방지하고자 할 때 기준 시간을 설정할 수 있음. 0을 설정하면 Pending timeout이 적용되지 않음.
- The number of retries to skip pending session: PENDING 세션을 건너뛸 때까지의 스케줄러 재시도 횟수. 한 PENDING 세션이 그 뒤에 요청된 세션의 스케줄링을 무한히 막는 경우 (Head-of-line blocking, HOL)를 방지하기 위해 설정할 수 있음. 따로 설정하지 않는 경우에는 Etcd 에 설정된 글로벌 값 (num_retries_to_skip, 기본 3 회)을 사용함.



CREATE 버튼을 클릭하여 새로운 자원 정책을 생성할 수 있습니다. 다른 생성하기 기능과 마찬가지로, 자원 정책 이름은 키 값이기 때문에, 이미 존재하는 자원 정책 이름과 동일한 이름을 갖는 자원 정책 생성은 불가능합니다.

Create Resource Group X

Select Domain*
default

Resource Group Name*

Description

Select Scheduler*
Fifo

Allowed session types*

interactive X batch X

WSProxy Server Address

Active Public

Scheduler Options ^

pending timeout

retries to skip pending session

21.12 저장소

STORAGES 탭에서는 시스템에 마운트 된 볼륨을 조회할 수 있습니다. 주로 NFS라고 생각하시면 됩니다. 23.03 버전부터는 쿼타 관리를 지원하는 스토리지에 대해 사용자별/프로젝트별 쿼타 설정이 가능합니다. 이 기능을 사용함으로써, 관리자는 좀 더 손쉽게 사용자와 프로젝트용 폴더가 스토리지 내에서 얼마만큼 공간을 차지하고 있는지 확인하고 관리할 수 있습니다.

Agent	Storage	Resource Group		
Endpoint	Backend Type	Resources	Capabilities	Control
local:volume1 /Users/seungwonlee/Documents/backend.ai/vroot/local	Backend vfs	Usage 289% 162.02 GB / 994.66 GB	vfolder	

쿼타를 설정하기 위해서는 자원 페이지의 '저장소(Storage)' 탭을 누르고, 컨트롤 컬럼의 '설정' 버튼을 클릭합니다.

참고: 쿼타 설정은 해당 기능을 제공하는 저장소 (예: XFS, CephFS, NetApp, Purestorage, 등)에 한해서만 가능합니다. 저장소의 쿼타 설정 기능 제공 여부에 관계없이 쿼타 설정 페이지를 제공하지만, 내부적으로 쿼타 설정을 지원하지 않는 저장소에 대해서는 쿼타를 설정할 수 없습니다.

Resources / Storage Setting					
local:volume1					
Usage	16.3%	Used: 162.02 GB / Total: 994.66 GB	Endpoint	/Users/seungwonlee/Documents/backend.ai/vroot/local	Backend Type
					vfs
Quota Settings					
 This storage backend does not support quota.					

21.12.1 쿼타 설정 패널

쿼타 설정 페이지에는 각 패널의 제목에 대응하는 두 개의 패널이 있습니다.

Resources / Storage Setting

local:volume1

Usage	 94.8% Used: 10.9 TB / Total: 11.5 TB		
Endpoint	/mnt/lustre	Backend Type	exascaler
	Capabilities	vFolder	quota

Quota Settings

For User **For Project**

User: Search and select User

Quota Scope ID	Hard Limit (GB)	Usage (GB)	Control

Please select a project or user first from the selector at the top left.

• 스토리지 정보 패널

- 사용량: 선택된 저장소에서의 실제 사용량
- 엔드포인트: 선택된 저장소가 스토리지 노드에 마운트된 경로
- 백엔드 타입: 저장소의 타입
- 지원 기능: 선택된 저장소에서 지원하는 기능

• 시스템 설정 조회

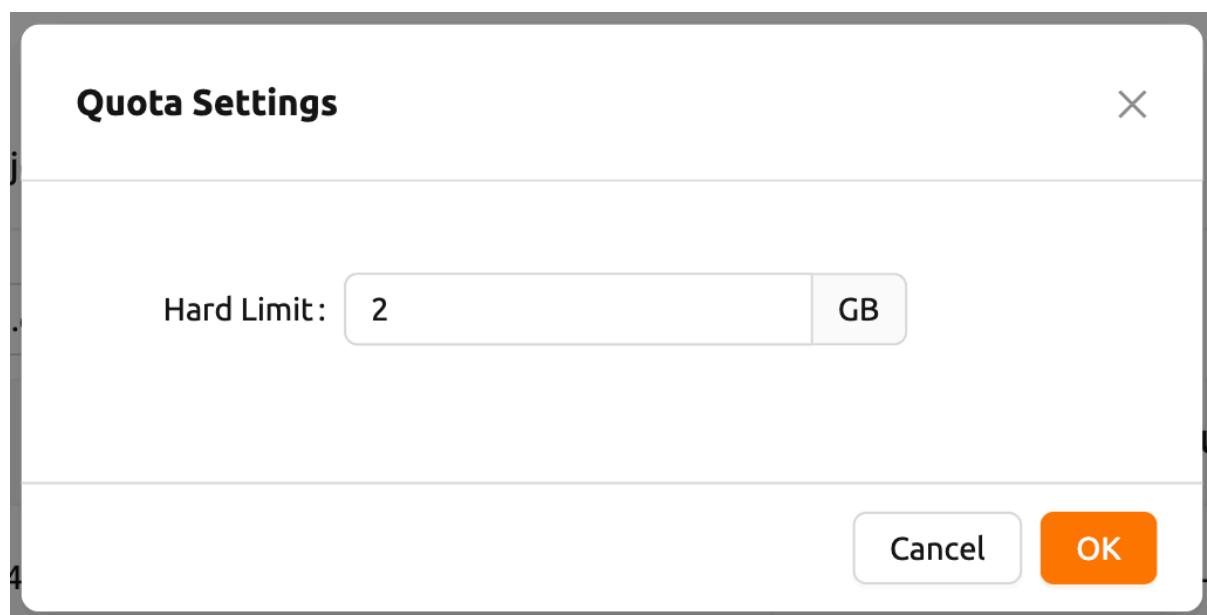
- 사용자 대상: 사용자별로 쿼타를 설정합니다.
- 프로젝트 대상: 프로젝트별로 쿼타(프로젝트-폴더)를 설정합니다.
- ID: 사용자나 프로젝트 ID에 대응합니다.
- Hard Limit (GB): 선택된 쿼타에서 현재 설정된 Hard Limit 쿼타를 표시합니다.
- 설정: Hard limit 과 같은 설정값을 수정하거나, 쿼타 설정값을 삭제하는 기능을 제공합니다.

21.12.2 사용자 쿼타 설정하기

Backend.AI에서는 사용자와 프로젝트 관리자에 의해 생성되는 두 가지 타입의 폴더가 있습니다. 이 섹션에서는, 사용자별로 현재 쿼타 설정값을 확인하고, 어떻게 값을 설정하는지 다룹니다. 우선, 쿼타 설정 패널의 활성화된 탭이 **For User** 인지 확인해주세요. 그 다음, 쿼타를 수정하고 확인할 대상인 사용자를 선택해주세요. 만일 쿼타가 이미 설정되어 있다면, 사용자 ID와 동일한 쿼타 ID 값이 테이블에 표시된 것을 확인하실 수 있습니다.

The screenshot shows the 'Storage Setting' section for 'local:volume1'. At the top, there is a usage summary: 'Usage' (94.8%, Used: 10.9 TB / Total: 11.5 TB), 'Endpoint' (/mnt/lustre), 'Backend Type' (exascaler), and 'Capabilities' (vfolder, quota). Below this, the 'Quota Settings' section is displayed. It has tabs for 'For User' (selected) and 'For Project'. A search bar for 'User' contains 'ad'. A dropdown menu lists 'admin@lblup.com' and 'admin@lblup.com'. A note at the bottom says 'Please select a project or user first from the selector at the top left.' There is also a small envelope icon.

물론, 쿼타를 수정하려고 할 경우, 컨트롤 컬럼의 수정 버튼을 선택하기만 하면 됩니다. **Edit** 버튼을 클릭하고 나면, 쿼타를 설정하는 작은 모달 창이 뜬 것을 확인하실 수 있습니다. 설정할 값을 정확히 입력 후, 반드시 **OK** 버튼을 클릭해주세요. 그렇지 않으면 변경 내용은 반영되지 않습니다.



21.12.3 프로젝트 쿼타 설정하기

프로젝트 폴더에 대해 쿼타를 설정하는 것은 사용자 쿼타 설정과 유사합니다. 프로젝트별로 적용되는 쿼타와 사용자별로 적용되는 쿼타의 차이점은, 프로젝트별로 적용되는 쿼타에서는 프로젝트가 소속되어 있는 도메인을 우선 선택하는 과정이 추가되었다는 것입니다. 그 이후 과정은 동일합니다. 하단 그림에서 볼 수 있듯이, 우선 도메인을 선택하고, 적용할 프로젝트를 선택하면 됩니다.

local:volume1

Usage: 94.8%
Used: 10.9 TB / Total: 11.5 TB

Endpoint: /mnt/lustre | Backend Type: exascaler | Capabilities: vfolder, quota

Quota Settings

For User | **For Project**

Domain: default | Project: Select Project

Quota Scope ID	Limit (GB)	Usage (GB)	Control
default	-	-	

Please select a project or user first from the selector at the top left.

local:volume1

Usage: 94.8%
Used: 10.9 TB / Total: 11.5 TB

Endpoint: /mnt/lustre | Backend Type: exascaler | Capabilities: vfolder, quota

Quota Settings

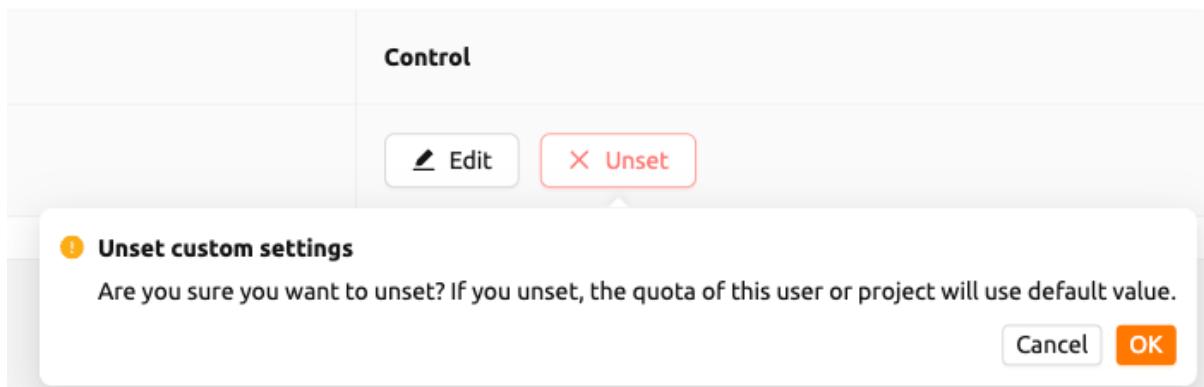
For User | **For Project**

Domain: default | Project: default

Quota Scope ID	General	Limit (GB)	Usage (GB)	Control
project:	default	-	-	<input type="button" value="Edit"/> <input type="button" value="Unset"/>
	test-project	-	-	
	model-store	-	-	

21.12.4 쿼타 해제하기

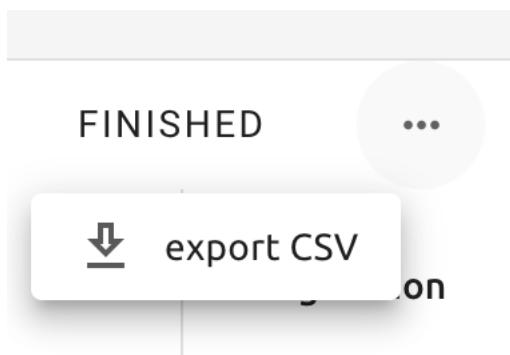
쿼타를 해제하는 기능도 제공합니다. 쿼타 설정값을 삭제한 뒤에는, 해당 프로젝트/사용자에 적용된 쿼타는 사용자 또는 프로젝트에서 기본적으로 적용된 쿼타를 따라가게 됩니다. 이 값은 WebUI에서 따로 설정하는 사용자 인터페이스를 지원하지 않습니다. 기본 쿼타 설정값을 변경하려면, 관리자 전용 페이지에 접근해서 해당 값을 변경해야 합니다. 컨트롤 컬럼에 있는 **Unset** 버튼을 클릭하면, 작은 스낵바 메시지 창이 뜨고, 현재 쿼타 세팅을 정말로 삭제할 것인지 확인하게 됩니다. 여기서 **OK** 버튼을 클릭하게 되면, 쿼타 설정값이 삭제되고, 자동으로 쿼타 설정값은 쿼타의 타입(사용자 / 프로젝트)에 대응하는 값으로 적용되게 됩니다.



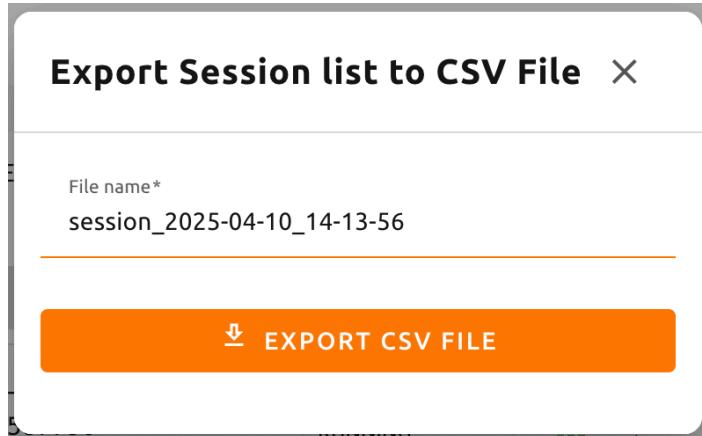
참고: 만일 사용자 별/프로젝트 별 설정값이 없을 경우, 사용자/프로젝트 별 자원 정책 내 대응하는 값들이 기본 값으로 설정되게 됩니다. 예를 들어 쿼타에서 hard limit 값이 설정되어 있지 않다면, 자원 정책 내 max_vfolder_size 값이 기본 값으로 쓰이게 됩니다.

21.13 세션 자원 다운로드

세션 페이지에는 관리자를 위한 추가 기능이 있습니다. FINISHED 탭 우측을 보면 … 으로 표시된 메뉴가 있습니다. 이 메뉴를 클릭하면 export CSV라는 하위 메뉴가 나옵니다.



이 메뉴를 클릭하면 현재까지 생성된 연산 세션의 정보를 CSV 형태로 다운로드 받을 수 있습니다. 다음과 같은ダイ얼로그가 열린 후, (필요한 경우) 적당한 파일 이름을 입력하고 EXPORT 버튼을 클릭하십시오. 파일 이름은 최대 255 자까지만 입력 가능한 점에 유의하십시오. 곧 CSV 파일 하나가 다운로드 될 것입니다.



21.14 시스템 설정 조회

Configuration 페이지에서 Backend.AI에 설정된 주요 설정값을 조회할 수 있습니다. 현재는 몇 가지 변경 기능 및 설정 조회 기능을 제공하고 있습니다.

자동 설치 및 업데이트 규칙을 Digest, Tag, None 중에서 선택할 수 있습니다. Digest 는 이미지에 대한 checksum 과 같은 것으로, image의 무결성을 검증하고, 중복된 레이어를 재사용 함으로써 이미지 다운로드의 효율성을 높이는데에 사용됩니다. Tag 는 개발용 옵션에만 사용할 수 있는데, 태그는 이미지의 무결성을 보장하지 않기 때문입니다.

경고: 각 규칙에 대해 완전히 이해하고 있지 않는 한 선택된 규칙을 변경하지 마십시오.

스케일링과 플러그인, 엔터프라이즈 기능에 대한 설정도 변경할 수 있습니다.

Overlay Network
Configuration to use when creating overlay networks.

Scheduler
Configuration per job scheduler.

Plugins

The settings below are automatically applied depending on the installation environment and status.

Open Source CUDA GPU support
 NVIDIA CUDA GPU support. Requires Backend.AI CUDA Plugin.

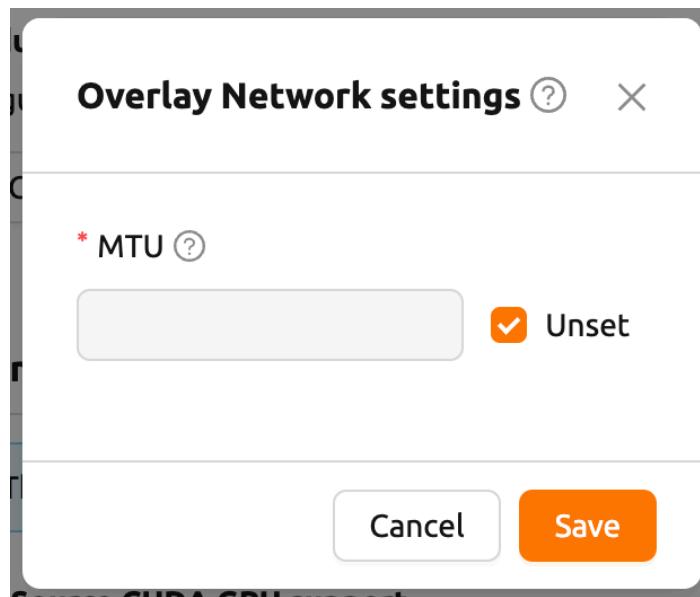
ROCM GPU support
 AMD ROCm GPU support. Requires Backend.AI ROCm Plugin.

Enterprise Features

The settings below are automatically applied depending on the installation environment and status.

Fractional GPU
 Use Fractional GPU feature with GPU virtualization.
Requires Backend.AI Virtual CUDA API Layer Plugin.

사용자가 멀티 노드 클러스터 세션(Backend.AI 20.09부터 지원)을 생성할 경우, Backend.AI는 노드간 사설 통신을 지원하기 위해 동적으로 오버레이 네트워크(overlay network)를 생성합니다. 이 오버레이 네트워크에서 사용하는 Maximum Transmission Unit (MTU) 값을 설정할 수 있습니다. 다만, 이 값이 네트워크 성능을 향상시키는 것이 확실할 때만 설정하십시오.



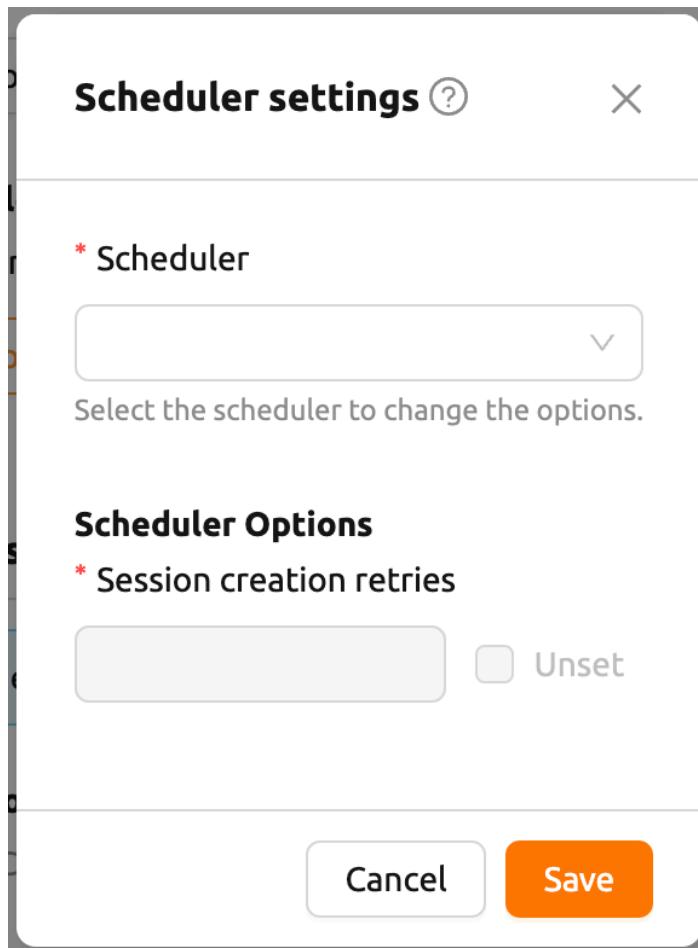
더 보기:

Backend.AI 클러스터 세션에 대해 더 자세한 정보를 확인하시려면, [Backend.AI 클러스터 연산 세션](#) 섹션을 참고하십시오.

Scheduler의 설정(CONFIG) 버튼을 누르면 스케줄러 별 설정을 할 수 있습니다. 스케줄러 별

설정값은 자원 그룹의 스케줄러 설정값이 없을 때 사용하는 기본 값을 의미합니다. 자원 그룹에 설정한 값이 있을 경우, 이 값은 무시됩니다.

현재 지원하는 스케줄링 방법에는 FIFO, LIFO, DRF가 있습니다. 각 스케줄링 방법은 위의 [스케줄링 방법](#)과 동일합니다. 스케줄러 옵션에는 세션 재시도 횟수가 있습니다. 세션 재시도 횟수란 세션 생성이 실패한 경우 세션 생성을 재시도 하는 횟수를 말합니다. 만약 지정한 횟수 안에 세션 생성을 하지 못하는 경우 해당 요청을 무시하고 다음 요청을 처리합니다. 현재는 스케줄러가 FIFO일 때만 가능합니다.



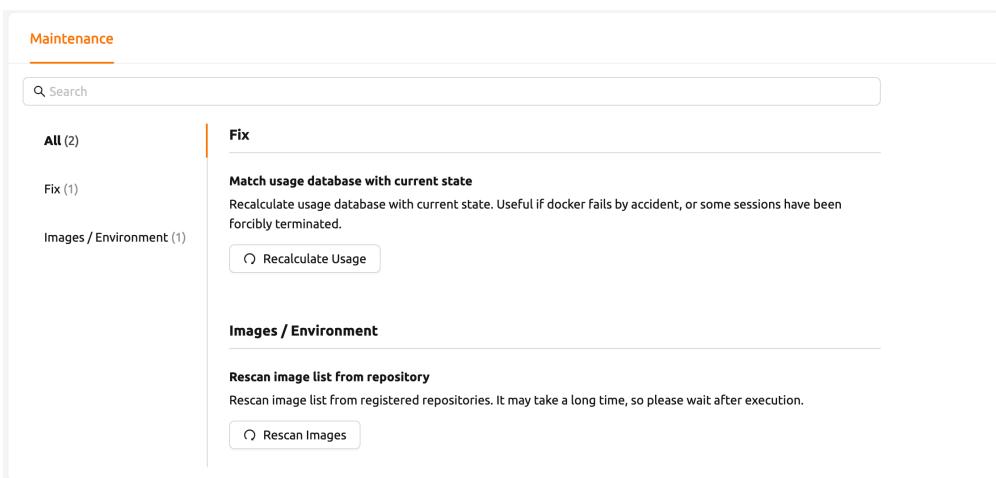
참고: 향후 CLI에서 지원하는 다양한 설정 변경 기능을 GUI에도 계속 추가할 예정입니다.

참고: 시스템 설정값은 기본 설정입니다. 자원 그룹에서 특정 값이 설정된 경우, 시스템 설정값이 아닌 자원 그룹 설정값을 적용합니다.

21.15 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.



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

21.16 상세 정보

Information 페이지에서는 여러가지 자세한 정보와 각 기능의 상태를 볼 수 있습니다. 매니저 버전과 API 버전을 보려면 Core 패널을 확인하십시오. Backend.AI를 구성하는 각 컴포넌트의 호환 가능 여부를 보려면 Component 패널을 확인하십시오.

참고: 이 페이지는 현재 정보를 보여주기 위한 것입니다.

Information /

Core

Manager version	Backend.AI 25.5.2 Installation 25.5.2	API version	v8.20240915
-----------------	--	-------------	-------------

Security

Default administrator account changed SHOULD change the default account / password for production use.		Uses SSL HTTPS with proper SSL setup helps system security.	
--	--	---	--

Component

Docker version Docker version on the cluster		PostgreSQL version Database system for Backend.AI	
ETCD version Settings registry for Backend.AI		Redis version Cache / temporary storage for Backend.AI. Also uses as asynchronous communication along agents.	

License

Validation Validates the license currently installed on the system.		License Type The type of license currently used in the system. For fixed licenses, execution rights are given only to predetermined hardware. In the case of variable licenses, execution rights are granted for cluster hardware within the limits of the agreement.	
Licensee Information about the organization or buyer licensed for use.		License Key Authentication key for permission to use. Please keep it secure.	
Expiration The validity period of the license. If the period is over, the cluster software will be stopped.			

CHAPTER 22

FAQ 및 문제 해결

22.1 사용자 문제 해결 가이드

22.1.1 연산 세션 리스트가 나타나지 않습니다

간헐적인 네트워크 문제나 기타 다양한 원인으로 인해 연산 세션 리스트가 정상적으로 표시되지 않을 수 있습니다. 대부분의 경우, 브라우저를 갱신하면 연산 세션이 정상적으로 보입니다.

- 웹 기반 Web-UI: 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 Web-UI: Ctrl-R 단축키를 클릭하여 앱 페이지를 갱신할 수 있습니다.

22.1.2 갑자기 로그인이 안 됩니다

간혹 브라우저의 쿠키 문제 및 캐시된 데이터로 인해 로그인이 되지 않는 경우가 있습니다. 브라우저의 시크릿 모드에서 로그인을 시도 해보십시오. 만약 로그인이 된다면, 브라우저의 쿠키 및 애플리케이션 데이터를 삭제한 후 다시 로그인 해보시기 바랍니다.

22.1.3 apt 패키지는 어떻게 설치하나요?

보안 문제로 연산 세션 내부에서 사용자는 root 계정으로 전환하거나 sudo 를 필요로 하는 명령을 실행할 수 없습니다. 따라서, apt, yum 과 같이 sudo 권한을 요구하는 패키지 매니저를 통 해서는 일반적으로 설치가 어렵습니다. 반드시 필요한 경우라면 관리자에게 sudo 권한을 허용해줄 것을 요청할 수 있습니다.

또는, OS 패키지 설치를 위해 Homebrew 패키지 매니저를 사용할 수도 있습니다. 자동 마운트 폴더로 Homebrew 패키지를 설치하고 사용하는 [가이드](#)를 참고 하십시오.

22.1.4 pip 패키지를 설치하고 싶어요

Pip 패키지를 설치하면 기본적으로 사용자의 `~/local` 디렉토리에 저장됩니다. 따라서, `local` 자동마운트 폴더를 생성해두면, 연산 세션이 종료되더라도 계속 패키지를 보관해두고 다음 세션에서 재사용 할 수 있습니다. 다음과 같이 패키지를 설치하면 됩니다.

```
$ pip install aiohttp
```

보다 자세한 사항은 자동 마운트 폴더를 사용해 Python 패키지 설치하는 방법 [가이드](#)를 참고하십시오.

22.1.5 연산 세션을 생성했는데 Jupyter Notebook 을 열 수가 없습니다

Jupyter 패키지를 직접 설치한 경우, 연산 세션이 기본으로 제공하는 jupyter 패키지와 충돌할 가능성이 있습니다. 특히, `~/local` 자동 마운트 폴더를 생성해서 사용하고 계시다면, 수동 설치한 jupyter 패키지가 모든 연산 세션에서 호환 문제를 일으킬 수 있습니다. 이 경우, `local` 데이터 폴더를 삭제한 후 다시 Jupyter Notebook을 시작해보십시오.

22.1.6 페이지가 이상하게 표시됩니다

Backend.AI Web-UI는 최신 JavaScript 와 브라우저의 기능을 적극 활용하고 있습니다. 가급적 최신 브라우저를 사용하십시오. 특히 Chrome 에서 가장 안정적인 레이아웃을 보입니다.

22.1.7 SFTP 연결이 끊기는 경우

Web-UI 앱을 통한 SFTP 연결은 Web-UI 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, Web-UI 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.

22.2 관리자 문제 해결 가이드

22.2.1 사용자가 Jupyter Notebook 등의 앱을 띄울 수 없는 경우

WSProxy 서비스 접근에 문제가 있을 수 있습니다. WSProxy 서비스를 시작/중단/재시작 하는 항목을 참고하여 서비스를 중단하고 다시 시작해 봅시다.

22.2.2 표시되는 자원 양이 실제 할당된 양과 다릅니다

가끔 네트워크 연결이 튕거나 Docker 데몬의 컨테이너 관리 지연 등의 사유로 Backend.AI 가 인식하는 자원의 할당양과 실제 컨테이너가 점유하고 있는 자원의 양이 달라질 수 있습니다. 이런 경우에는 다음 과정을 따라 하십시오.

- 어드민 계정으로 로그인
- Maintenance 페이지 방문.
- RECALCULATE USAGE 버튼을 클릭하여 수동으로 자원 할당량 조정.

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

참고: 이 기능은 super admin만 사용할 수 있습니다.

사설 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 받아옵니다.

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 23

Appendix

23.1 GPU 가상화를 통한 컨테이너 별 GPU 분할 할당

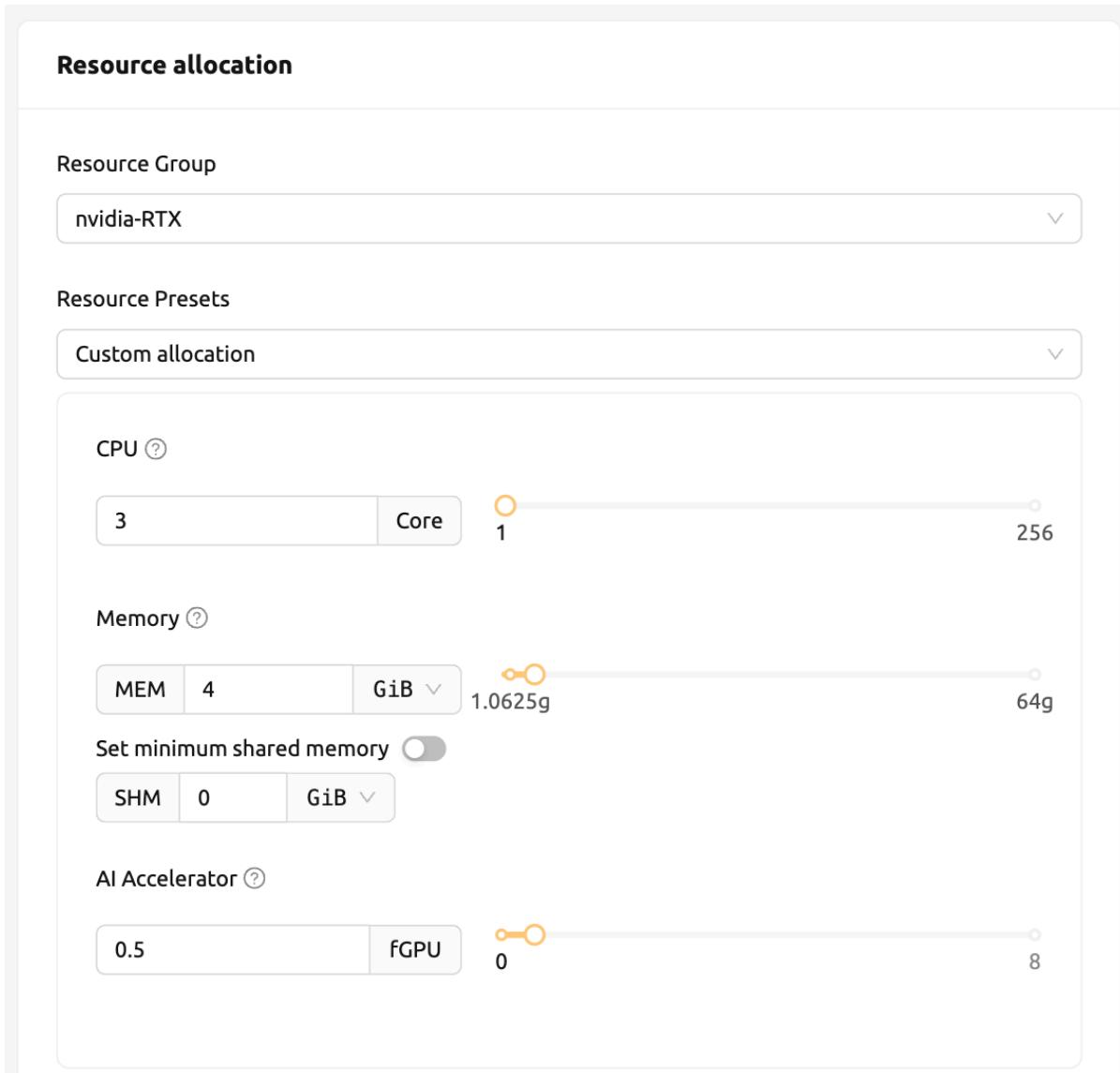
Backend.AI 는 하나의 물리 GPU 를 여러 개로 분할해서 여러 사용자가 나누어 사용할 수 있는 가상화 기술을 지원하고 있습니다. 따라서, GPU 연산 소요가 크지 않은 작업을 수행하고자 할 경우에는 GPU 의 일부만 할당하여 연산 세션을 생성할 수 있습니다. 1 fGPU 가 실제로 할당하는 GPU 자원의 양은 관리자 설정에 따라 시스템 별로 다양할 수 있습니다. 예를 들어, 관리자가 하나의 GPU 를 다섯 조각으로 분할 설정한 경우, 5 fGPU 가 1 물리 GPU, 또는 1 fGPU 가 0.2 물리 GPU 를 뜻합니다. 이 때 1 fGPU 를 설정하여 연산 세션을 생성하면, 그 세션에서는 0.2 물리 GPU 에 해당하는 SM(streaming multiprocessor) 과 GPU 메모리를 활용할 수 있습니다.

이번에는 GPU 를 일부만 할당하여 연산 세션을 생성한 후 연산 컨테이너 내부에서 인식하는 GPU 가 정말 물리 GPU 의 일부분인지 확인 해보도록 하겠습니다.

먼저 호스트 노드에 장착되어 있는 물리 GPU 의 종류와 메모리 용량 등의 정보를 확인 해보겠습니다. 이 가이드를 작성하면서 사용한 GPU 노드에는 다음과 같이 8 GB 메모리의 GPU 가 장착되어 있습니다. 그리고 관리자 설정을 통해 1 fGPU 를 0.5 개의 물리 GPU(또는 1 개의 물리 GPU 가 2 fGPU) 에 해당하는 양으로 설정하였습니다.

```
/home/devops/backend.ai-master/console main adrysn@manager
> nvidia-smi
Fri Sep 11 19:31:25 2020
+-----+
| NVIDIA-SMI 440.33.01      Driver Version: 440.33.01      CUDA Version: 10.2 |
+-----+
| GPU  Name      Persistence-M| Bus-Id      Disp.A  | Volatile Uncorr. ECC | | | | | |
| Fan  Temp     Perf  Pwr:Usage/Cap| Memory-Usage | GPU-Util  Compute M. |
|====|=====|=====|=====|=====|=====|=====|=====|
|  0  GeForce GTX 1070     Off  | 00000000:01:00.0 Off |          N/A |
| 47%   56C    P0    40W / 151W |      10MiB /  8119MiB |     0%       Default |
+-----+
+-----+
| Processes:                               GPU Memory |
| GPU  PID  Type  Process name             Usage      |
|====|====|====|====|
| No running processes found               |
+-----+
```

이제 Sessions 페이지로 이동하여 다음과 같이 0.5 개의 fGPU 를 할당하여 연산 세션을 생성해봅시다:



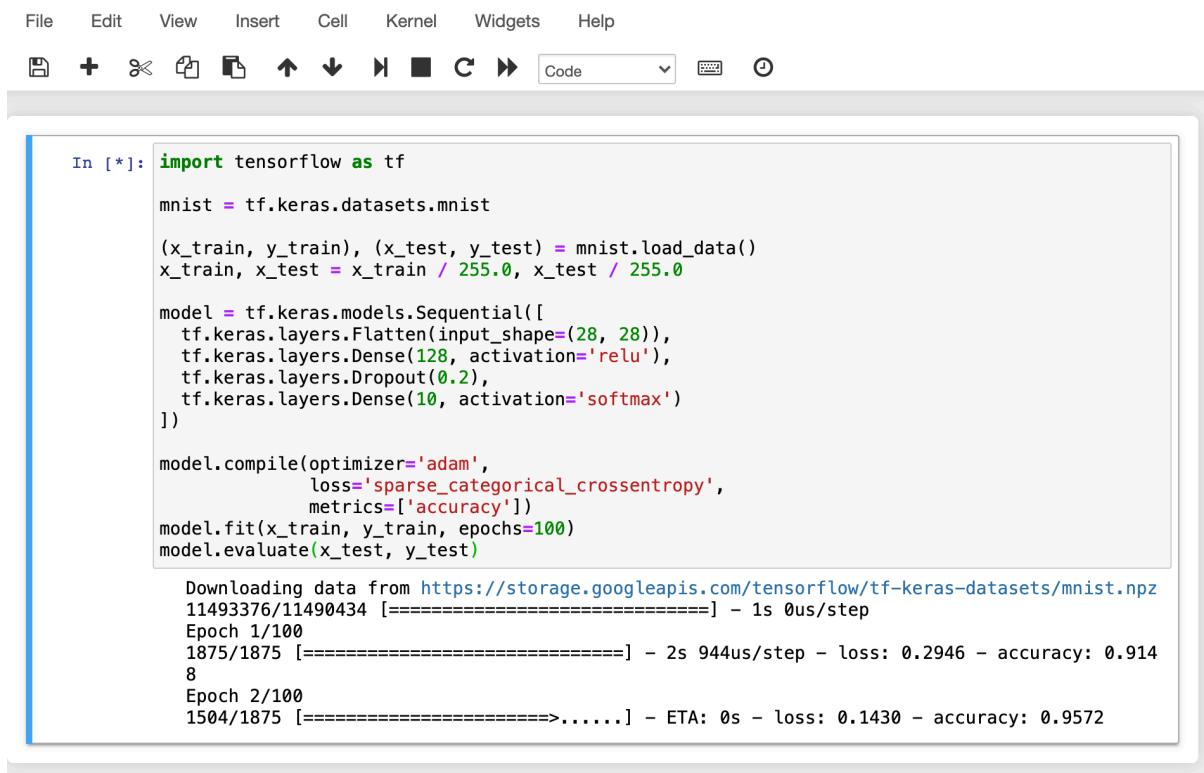
연산 세션 리스트의 AI Accelerator 열에서 0.5 의 fGPU 가 할당된 것을 확인할 수 있습니다.

Sessions						<input type="button" value="C"/>	<input type="button" value="Start Session"/>	
<input type="button" value="All"/> 49		<input type="button" value="Interactive"/> 5	<input type="button" value="Batch"/> 11	<input type="button" value="Inference"/> 33	<input type="button" value="Upload Sessions"/>			
<input type="button" value="Running"/>		<input type="button" value="Finished"/>	<input type="button" value="Session Name"/>	<input type="button" value="Search"/>	<input type="button" value=""/>			
<input type="checkbox"/>	Session Name		Status	Utilization	AI Accelerator	CPU	Memory	Elapsed Time
<input type="checkbox"/>	test-session-for-docs		RUNNING	0% 3% 0%	0.5 fGPU	3	4 GiB	00:04:13

이제 컨테이너에 직접 연결하여 할당 된 GPU 메모리가 실제로 0.5 단위 (~ 2GB)와 동일한지 확인하겠습니다. 웹 터미널을 띄웁니다. 터미널이 나타나면 `nvidia-smi` 명령을 실행합니다. 다음 그림에서 볼 수 있듯이 약 2GB의 GPU 메모리가 할당 되었음을 알 수 있습니다. 이는 물리적 GPU가 실제로 네 부분으로 나뉘어 이 연산 세션에 할당되었음을 보여줍니다. 이는 PCI 패스스루 (passthrough)와 같은 방식으로는 불가능합니다.

```
work@main1[AgWAvDC1-session]:~$ nvidia-smi
Thu Mar 28 11:19:20 2024
+-----+
| NVIDIA-SMI 530.30.02      Driver Version: 530.30.02    CUDA Version: 12.1 |
+-----+
| GPU  Name        Persistence-M | Bus-Id     Disp.A  Volatile Uncorr. ECC | | |
| Fan  Temp  Perf  Pwr:Usage/Cap| Memory-Usage | GPU-Util  Compute M. |
|                               |             |            | MIG M. |
+-----+
| 0  CUDA GPU     On           00000000:3F:00.0 Off   0%          Off |
| 33% 27C   P8      13W / 260W  48MiB / 2025MiB |          0%  Default |
|                                         |             N/A        |
+-----+
+-----+
| Processes:                               GPU Memory |
| GPU  GI  CI      PID  Type  Process name        Usage  |
| ID  ID
+-----+
work@main1[AgWAvDC1-session]:~$
```

이번에는 Jupyter Notebook 을 띄워서 간단한 ML 학습 코드를 실행해보겠습니다.



```
In [*]: import tensorflow as tf
mnist = tf.keras.datasets.mnist
(x_train, y_train), (x_test, y_test) = mnist.load_data()
x_train, x_test = x_train / 255.0, x_test / 255.0

model = tf.keras.models.Sequential([
    tf.keras.layers.Flatten(input_shape=(28, 28)),
    tf.keras.layers.Dense(128, activation='relu'),
    tf.keras.layers.Dropout(0.2),
    tf.keras.layers.Dense(10, activation='softmax')
])

model.compile(optimizer='adam',
              loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])
model.fit(x_train, y_train, epochs=100)
model.evaluate(x_test, y_test)

Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.npz
11493376/11490434 [=====] - 1s 0us/step
Epoch 1/100
1875/1875 [=====] - 2s 944us/step - loss: 0.2946 - accuracy: 0.914
8
Epoch 2/100
1504/1875 [=====>.....] - ETA: 0s - loss: 0.1430 - accuracy: 0.9572
```

학습이 진행되는 동안 GPU 호스트 노드의 쉘로 접속해서 `nvidia-smi` 명령을 실행합니다. 다음과 같이 하나의 GPU 사용 프로세스가 있고 이 프로세스는 물리 GPU의 약 25% 에 해당하는 자원을 점유중임을 알 수 있습니다. (GPU 점유량은 학습 코드와 GPU 모델에 따라 크게 다를 수 있습니다.)

```
/home/devops/backend.ai-master/console main adrdsn@manager
> nvidia-smi
Fri Sep 11 19:50:11 2020
+-----+
| NVIDIA-SMI 440.33.01      Driver Version: 440.33.01      CUDA Version: 10.2 |
+-----+
| GPU  Name      Persistence-M| Bus-Id      Disp.A  | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap| Memory-Usage | GPU-Util  Compute M. |
|=====+=====+=====+=====+=====+=====+=====+=====|
|  0  GeForce GTX 1070     Off  | 00000000:01:00.0 Off |                  N/A |
| 47%   57C    P2    40W / 151W |   2025MiB /  8119MiB |    25%     Default |
+-----+
+-----+
| Processes:                               GPU Memory |
| GPU  PID  Type  Process name          Usage        |
| =====+=====+=====+=====
|  0    29181  C    /usr/bin/python3      2015MiB |
+-----+
```

또는, 아까 띄워둔 웹 터미널에서 `nvidia-smi` 명령을 내려 컨테이너 내부에서 인식하는 GPU 사용 내역을 조회해보는 것도 가능합니다.

23.2 GUI를 통한 자원 모니터링 및 스케줄링 자동화

Backend.AI 서버는 자체 개발한 작업 스케줄러를 내장하고 있습니다. 자동으로 모든 워커 (worker) 노드의 자원 상태를 확인하여 사용자의 자원 요청에 맞는 워커로 연산 세션 생성 요청을 위임 합니다. 또한, 자원이 부족할 경우에는 일단 작업 큐에 사용자의 연산 세션 생성 요청을 대기 (pending) 시키고 나중에 자원이 다시 가용 상태가 되면 대기 요청을 활성화 해서 연산 세션 생성 작업을 수행하게 됩니다.

사용자 Web-UI에서 간단한 방법으로 작업 스케줄러의 작동을 확인할 수 있습니다. GPU 호스트가 최대 2 단위의 fGPU를 할당 할 수 있는 경우, 1 단위의 fGPU 할당을 요청하는 3 개의 연산 세션을 동시에 생성해 보겠습니다. 세션 시작 대화 상자의 사용자 지정 할당 패널에는 GPU 및 세션 슬라이더가 있습니다. Sessions에서 1보다 큰 값을 지정하고 LAUNCH 버튼을 클릭하면 여러 개의 세션이 동시에 생성됩니다. GPU와 Sessions을 각각 1과 3으로 설정하겠습니다. 이는 fGPU가 2 단위밖에 없는 상황에서 총 3 단위의 fGPU를 요청하는 상황입니다.

Resource allocation

Resource Group

nvidia-RTX

Resource Presets

Custom allocation

CPU ②

2 Core 1 256

Memory ②

MEM 4 GiB 1.0625g 64g

Set minimum shared memory

SHM 0 GiB

AI Accelerator ②

1.0 fGPU 0 8

Sessions ②

3 # 1 3

잠시 기다리면 세 개의 연산 세션이 나타납니다. 상태 패널을 자세히 살펴보면 세 개의 연산 세션 중 두 개는 RUNNING 상태에 있지만 다른 연산 세션은 PENDING 상태로 남아 있음을 알 수 있습니다. 이 PENDING 세션은 작업 대기열에만 등록되며 GPU 자원 부족으로 인해 실제로 컨테이너 할당을 받지 못했습니다.

<input type="checkbox"/> Session Name	Status	Utilization	AI Accelerator	CPU	Memory	Elapsed Time
<input type="checkbox"/> test-for-multi-QLO-1	PENDING	0% 3% 0%	1 fGPU	2	4 GiB	00:02:57
<input type="checkbox"/> test-for-multi-FSp-2	RUNNING	0% 3% 0%	1 fGPU	2	4 GiB	00:02:57
<input type="checkbox"/> test-for-multi-Omk-0	RUNNING	0% 3% 0%	1 fGPU	2	4 GiB	00:02:57

이제 RUNNING 상태의 세션 두 개 중 하나를 삭제 해보겠습니다. 그러면 PENDING 상태의 연산 세션은 곧 작업 스케줄러에 의해 자원을 할당 받고 RUNNING 상태로 변환되는 것을 볼 수 있습니다. 이처럼, 작업 스케줄러는 작업 큐를 활용해 사용자의 연산 세션 요청을 간직하고 있다가 자원이 가용해질 때 자동으로 요청을 처리하게 됩니다.

<input type="checkbox"/> Session Name	Status	Utilization	AI Accelerator	CPU	Memory	Elapsed Time
<input type="checkbox"/> test-for-multi-QLO-1	RUNNING	0% 3% 0%	1 fGPU	2	4 GiB	00:04:34
<input type="checkbox"/> test-for-multi-Omk-0	RUNNING	0% 3% 0%	1 fGPU	2	4 GiB	00:04:34

23.3 Multi-version 머신러닝 컨테이너 지원

Backend.AI 는 다양한 ML 및 HPC 커널 이미지를 사전 빌드하여 제공 합니다. 따라서, 사용자는 스스로 패키지 설치를 굳이 하지 않더라도 주요 라이브러리 및 패키지를 즉시 활용할 수 있습니다. 여기서는 다종 ML 라이브러리의 여러 버전을 즉시 활용하는 예제를 진행합니다.

Sessions 페이지로 이동하여 연산 세션 생성 디아일로그를 엽니다. Backend.AI에서는 설치 환경에 따른 다양한 커널 이미지를 제공합니다.

The screenshot shows the 'Environments' section of the Backend.AI WebUI. A search bar at the top contains the text 'TensorFlow'. Below the search bar, a list of environments is displayed, each with a small icon and a name. Some environments have additional status indicators like 'NVIDIA GPU Cloud' or 'testing'.

Environment	Status
PyTorch (Fashion-How)	
PyTorch (NGC)	NVIDIA GPU Cloud
PyTorch (NGC)	testing
PyTorch (NGC)	NVIDIA GPU Cloud
PyTorch (NGC)	customized
PyTorch (NGC)	NVIDIA GPU Cloud
Swift For TensorFlow	testing
TensorFlow	
TensorFlow	testing
TensorFlow (Fashion-How)	

여기서는 TensorFlow 2.3 환경을 선택하고 세션을 생성해보았습니다.

Start new session

Session Type [Edit](#)

SessionType: **interactive**

Environments [Edit](#)

Project: **default**

Image: Python Tensorflow | 2.3 | x86_64 | **Python 3.6** | GPU: CUDA10.1 |

Resource allocation [Edit](#)

Resource Group: **nvidia-RTX**

Resource Allocation Per Container: **minimum-required** 1 Core 1.063 GiB (SHM: 0.06GiB)

Agent: **auto** Number of container: **1**

Cluster mode: **Single Node**

Total Allocation

1 Core 1.063 GiB (SHM: 0.06GiB)

Data & Storage [Edit](#)

No storage folder is mounted

Network [Edit](#)

Preopen Ports: None

Reset [Previous](#) [Launch](#)

생성된 세션의 웹 터미널을 열고 다음 Python 명령을 실행합니다. TensorFlow 2.3 버전이 실제 설치되어 있음을 확인할 수 있습니다.

```
work@main1:~$ clear
work@main1:~$ python -c "import tensorflow as tf; print(tf.__version__)"
2024-03-28 11:40:54.873813: I tensorflow/stream_executor/platform/default/dso_loader.cc:48] Successfully opened dynamic library libcudart.so.10.1
2.3.2
work@main1:~$
```

이번에는 TensorFlow 1.15 환경을 선택해서 연산 세션을 생성합니다. (자원이 부족한 경우 이전

세션은 삭제합니다)

Start new session

Session Type Edit

SessionType: interactive

Environments Edit

Project: default

Image: Python Tensorflow | 1.15 | x86_64 | Python 3.6 GPU: CUDA10

Resource allocation Edit

Resource Group: nvidia-H100

Resource Allocation Per Container: minimum-required 1 Core 1.063 GiB (SHM: 0.06GiB)

Agent: auto Number of container: 1

Cluster mode: Single Node

Total Allocation

1 Core 1.063 GiB (SHM: 0.06GiB)

Data & Storage Edit

No storage folder is mounted

Network Edit

Preopen Ports: None

Reset Previous Launch

생성된 세션의 웹 터미널을 열고 이전과 동일한 Python 명령을 실행합니다. TensorFlow 1.15(.4) 버전이 실제 설치되어 있음을 확인할 수 있습니다.

```
work@main1:~$ python -c "import tensorflow as tf; print(tf.__version__)"
1.15.4
work@main1:~$
```

마지막으로 PyTorch 1.7 버전을 이용해서 연산 세션을 생성합니다.

Start new session

Session Type Edit

SessionType: interactive

Environments Edit

Project: default

Image: Python Tensorflow | 1.7 | x86_64 | Python 3.6 GPU: CUDA10

Resource allocation Edit

Resource Group: nvidia-H100

Resource Allocation Per Container: 0-CPUmedium 8 Core 16 GiB (SHM: 1.00GiB)

Agent: auto Number of container: 1

Cluster mode: Single Node

Total Allocation

8 Core 16 GiB (SHM: 1.00GiB)

Data & Storage Edit

No storage folder is mounted

Network Edit

Preopen Ports: None

Reset Previous Launch

생성된 세션의 웹 터미널을 열고 다음 Python 명령을 실행합니다. PyTorch 1.8 버전이 실제 설치

되어 있음을 확인할 수 있습니다.

```
work@main1:~$ python -c "import torch; print(torch.__version__)"
1.7.1+cu110
work@main1:~$
```

이처럼, Backend.AI 를 통해 TensorFlow, PyTorch 등 주요 라이브러리의 다양한 버전을 불필요한 설치 노력 없이 활용할 수 있습니다.

23.4 실행 중인 연산 세션을 새로운 사용자 이미지로 변환하는 방법

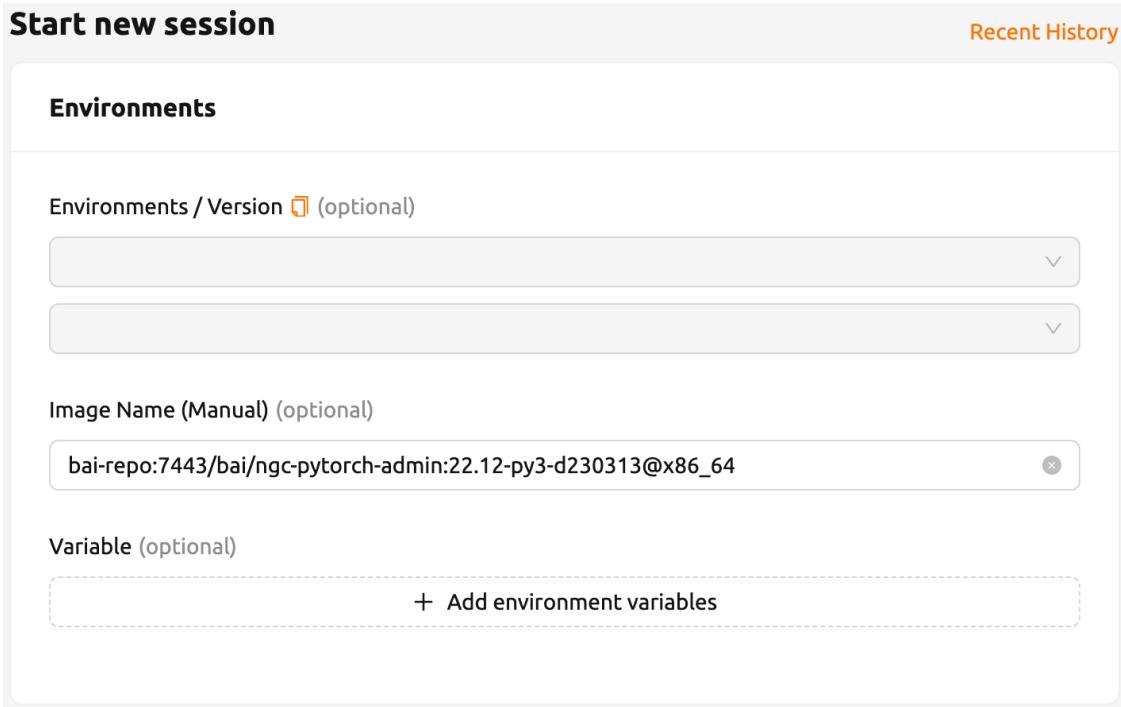
실행 중인 연산 세션(컨테이너) 환경을 새로운 이미지로 변환하고 추후 연산 세션 생성시 사용하고자 하는 경우, 연산 세션 내 환경을 구성한 후 관리자에게 변환 요청을 할 수 있습니다.

- 먼저, 연산 세션을 준비합니다. 필요한 패키지를 설치하거나 환경을 구성합니다.

참고: `apt` 등과 같은 명령을 통해 OS 패키지를 설치하려면 `sudo` 권한이 필요합니다. 플랫폼 제공사의 보안 정책에 따라 연산 세션 내에서 `sudo` 사용이 허용되지 않을 수 있습니다.

`Python` 패키지를 `pip`을 통해 설치하려는 경우, 자동 마운트 폴더 사용을 권장 드립니다. 하지만, 새 이미지 자체에 Python 패키지를 추가하려면 `sudo pip install <package-name>` 과 같이 실행하여 패키지를 홈 디렉토리가 아닌 시스템 디렉토리에 설치하여야 합니다. 연산 세션 내 홈 디렉토리(일반적으로 `/home/work`)는 호스트에서 마운트된 폴더이므로 신규 이미지로 변환할 때 내용이 포함되지 않습니다.

-
- 연산 세션 환경이 준비되면 관리자에게 이미지로의 변환을 요청합니다. 관리자에게 플랫폼 내 사용자의 이메일과 변환하고자 하는 연산 세션의 이름 또는 ID를 전달해야 합니다.
 - 관리자가 일정 주기로 연산 세션을 이미지로 변환한 후 이미지의 이름과 태그 정보를 전달할 것입니다.
 - 연산 세션 생성 디아얼로그에서 수동으로 이미지 이름을 입력한 후 연산 세션을 생성합니다. 변환된 이미지는 다른 사용자에게 노출되지 않습니다.



- 새 이미지를 활용해 연산 세션이 정상적으로 실행되어야 합니다.

23.5 Backend.AI 서버 설정 가이드

Backend.AI 데몬/서비스 구동을 위해서는 다음과 같은 하드웨어가 필요합니다. 최적 성능을 위해서는 아래 명기된 사양의 두 배 이상 필요합니다.

- Manager: 2 cores, 4 GiB memory
- Agent: 4 cores, 32 GiB memory, NVIDIA GPU (for GPU workload), > 512 GiB SSD
- Webserver: 2 cores, 4 GiB memory
- WSProxy: 2 cores, 4 GiB memory
- PostgreSQL DB: 2 cores, 4 GiB memory
- Redis: 1 core, 2 GiB memory
- Etc: 1 core, 2 GiB memory

각 서비스를 설치하기 전에 사전에 설치되어야 할 주요 의존 호스트 패키지는 다음과 같습니다:

- Web-UI: 최신 브라우저를 구동할 수 있는 운영체제 (Windows, Mac OS, Ubuntu 등)
- Manager: Python (≥ 3.8), pyenv/pyenv-virtualenv (≥ 1.2)

- Agent: docker (≥ 19.03), CUDA/CUDA Toolkit (≥ 8 , 11 recommend), nvidia-docker v2, Python (≥ 3.8), pyenv/pyenv-virtualenv (≥ 1.2)
- Webserver: Python (≥ 3.8), pyenv/pyenv-virtualenv (≥ 1.2)
- WSProxy: docker (≥ 19.03), docker-compose (≥ 1.24)
- PostgreSQL DB: docker (≥ 19.03), docker-compose (≥ 1.24)
- Redis: docker (≥ 19.03), docker-compose (≥ 1.24)
- Etcd: docker (≥ 19.03), docker-compose (≥ 1.24)

엔터프라이즈 버전의 Backend.AI 서버 데몬은 래블업의 지원팀에서 설치합니다. 초기 설치 후 기본적으로 다음과 같은 자료 및 서비스가 제공됩니다:

- DVD 1 장 (Backend.AI 패키지 포함)
- 사용자 GUI 가이드 매뉴얼
- 관리자 GUI 가이드 매뉴얼 (엔터프라이즈 고객 전용)
- 설치 리포트
- 사용자/관리자 초기 방문 교육 (3-5 시간)

제품의 유지보수 및 지원 정보: 상용 계약에는 기본적으로 엔터프라이즈 버전의 월간/연간 구독 사용료가 포함됩니다. 최초 설치 후 약 2주 간 초기 사용자/관리자 교육 (1-2 회) 및 유무선 상의 고객 지원 서비스가 제공되며, 3-6 개월 간 마이너 릴리즈 업데이터 지원 및 온라인 채널을 통한 고객 지원 서비스가 제공됩니다. 이후 제공되는 유지보수 및 지원 서비스는 계약 조건에 따라 세부 내용이 다를 수 있습니다.

23.6 통합 예제

이번 섹션에서는 Backend.AI 플랫폼에서 활용할 수 있는 여러 일반적인 응용 프로그램, 툴킷 및 머신러닝 도구의 예제를 소개하고자 합니다. 각 도구의 기본 사용법과 Backend.AI 환경에서 설정하는 방법, 그리고 간단한 예제를 알려드리겠습니다. 이를 통해 프로젝트에 필요한 도구를 선택하고 활용하는 데 도움이 되기를 바랍니다.

이 가이드에서 다루는 내용은 특정 버전의 프로그램을 기반으로 하고 있으므로, 향후 업데이트에 따라 사용법이 달라질 수 있습니다. 따라서, 이 문서는 참고하시되, 변경 사항을 확인하기 위해서는 최신 공식 문서를 참조하시기를 바랍니다. 이제 Backend.AI에서 사용할 수 있는 강력한 도구들을 하나씩 살펴보겠습니다. 이 섹션이 여러분의 연구 및 개발에 유용한 가이드가 되기를 바랍니다.

23.6.1 MLFlow 사용하기

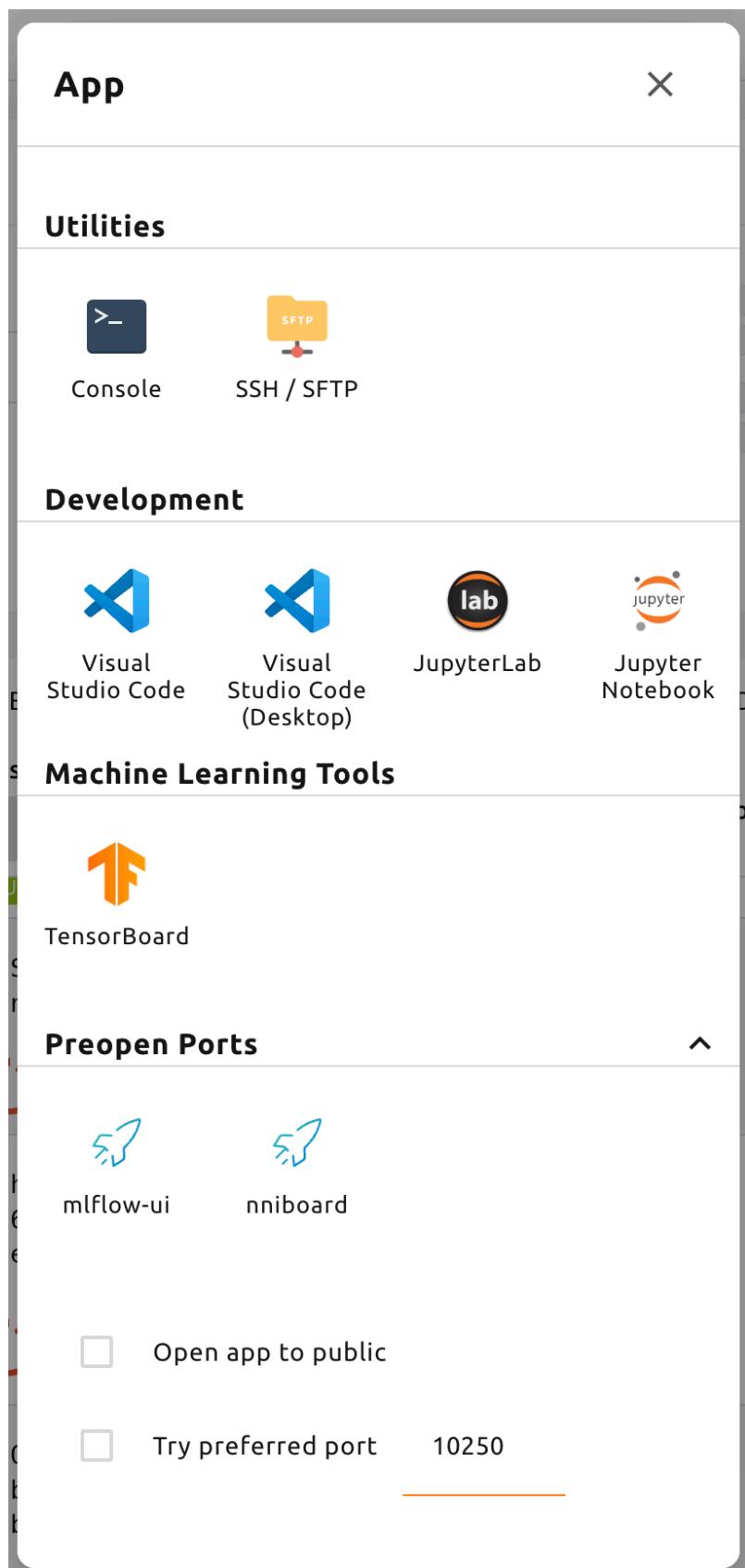
Backend.AI에서 제공하는 대부분의 이미지는 MLFlow와 MLFlow UI를 빌트인 앱으로 지원합니다. 하지만 실행하기 위해서는 기존 앱과 다르게 몇 가지 추가 작업이 필요합니다. 아래의 설명을 따라하면, 로컬 환경에서 MLFlow를 사용하셨던 것처럼 Backend.AI에서도 파라미터와 결과 값을 추적하실 수 있습니다.

참고: 이 섹션에서는, 세션을 이미 성공적으로 생성하고, 세션 내 앱을 실행할 수 있는 상태라고 가정합니다. 만약 세션을 생성하고 세션 내에서 앱을 실행하는 방법을 모르신다면, [세션 생성하기](#)를 반드시 먼저 보고 오시기 바랍니다.

우선, 콘솔 앱을 먼저 실행한 뒤, 아래의 명령어를 입력하면 mlflow UI 서버가 실행됩니다.

```
$ mlflow ui --host 0.0.0.0
```

그 다음, 앱 런처 디아얼로그에서 MLFlow UI 앱을 클릭합니다.



잠시 뒤, MLFlow UI 가 새 창에 띄워집니다.

MLFlow 를 사용함으로써, 매번 학습할 때마다 메트릭이나 파라미터 값과 같은 실험 결과를 추적할 수 있습니다. 아주 간단한 예제로 실험 결과를 확인해봅시다.

```
$ wget https://raw.githubusercontent.com/mlflow/mlflow/
  ↳master/examples/sklearn_elasticnet_diabetes/linux/train_
  ↳diabetes.py
$ python train_diabetes.py
```

Python 코드를 실행하면, MLFlow 에서 결과값을 확인할 수 있습니다.

			Parameters	Metrics						
	Start Time	Run Name	User	Source	Version	Models	alpha	l1_ratio	mae	r2
2024-04-04 18	-	work	train	-	sklearn		0.05	0.05	66.31	0.066

코드 실행시 인자값을 주어 하이퍼파라미터를 직접 설정할 수도 있습니다.

```
$ python train_diabetes.py 0.2 0.05
```

몇번의 학습이 끝나면, 모델 학습 결과 값들을 서로 비교해볼 수 있습니다.

The screenshot shows the mlflow UI interface. At the top, there is a navigation bar with 'mlflow' logo, 'Experiments', 'Models', 'GitHub', and 'Docs'. Below the navigation bar, the 'Default' experiment is selected. A modal window titled 'Default' contains the message: 'Track machine learning training runs in an experiment. Learn more'. Below the modal, the 'Experiment ID' is listed as '0' and the 'Artifact Location' is '/mlruns/0'. Under the 'Notes' section, it says 'None'. There is a search bar with the query 'metrics.rmse < 1 and params.model = "tree" and tags.mlflow.source.type...'. Below the search bar, it says 'Showing 6 matching runs'. There are three buttons: 'Compare', 'Delete', and 'Download CSV'. To the right of these buttons are 'Filter', 'Search', and 'Clear' buttons. Below these buttons are 'Columns' and 'Load more' buttons. The main area displays a table with the following data:

	Start Time	Run Name	User	Source	Version	Models	Parameters		Metrics		
							alpha	I1_ratio	mae	r2	rmse
<input type="checkbox"/>	2024-04-04 18:01	-	work	train_d	-	sklearn	0.2	0.05	68.87	-0.002	81.42
<input type="checkbox"/>	2024-04-04 18:01	-	work	train_d	-	sklearn	0.4	0.05	69.37	-0.016	81.98
<input type="checkbox"/>	2024-04-04 18:01	-	work	train_d	-	sklearn	0.5	0.05	69.47	-0.019	82.1
<input type="checkbox"/>	2024-04-04 18:01	-	work	train_d	-	sklearn	0.01	0.05	59.89	0.234	71.19
<input type="checkbox"/>	2024-04-04 18:01	-	work	train_d	-	sklearn	0.05	0.05	66.31	0.066	78.59
<input type="checkbox"/>	2024-04-04 18:01	-	work	train_d	-	sklearn	0.05	0.05	66.31	0.066	78.59

CHAPTER 24

사용권 계약 조건

24.1 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. LGPL를 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 LGPL 을 따라야 합니다. 하지만, 배포하는 소프트웨어가 Backend.AI 서버를 변경하지 않고 모듈 형태(Python import 등)로 불러서 사용하는 경우에는, 동적 링크를 사용하는 것으로 보아 LGPL에 따른 코드 공개 의무가 없다고 간주합니다. Backend.AI

서버를 하드웨어에 설치하여 일반 공중이 네트워크를 통해 사용하는 경우에는 LGPL에 따른 의무가 없습니다.

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 LGPL를 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 설치하여 일반 공중이 네트워크를 통해 사용하는 경우.
 - 무료로 사용하게 하는 경우: 래블업과의 별도 계약이 필요하지 않습니다.
 - 유료로 사용하게 하는 경우: 래블업과의 상용 계약이 필요합니다.
- Backend.AI 서버가 설치된 하드웨어를 배포하는 경우,
 - 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.

CHAPTER 25

참고 자료

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

이 문서의 최신 버전은 아래 사이트에서 볼 수 있습니다:

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