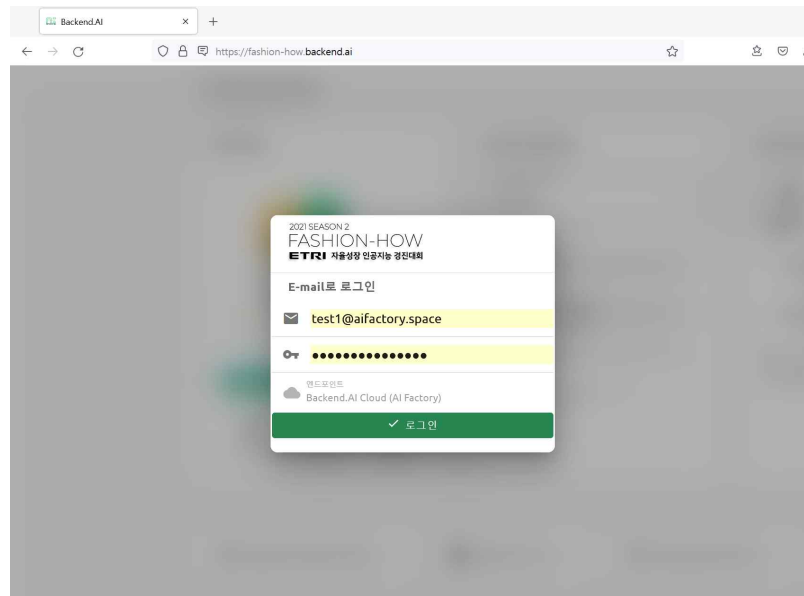


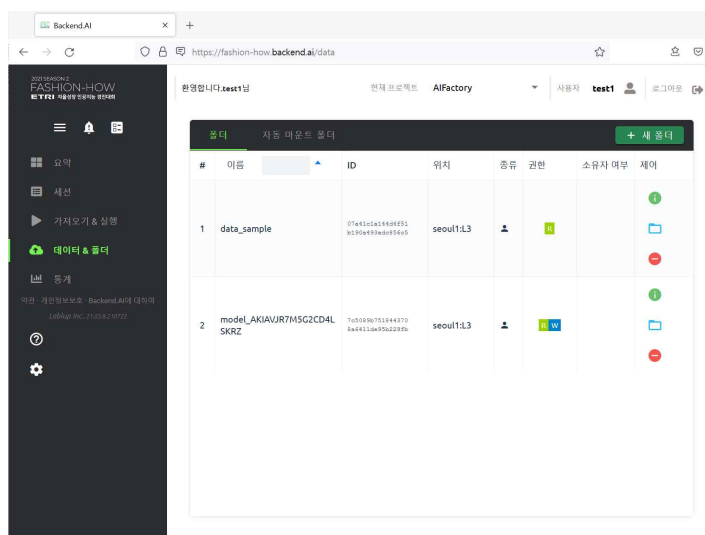
# 제출방법 가이드

## [로그인]

- <https://fashion-how.backend.ai> 접속
- 팀별 email로 전달된 계정정보 사용



## [데이터&폴더]



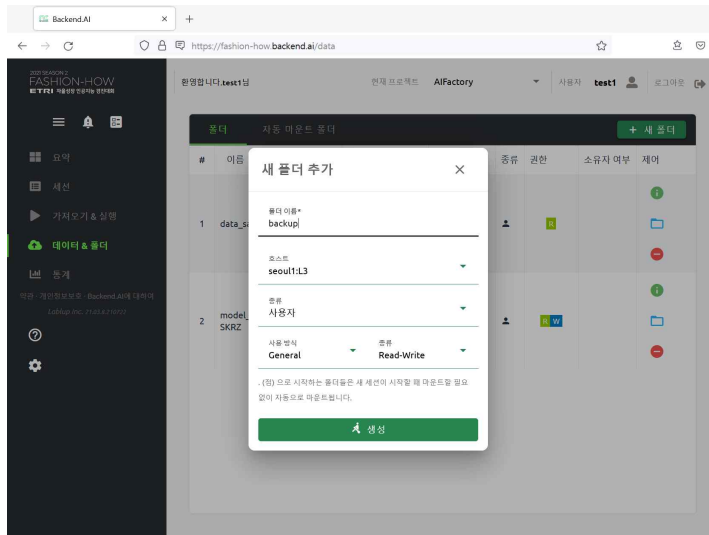
### • data\_sample

- 샘플 평가 데이터가 있는 폴더 (읽기 전용)
- 샘플 데이터는 베이스라인 모델에 포함된 데이터와 동일한 것으로 실제 평가 데이터와 동일한 양식으로 이루어져 있음
- 그밖에 학습셋, 이미지 추출데이터, raw 이미지 데이터도 모두 포함 되어있음

### • model\_\*

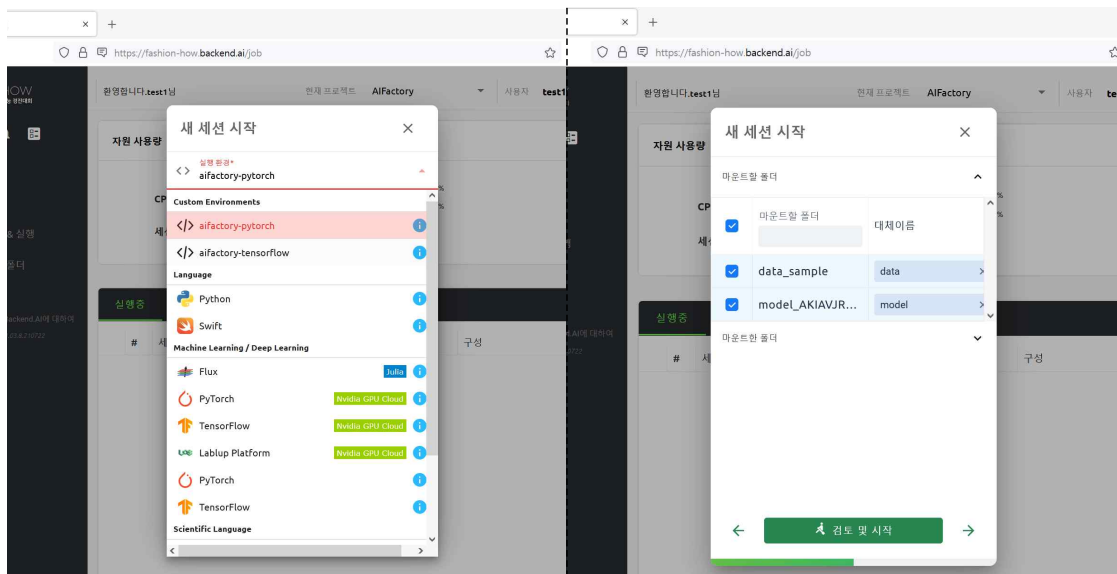
- 사용자가 제출할 모델을 업로드 하기 위한 폴더

※ 위 두 폴더는 반드시 유지해야 함



- 폴더 추가 가능
  - 폴더를 생성하여 마운트하면 세션에서 사용할 수 있음
  - model\_\* 폴더 및 사용자 생성 폴더의 내용은 세션이 종료되어도 보존되므로 중요한 파일은 생성한 폴더에 저장
  - 단, 사용자가 생성한 폴더는 [제출→모델가동→결과생성] 과정에서 사용되지 않음에 유의
- 파일 업로드
  - 드래그 앤 드롭으로 파일 업로드
  - 업로드 가능 단일 파일 용량을 초과하는 경우 분할압축 이용

## [세션 생성]



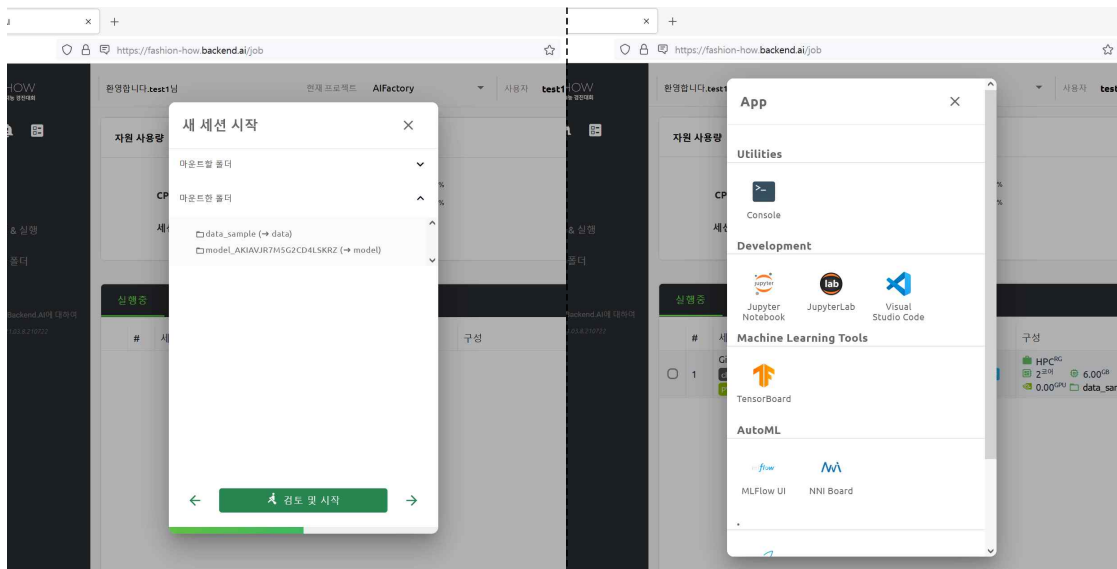
- 환경 선택: 1) aifactory\_pytorch  
2) aifactory\_tensorflow 두 환경 가운데 하나를 사용  
(※ 그밖의 경우 기초 환경이 준비되어있지 않음)
- 폴더 마운트: data\_sample → data  
model\_\*... → model 로 대체이름을 지정하여 마운트해야 함(필수)

※ 마운트한 폴더는 세션에서 각각 /home/work/data, /home/work/model의 경로로 인식되며 사용자가 폴더를 추가하여 마운트하면 역시 /home/work/ 아래에 인식

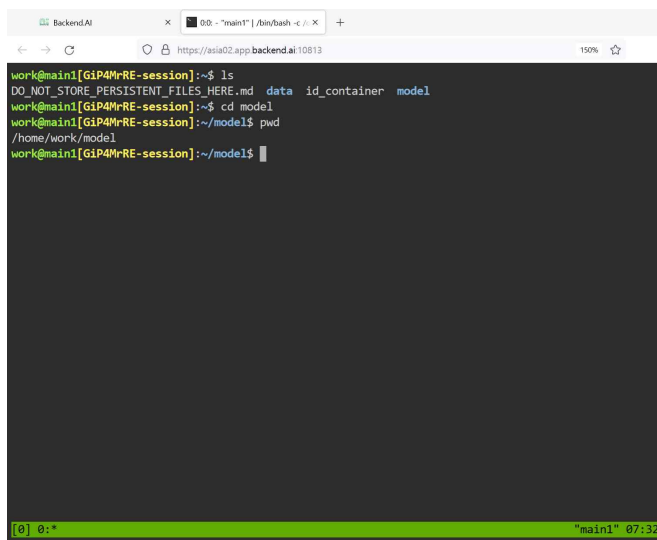
## ※ !주의!



- 폴더 마운트시 그림과 같이 [마운트한 폴더] 메뉴를 클릭하여 지정된 이름으로 마운트되었는지 반드시 확인이 필요
- 간혹 대체 이름이 적용되지 않은 경우 대체 이름을 삭제하고, 다시 입력한 후 마운트해서 확인
- 대체이름이 입력되지 않은 경우 브라우저의 입력이 한글로 설정되어있는지 확인 → 영어로



- 마운트 설정이 마무리되었으면 [검토 및 시작] → [시작]을 눌러서 세션 생성
- 세션이 정상적으로 생성되면 우측과 같이 App 메뉴가 나타나며 Console 및 여타의 개발환경 앱(VS Code, Jupyter Notebook 등)을 열어서 세션에 접근할 수 있음
- 환경 및 상황에 따라 세션 개설까지 약간의 딜레이가 있을 수 있음



- 현재까지의 과정이 정상적으로 진행된 경우 좌측과 같이 마운트한 폴더가 data, model로 잡힘 (최초 경로는 /home/work)

## [경로]

- 본 세션은 사용자가 각자의 환경에서 개발한 모델이 평가 세션에서 정상적으로 실행-추론-결과생성을 할 수 있는지 확인하는 목적으로 제공됨
- 사용자가 생성한 세션에서 이하의 경로를 맞추어 작성한 모델이 평가 데이터 샘플을 이용하여 정상적으로 가동, 결과를 생성하는 경우 제출 이후 실제 평가 데이터를 이용한 추론 과정 역시 정상적으로 작동함
- 제공되는 평가 데이터 샘플로 가동할 때 문제가 생기는 경우 실제 평가 데이터로도 정상 작동하지 않기 때문에 본 환경 내에서 가능한 오류를 파악하고 제출할 것이 권장됨
- 정상 작동을 위해서는 이하의 경로를 반드시 준수해야 함
- 사용자가 제출을 실행할 경우 아래와 같은 과정으로 평가가 수행됨
  1. 사용자 세션과 독립된 별도의 임시 평가용 세션 생성
  2. 평가용 세션은 /home/work/data에 샘플 평가 데이터가 아닌 **실제 평가 데이터**를 불러옴. 샘플과 실제 평가 데이터 사이의 양식은 동일
  3. 평가용 세션은 **사용자의 model\_\* 폴더를 /home/work/model** 경로로 마운트 (그밖의 폴더는 마운트되지 않음에 유의)
  4. 평가용 세션은 model 폴더 내의 **/home/work/model/predict.py** 실행
  5. **/home/work/model/prediction.csv** 생성
  6. 생성된 **prediction.csv**를 평가서버로 전송 → 평가
- 따라서 사용자는 **/home/work/model/predict.py**를 실행하면 모델의 추론이 실행되고 **/home/work/model/prediction.csv**로 결과를 저장하도록 모델의 코드를 작성해야 함
- 또한 pip 등을 이용하여 추가로 라이브러리를 설치하는 경우, 해당 라이브러리가 실제 평가용 세션에는 설치되어있지 않다는 점에 주의하고 **predict.py** 실행 시 필요한 라이브러리 설치가 함께 이루어질 수 있게 해야 함
- 평가용 세션에 마운트되는 폴더는 **/model\_\*** 하나이기 때문에, 모델이 정상 동작하는 데에 필요한 모든 리소스는 **/model\_\*** 폴더 하위에 저장하고 경로를 **/home/work/model/이하로** 지정해야 함

※ 이상의 경로는 반드시 준수해야 하며 (절대경로지정 권장)  
※ prediction.csv 양식 또한 준수해야 함 (인덱스, 헤더 없음)

## [베이스라인 테스트]

```
requirement._memn2n._B
requirement._memn2n._C
requirement._memn2n._H
requirement._memn2n._W
policy._mlp_eval.layer1_linear.weight
policy._mlp_eval.layer1_linear.bias
policy._mlp_eval.layer2_linear.weight
policy._mlp_eval.layer2_linear.bias
policy._mlp_eval.layer3_linear.weight
policy._mlp_eval.layer3_linear.bias
policy._mlp_eval.layer4_linear.weight
policy._mlp_eval.layer4_linear.bias
policy._mlp_rnk.layer1_linear.weight
policy._mlp_rnk.layer1_linear.bias
policy._mlp_rnk.layer2_linear.weight
policy._mlp_rnk.layer2_linear.bias
policy._mlp_rnk.layer3_linear.weight
policy._mlp_rnk.layer3_linear.bias

<Predict>
[*] load success: /home/work/model/gAia_model/gAia-10.pt

-----
Prediction Time: 2.88sec
# of Test Examples: 200
-----
work@main1[s0eityk-session]:~/model$ ls
LICENSE.txt  data          gaia.py      pip.txt      prediction.csv  run_test.sh    submit.sh
README.pdf   file_io.py    log.txt      policy.py    requirement.py  run_train.sh
__pycache__  gAia_model   main.py      predict.py   run_pred.sh    sstm_v0p5_deploy
work@main1[s0eityk-session]:~/model$
```

- /home/work/model 에서  
> **bash run\_pred.sh**로 실행
- 실행 종료되면 그림과 같이 prediction.csv가 생성되는 것을 확인할 수 있음
- 베이스라인 상세 내용은 함께 제공되는 베이스라인 설명서를 참고

## [제출]

- 1) 베이스라인에 포함되어 있는 submit.sh를 /home/work/model에 업로드하여 실행  
> bash /home/work/model/submit.sh

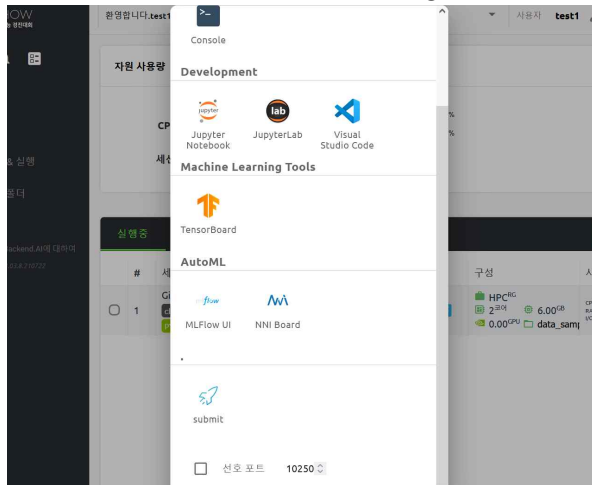
```
work@main1[sOeitKyK-session]:~/model$ bash submit.sh
{
  "message": "Submitted and queued the evaluation task",
  "data": {
    "evaluation_status": "RUNNING",
    "timestamp": "2021-08-08T09:32:10.774130+00:00"
  }
}
work@main1[sOeitKyK-session]:~/model$
```


- 정상 제출 시 좌측과 같은 메시지 출력

```
{"message": "Endpoint request timed out"}
work@main1[sOeitKyK-session]:~/model$
```

- “Endpoint request time out”이 발생하는 경우 일시적 장애이므로 곧바로 바로 재시도하면 됨

- 2) 세션 메뉴의 submit 아이콘 실행



- 세션-제어 아래 메뉴 버튼(  ) 클릭

- 스크롤을 내리면 submit 아이콘이 있음



- 클릭하면 bash 명령과 동일하게 제출됨

- 3) 출력 메시지 및 오류:

- 추론이 모두 종료되면 생성-전송된 결과는 삭제되고, 아래와 같은 파일이 생성됨
  - /home/work/model/log.txt (화면 출력, 에러 포함 전체 메시지)
  - /home/work/model/error\_log.txt (에러가 있을 시)

- 정상 제출 완료시 log.txt에서 다음과 같은 메시지를 확인할 수 있음

```
-----
100%|██████████| 1.37k/1.37k [00:00<00:00, 73.6kB/s]
success: 200
```

- ※ 결과가 정상 제출되더라도 생성한 prediction.csv의 양식이 잘못된 경우 평가서버의 점수 계산과정에서 오류가 발생할 수 있으므로 양식 준수 필요

## [정답양식 및 평가방식 이해]

※ 작성해야 하는 답안 = 각 대화별로 제시된 3개의 코디 조합의 순위 정하기

```
; 0
US   가을 축제 에 입고 갈 스타일 로 코디 해 주 세 요
CO   치마 바지 원피스 중 어떤 옷 이 포함_된 코디 를 추천_해 드릴_까 요
US   원피스 나 치마 로 추천_해 주 세 요
CO   가을 에 입 기 좋 은 적당_한 두께감 의 아우터 와 함께 추천_해 드릴_까 요
US   네
CO   상의 는 무늬 없 는 티 로 추천_해 드리 는 것 은 어떠 신가 요
US   좋 습니다
R1   JP-137 KN-008 SK-047 SE-042
R2   JP-137 KN-045 SK-047 SE-004
R3   JP-137 KN-045 SK-047 SE-052
```

← 입력데이터에 나열된 코디 조합의 순위를  
순서대로 나열

- 각 대화에 제시된 코디의 순위를

1위 → 0  
2위 → 1  
3위 → 2로 인덱싱하고,

나열된 순위 순서 리스트를 랭크테이블(우측 이미지)  
상의 인덱스로 대체하여 답안으로 기록

0	0	1	2
1	0	2	1
2	1	0	2
3	1	2	0
4	2	0	1
5	2	1	0

e.g. 문제 답안이 [1, 0, 2] 순서인 경우 → 2  
[2, 1, 0] 순서인 경우 → 5

- 결과로 생성할 **prediction.csv**는 index, column명 없이 row 순서대로 랭크테이블 인덱스를 기록한 utf-8 인코딩의 텍스트 문서

※ numpy array 기준

`np.savetxt("/home/work/model/prediction.csv", "변수명", fmt="%d")`

명령으로 저장

※ **Weighted Kendall's Tau 계산**

- 두 순위 배열 사이의 Correlation을 -1 ~ 1 사이의 스코어로 산출
- WKT는 순위가 높을수록 큰 값이 들어오도록 되어있음 (인덱스와 반대)
- 따라서 점수를 계산할 때에는 0 → 2,  
1 → 1,  
2 → 0 으로 인덱스를 역으로 변환하여 계산함
- 실제 계산 함수에 대해서는 베이스라인에 포함된 score.py 참조

## [기타]

- 사용자 세션과 마찬가지로 평가용 임시 세션은 GPU없이 CPU를 사용하여 추론을 수행하게 됨. 따라서 코드상에서 CPU 멀티코어를 활용하도록 구성하면 연산 속도에 다소 이득이 있음 (2CPU, 6GB 환경)
- 베이스라인 모델을 이용한 실제 평가 데이터 추론의 경우 동일한 환경에서 파일 로드-전처리 제외 약 10초 정도가 딥러닝 연산에 소요되며, 시스템 RAM은 최대 3GB 미만으로 사용됨
- 순위 상위를 기록하여 입상 대상이 되는 후보들의 경우
  - 코드분석
  - 재학습 및 재추론을 이용한 결과 동일성 검증을 거치게 됨
- 따라서 모델을 설계, 학습, 추론하는 과정에 결과 재현이 가능하도록 seed를 지정하여 무작위성을 최대한 통제해야 함