The AI Rewsletter

Late Edition

코딩실습과 기술정보 등 다양한 내용과 난이도로 담아냈습니다.

AI Newsletter No.8B

AI LAB, MONDAY, MAR 25, 2024

Al Tech



Tips

Chat GPT 자동번역기 "프롬프트 지니"

ChatGPT, 아직 영어로 쓰니? ① Chrome 웹스토어 접속 -> "프

ChatGPT는 영어로 질문해야 답 지니: ChatGPT 자동 번역기" 클릭 변 속도도 빠르고 정확도가 높다는 사실 알고 계신가요? 그 이유는 ChatGPT를 학습시킬 때 영어로 된 문서가 훨씬 많았기 때문이라고 하 는데요. 그럼, ChatGPT에게 영어로 질문을 해볼까요? ▶번역 사이트에 서 질문을 영어로 번역 ▶질문을 ChatGPT에 입력 ▶ ChatGPT 답변을 다 시 번역 사이트에서 우리말로 번역.

너무 번거롭죠? 이 문제를 해결할 플러그인이 바로 프롬프트 지니입 니다. 프롬프트 지니는 ChatGPT를 사용할 때, 질문을 영어로 번역해 주고, 답변은 다시 우리말로 번역 해 주는 크롬 확장프로그램입니다. 사용법을 알아보겠습니다.

롬프트 지니" 검색 -> "프롬프트



② 우측상단 [Chrome에 추가] 클릭



③ ChatGPT에 접속해보면 달라져 있는 채팅창을 확인할 수 있습니다.



④ 채팅창에 우리말을 입력하고 Ctrl+Enter, 채팅창에서 바로 번역 본을 확인할 수 있습니다.

#가상환경 활성화됨. base → env test



인공지능 기반 검색엔진, "퍼플렉시티 AI"

구글, 네이버 등 검색엔진마다 검 색 결과가 달라서 같은 질문을 여러 번 해야 할 때 번거로우셨죠? "Perplexity AI(perplexity.ai)"를 활용 해 보세요. Perplexity AI는 인공지능 기반 검색엔진으로, 웹의 다양한 소 스의 정보를 결합하여 답변을 생성 합니다. 또한 답변 제공에 활용된 온 라인 소스가 함께 제공되기 때문에, 사용자가 정보를 쉽게 검증해볼 수 있습니다. 답변 이후에는 관련 질문 들을 하단에 보여줌으로써 계속 질 문을 확장해 나갈 수도 있습니다.



Hands-on AI Project

SQL의 주요 기능을 Python Pandas로 실현하기

기 등 데이터를 다루기 위해 SQL 보시기 바랍니다. 쿼리가 Python 라이브러리인 Pandas로 어떻게 대칭되는지 실습 으로 설명합니다. Pandas는 전처리, 특성분석, 시각화 등 다양한 작업 을 지원, SQL로 수행하기 어려운 연산과 분석기능을 제공합니다. 아

① Dataframe의 정보 확인

select count(*) from airports;

1) SQL

이번 Hands-on에서는 Pandas 래의 실습 예제를 통해 동일한 작 Dataframe의 생성, 읽기, 갱신, 쓰 업을 SQL로 수행할 경우와 비교해

- AIRPORTS : 모든 공항에 대한 정보
- AIRPORT_FREQ: 공항의 통신 주파수
- RUNWAYS:공항의 활주로 정보

※ 위 3개 테이블 명세서 및 전체 Source 코드 및 결과는 분량상 생략, 전체 코 드는 아래 링크를 참고해주세요.

구글 코랩에서 열기(인터넷망)

② Jupyter를 설치합니다.

(env_test) C:\Users\SR>

(env_test) C:\Users\SR> pip install jupyter

(base) C:\Users\SR> conda activate env_test

#주피터 설치

③ Jupyter Notebook을 사용하려면 jupyter notebook 입력, Jupyter Lab을 사용하고 싶으면 jupyterlab 입력, Lab 실행이 안되면 pip install jupyter lab을 입력하세요.

가상환경 시리즈 3탄. 가상환경 실행해보기 feat. Jupyter

지난 호에서 생성한 가상환경을 게요! Jupyter는 각 셀 별로 실행

통합 개발 환경(IDE, Integrated 할 수 있어 중간중간 코딩 결과를

Development Environment) 에서 보기 좋습니다. 개인적으로 가장

실행해보겠습니다. 여러 IDE 중 자 많이 사용합니다. 주피터에는 주피

주 사용하는 주피터, VSCode, 터 노트북, 주피터 랩이 있는데, 노

PyCharm에서 실습해보겠습니다. 트북들을 모아서 볼 수 있는 것이

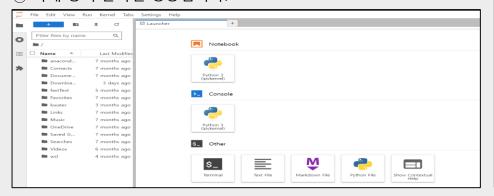
① Anaconda Prompt에서 env_test라는 가상환경을 만들고 실행합니다.

이번 8호에서는 주피터에서 해 볼 랩이라고 생각하면 됩니다.

(env_test) C:\Users\SR> jupyter notebook (env_test) C:\Users\SR> jupyterlab

주피터 노트북 사용시

④ 아래 창이 뜬다면 성공입니다!



select column_name, data_type from all_tab_columns where table_name = 'AIRPORTS';

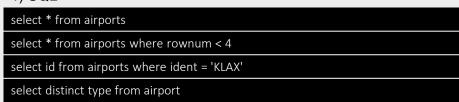
2) Pandas

airports.shape

Dataframe의 각 컬럼의 데이터 타입을 표시 airports.dtypes # Dataframe의 간단한 통계를 표시 airports.describe() # Dataframe의 shape을 표시

② distinct(unique), 단일 조건 검색

1) SQL



2) Pandas

airports airports.head(3)

airports[airports.ident == 'KLAX'].id

#ident 컬럼이 'KLAX'인 행의 id를 선택

unique한 type 값을 선택 airports.type.unique()



Hands-on AI Project



Al Tech

③ 다중조건 검색

1) SQL

select * from airports where iso region = 'US-CA' and type = 'seaplane base'

select ident, name, municipality from airports where iso_region= 'US-CA' and type= 'large_airport'

2) Pandas

#'iso_region' 컬럼이 'US-CA'이고 'type' 컬럼이 'seaplane_base'인 전체 행을 선택 airports[(airports.iso_region == 'US-CA') & (airports.type == 'seaplane_base')]

'iso_region' 컬럼이 'US-CA'이고 'type' 컬럼이 'large_airport'인 행을 선택 # 선택된 행에서 'ident', 'name', 'municipality' 컬럼만 가져옴 airports[(airports.iso_region== 'US-CA') & (airports.type== 'large_airport')][['ident', 'name', 'municipality']]

④ 특정 데이터를 포함하는 행 선택

SQL의 in과 not in 명령어를 이용하면 데이터셋 내에 특정 유형의 일치하는 요소를 확인하여 선택하거나 배제할 수 있습니다. 이와 같은 기능 ⑦ 집계 함수(Aggregate functions)를 활용한 데이터 통계 계산하기으로 Pandas에서는 isin() 메서드를 사용할 수 있습니다. 최대값, 최소값, 평균값, 중앙값 등 통계량을 계산하기 위해 SQL 쿠

1) SQL

select * from airports where type in ('heliport', 'balloonport')

select * from airports where type not in ('heliport', 'balloonport')

2) Pandas

'type' 컬럼'에서 값이 heliport' 또는 'balloonport' 행을 선택 airports[airports.type.isin(['heliport', 'balloonport'])]

'type' 컬럼'에서 값이 heliport' 또는 'balloonport'가 아닌 행을 선택 airports[~airports.type.isin(['heliport', 'balloonport'])]

⑤ 특정 컬럼을 기준으로 그룹화하고 그룹별 데이터 개수 계산하기 SQL에서는 특정 컬럼들을 골라 그룹화할 때 group by 명령어를 이용합니다. 이렇게 하면 특정 컬럼을 분류기준으로 하여 다른 컬럼들의 데이터를 나타낼 수 있습니다. 한편, Pandas에서는 groupby() 함수를 사용합니다. SQL에서는 주어진 조건에 따른 데이터 수를 알아내기 위해 count 명령어를 이용합니다. Pandas Dataframe에서는 비슷한 기능으로 size() 속성을 이용합니다. Pandas에서도 count() 메서드가 있지만, 이는 NaN을 제외한 데이터 수를 반환합니다.

1) SQL

select iso_country, type, count(*) from airports group by iso_country, type order by iso_country, type

select iso_country, type, count(*) from airports group by iso_country, type order by iso_country, count(*) desc

2) Pandas

groupby 함수를 이용해 'iso_country' 컬럼과 'type' 컬럼의 데이터들을 조합, 그룹화하고 size()함수를 이용해 그룹별 데이터 수를 반환 airports.groupby(['iso_country', 'type']).size()

반환된 데이터 수는 Pandas 시리즈 형식이므로 to_frame 메서드를 이용해 'size'라는 컬럼명을 가진 Dataframe으로 변환 airports.groupby(['iso_country', 'type']).size().to_frame('size')

sort_values() 메서드를 이용해 'iso_country' 컬럼을 기준으로 Dataframe을 오름차순(ascending=True)으로 정렬하고 'size' 컬럼을 기준으로는 내림차순(ascending=False)으로 정렬합니다. 정렬 후에는 기존 인덱스가 섞여버렸으므로 reset_index() 함수를 이용해 새로운 인덱스를 생성합니다(인수로 drop=true 설정 시 기존 인덱스는 삭제, drop=false면 기존 인덱스를 컬럼으로 변환해 저장)

airports.groupby(['iso_country','type']).size().to_frame('size').reset_index().sort_values(['iso_country', 'size'], ascending=[True, False])

⑥ 상위 N개 데이터 선택하기

다음 예제에서는 공항 수를 기준으로 정렬하고 가장 많은 상위 10개 국가만 선택합니다. 두 번째 예는 "상위 10개 다음 10개"를 추출하는 경우입니다.

1) SQL

select id from airports order by size desc fetch first 10 rows only

select id from (select id, row_number() over (order by size desc) as rn from airports) where rn > 10 and rn <= 20

2) Pandas

airports.nlargest(10, columns='id') # 'id' 컬럼을 기준으로 상위 10개의 행을 반환

airports Dataframe에서 'id' 컬럼을 기준으로 상위 20개의 행을 반환한 후, 마지막 10개 행을 반환

airports.nlargest(20, columns='id').tail(10)

⑦ 집계 함수(Aggregate functions)를 활용한 데이터 통계 계산하기 최대값, 최소값, 평균값, 중앙값 등 통계량을 계산하기 위해 SQL 쿼리를 활용하면 각 결과 통계량이 컬럼으로 나타납니다. 그러나 Pandas의 agg() 함수를 이용하면 각 통계량이 행으로 나타나게 됩니다. 이 경우 Dataframe을 전치하여(.T를 사용) 컬럼으로 변환할 수 있습니다.

1) SQL

select max(length_ft), min(length_ft), avg(length_ft), median(length_ft) from runways

2) Pandas

#runways Dataframe에서 'length_ft' 컬럼에 대한 최소값, 최대값, 평균 및 중앙값을 계산

runways.agg({'length_ft': ['min', 'max', 'mean', 'median']}) # runways Dataframe에서 'length_ft' 컬럼에 대한 최소값, 최대값, 평균 및 중앙값을 계산 후, Dataframe을 전치(Transpose) runways.agg({'length_ft': ['min', 'max', 'mean', 'median']}).T

® 조인(JOIN)하기

SQL에서는 두 개의 테이블을 병합할 때 join을 이용합니다. 이와 같은 기능으로 Pandas에서는 merge() 함수가 있습니다. merge() 함수는 병합할 데이터셋 명칭과, 병합에 기준이 되는 컬럼 등을 결정함으로써 두 객체를 열방향으로 조인합니다.

1) SQL

select a.airport_ident, a.type, a.description, a.frequency_mhz from airport_freq a, (select id from airports where ident='klax') b where a.airport ref = b.id

2) Pandas

#'ident'가 'KLAX'인 airports의 'id' 컬럼을 선택하여, 'airport_freq'와 'id'를 기준으로 inner 조인한 후 필요한 컬럼을 선택합니다. how paramter에서 inner는 두 데이터셋의 inner join, outer는 두 데이터셋의 outer join airport_freq.merge(airports[airports.ident == 'KLAX'][['id']], left_on='airport_ref', right_on='id', how='inner')[['airport_ident', 'type', 'description', 'frequency_mhz']]

