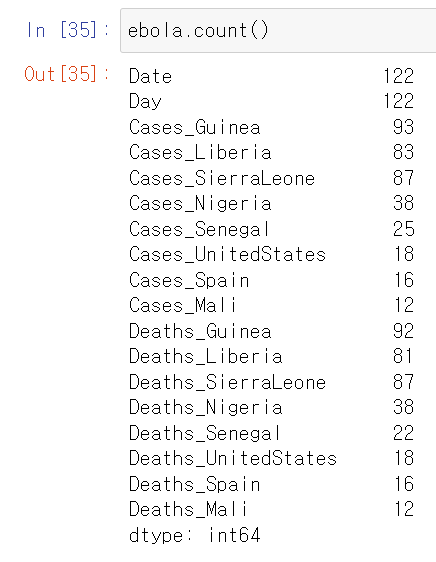
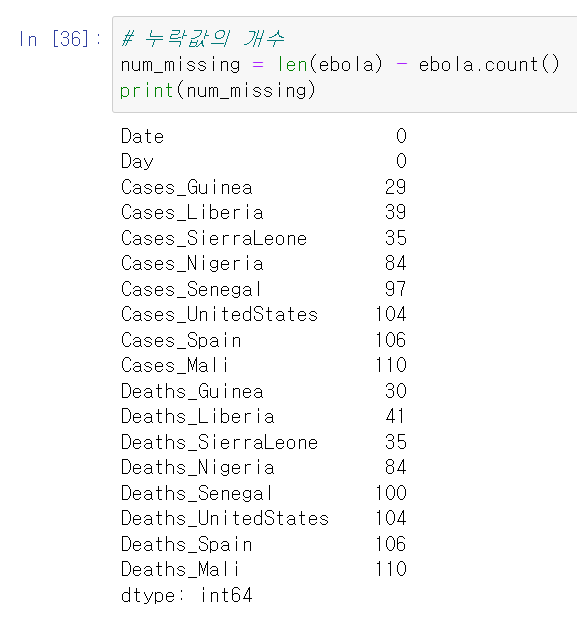
5/2 강의노트



Count 명령어는 결측치(NaN)을 세지 않음



누락값(NaN)의 개수는 전체 개수(len)에서 count를 빼면 됨

테이블이(가) 표시된 사진

자동 생성된 설명

Isnull 명령어를 쓰면 NaN인 데이터는 True로 출력

테이블이(가) 표시된 사진

자동 생성된 설명

해당 데이터프레임에 sum을 쓰면 NaN인 데이터의 개수가 나옴 (True = 1, False = 0이므로)

테이블이(가) 표시된 사진

자동 생성된 설명

특정 컬럼에서 value\_counts를 쓸 때 dropna = False를 쓰면 NaN 개수도 측정해줌

테이블이(가) 표시된 사진

자동 생성된 설명

결측치 대체

테이블이(가) 표시된 사진

자동 생성된 설명

Fillna 명령어를 쓰면 결측치를 지정한 값으로 변환이 가능함

테이블이(가) 표시된 사진

자동 생성된 설명

method 매개변수를 ffill로 지정하면 결측치가 나타나기 전의 값으로 채워넣게 됨, 앞의 값이 없으면 결측치로 유지됨

테이블이(가) 표시된 사진

자동 생성된 설명

Method 매개변수의 bfill 옵션 : ffill의 반대방향 개념

테이블이(가) 표시된 사진

자동 생성된 설명

Interpolate 명령어 : 누락값 양쪽의 중간값으로 처리함 (fillna와 다른 명령어)

텍스트이(가) 표시된 사진

자동 생성된 설명

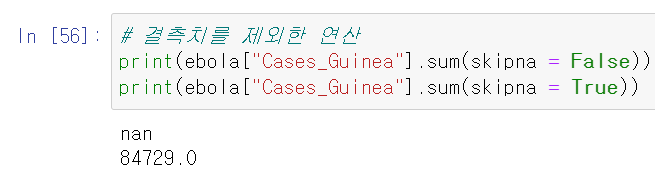
결측치가 너무 많아 사용하기 어려울 경우 마지막 방법으로 결측치 삭제 가능

Dropna 명령어는 결측치가 하나라도 있는 행을 모두 제거하지만 상당한 데이터 파괴가 일어날 수 있음 따라서 결측치를 대체할 수 있는 것들은 최대한 대체를 하는 것이 좋음

테이블이(가) 표시된 사진

자동 생성된 설명

매개변수 subset을 사용하여 특정 컬럼에서 NaN인 행만 제거가 가능함



결측치를 제외한 연산 : sum 명령어의 매개변수 skipna를 불 타입으로 지정하는 것으로 결측치 제외 여부를 결정 가능 (결측치는 확정되지 않은 값이므로 연산에 포함되면 NaN 값을 반환하게됨)

크롤링

크롤러와 스크래퍼

크롤러 : 알고리즘에 의해 인터넷을 탐색하는 프로그램

크롤링 : 크롤러가 웹페이지들을 돌아다니는 행위

스크래퍼 : 웹페이지의 데이터를 수집하는 프로그램

스크래핑 : 특정 페이지의 데이터를 수집하는 행위

일반적으로 부르는 크롤링 = 크롤링 + 스크래핑

클라이언트 : 웹 브라우저 및 웹 (크롬, 엣지, 스마트폰 앱 등)

서버 : 클라이언트의 처리를 돕는 것, 클라이언트가 요청하는 데이터를 제공

응답코드 : 서버가 클라이언트 요청을 처리하고 난 후 상태에 따라 반환하는 코드

보통 100번대 단위로 범주가 나뉨

100번대 : 클라이언트로부터 추가 데이터 요청 필요

200번대 : 성공

300번대 : 로그인 등으로 특정 페이지로 강제 이동을 처리해야 할 때 + 새로고침 등

400번대 : (클라이언트측) 요청 오류 (ex. 404 Page not Found)

500번대 : (서버측) 서버 오류

정적/동적 크롤링

정적 크롤링 : 어느 상황에서나 같은 주소에서 변하지 않는 데이터를 기대할 수 있는 경우 사용, 수집 대상에 한계가 있으나 속도가 빠름 (모듈 : BeautifulSoup 등)

동적 크롤링 : 입력, 클릭 등 실제 브라우저에서 행하는 행동을 수반해야 데이터를 수집할 수 있는 경우 사용, 수집 대상에 한계가 적으나 속도가 느림 (모듈 : Selenium 등)

경우에 따라 혼용함

DOM 트리

DOM : 웹 문서를 브라우저가 이해할 수 있는 구조로 구성하여 메모리에 적재하는 것 (계층 구조)

DOM 트리 : HTML 문서를 로드한 후 파싱하여 생성하는 모델

텍스트이(가) 표시된 사진

자동 생성된 설명

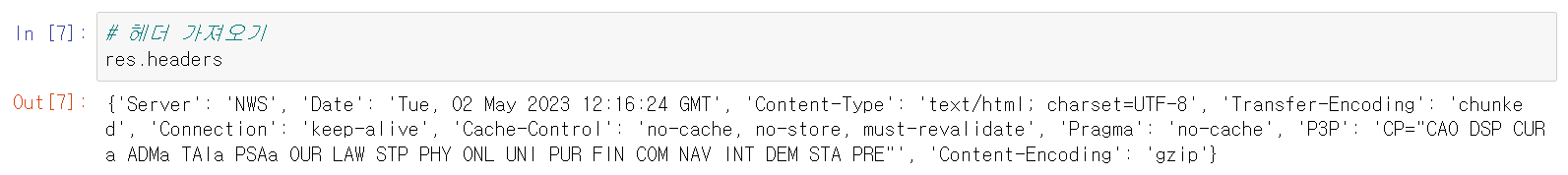
Requests로 웹페이지 접속 : get / post 명령어로 접속

Response 200 코드 : 정상 접속 완료

텍스트이(가) 표시된 사진

자동 생성된 설명

Status\_code로 응답코드를 받아올 수 있음



헤더 : 웹페이지의 본문 이전에 형식을 안내해줌, 본문은 바디로서 이후 들어옴, 명령어 headers

텍스트, 편지이(가) 표시된 사진

자동 생성된 설명

Text 명령어로 페이지의 html 코드를 확인 가능

텍스트이(가) 표시된 사진

자동 생성된 설명

쿼리스트링을 포함하는 데이터 생성 : get 명령어의 매개변수 params에 딕셔너리 형태의 쿼리스트링을 삽입하기 가능

텍스트이(가) 표시된 사진

자동 생성된 설명

urllib으로 웹페이지 접속하기

텍스트이(가) 표시된 사진

자동 생성된 설명

urllib으로 html 코드 표시 : read() 명령어 사용, 한글이 인코딩되어 출력됨

정적 크롤링

텍스트이(가) 표시된 사진

자동 생성된 설명

BeautifulSoup 명령으로 html 코드를 파이썬이 이해할 수 있는 형태로 가공할 수 있음, “lxml”은 파서

파서 : 데이터를 특정한 패턴이나 순서로 추출하는 프로그램, lxml은 그 중 하나임