Linux:
wget https://catalog.ourworldindata.org/garden/covid/latest/compact/compact.csv

hdfs dfs -mkdir final
hdfs dfs -put compact.csv final

hdfs dfs -mkdir -p final/hive_db
hdfs dfs -chmod 775 final/hive_db

hdfs dfs -mkdir -p /user/gn2279_nyu_edu/final/ods
hdfs dfs -mv /user/gn2279_nyu_edu/final/compact.csv /user/gn2279_nyu_edu/final/ods/

hive

Hive:

```
SET hive.metastore.warehouse.dir=final/hive_db;
SET hive.metastore.warehouse.dir;

CREATE DATABASE IF NOT EXISTS covid_db;
USE covid_db;

CREATE EXTERNAL TABLE IF NOT EXISTS ods_covid_raw (
    country STRING,
    `date` STRING,

    total_cases STRING,
    new_cases STRING,
    new_cases_smoothed STRING,
    total_cases_per_million STRING,
    new_cases_per_million STRING,
    new_cases_smoothed_per_million STRING,

    total_deaths STRING,
    new_deaths STRING,
    new_deaths_smoothed STRING,
    total_deaths_per_million STRING,
    new_deaths_per_million STRING,
    new_deaths_smoothed_per_million STRING,

    excess_mortality STRING,
    excess_mortality_cumulative STRING,
    excess_mortality_cumulative_absolute STRING,
    excess_mortality_cumulative_per_million STRING,
```

```
hosp_patients STRING,
hosp_patients_per_million STRING,
weekly_hosp_admissions STRING,
weekly_hosp_admissions_per_million STRING,

icu_patients STRING,
icu_patients_per_million STRING,
weekly_icu_admissions STRING,
weekly_icu_admissions_per_million STRING,

stringency_index STRING,
reproduction_rate STRING,

total_tests STRING,
new_tests STRING,
total_tests_per_thousand STRING,
new_tests_per_thousand STRING,
new_tests_smoothed STRING,
new_tests_smoothed_per_thousand STRING,

positive_rate STRING,
tests_per_case STRING,

total_vaccinations STRING,
people_vaccinated STRING,
people_fully_vaccinated STRING,
total_boosters STRING,
new_vaccinations STRING,
new_vaccinations_smoothed STRING,

total_vaccinations_per_hundred STRING,
people_vaccinated_per_hundred STRING,
people_fully_vaccinated_per_hundred STRING,
total_boosters_per_hundred STRING,

new_vaccinations_smoothed_per_million STRING,
new_people_vaccinated_smoothed STRING,
new_people_vaccinated_smoothed_per_hundred STRING,

code STRING,
continent STRING,
population STRING,
population_density STRING,
median_age STRING,
```

```
    life_expectancy STRING,
    gdp_per_capita STRING,
    extreme_poverty STRING,
    diabetes_prevalence STRING,
    handwashing_facilities STRING,
    hospital_beds_per_thousand STRING,
    human_development_index STRING
)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
STORED AS TEXTFILE
LOCATION '/user/gn2279_nyu_edu/final/ods'
TBLPROPERTIES ("skip.header.line.count"="1");


CREATE TABLE dwd_covid_clean_full (
    country STRING,
    `date` STRING,

    total_cases DOUBLE,
    new_cases DOUBLE,
    new_cases_smoothed DOUBLE,
    total_cases_per_million DOUBLE,
    new_cases_per_million DOUBLE,
    new_cases_smoothed_per_million DOUBLE,

    total_deaths DOUBLE,
    new_deaths DOUBLE,
    new_deaths_smoothed DOUBLE,
    total_deaths_per_million DOUBLE,
    new_deaths_per_million DOUBLE,
    new_deaths_smoothed_per_million DOUBLE,

    excess_mortality DOUBLE,
    excess_mortality_cumulative DOUBLE,
    excess_mortality_cumulative_absolute DOUBLE,
    excess_mortality_cumulative_per_million DOUBLE,

    hosp_patients DOUBLE,
    hosp_patients_per_million DOUBLE,
    weekly_hosp_admissions DOUBLE,
    weekly_hosp_admissions_per_million DOUBLE,

    icu_patients DOUBLE,
```

```
icu_patients_per_million DOUBLE,
weekly_icu_admissions DOUBLE,
weekly_icu_admissions_per_million DOUBLE,

stringency_index DOUBLE,
reproduction_rate DOUBLE,

total_tests DOUBLE,
new_tests DOUBLE,
total_tests_per_thousand DOUBLE,
new_tests_per_thousand DOUBLE,
new_tests_smoothed DOUBLE,
new_tests_smoothed_per_thousand DOUBLE,

positive_rate DOUBLE,
tests_per_case DOUBLE,

total_vaccinations DOUBLE,
people_vaccinated DOUBLE,
people_fully_vaccinated DOUBLE,
total_boosters DOUBLE,
new_vaccinations DOUBLE,
new_vaccinations_smoothed DOUBLE,

total_vaccinations_per_hundred DOUBLE,
people_vaccinated_per_hundred DOUBLE,
people_fully_vaccinated_per_hundred DOUBLE,
total_boosters_per_hundred DOUBLE,

new_vaccinations_smoothed_per_million DOUBLE,
new_people_vaccinated_smoothed DOUBLE,
new_people_vaccinated_smoothed_per_hundred DOUBLE,

code STRING,
population DOUBLE,
population_density DOUBLE,
median_age DOUBLE,
life_expectancy DOUBLE,
gdp_per_capita DOUBLE,
extreme_poverty DOUBLE,
diabetes_prevalence DOUBLE,
handwashing_facilities DOUBLE,
hospital_beds_per_thousand DOUBLE,
human_development_index DOUBLE
```

```
)
PARTITIONED BY (
    continent STRING,
    year STRING
)
STORED AS PARQUET;

SET hive.exec.dynamic.partition = true;
SET hive.exec.dynamic.partition.mode = nonstrict;

INSERT OVERWRITE TABLE dwd_covid_clean_full
PARTITION (continent, year)
SELECT
    country,
    `date`,

    CAST(total_cases AS DOUBLE),
    CAST(new_cases AS DOUBLE),
    CAST(new_cases_smoothed AS DOUBLE),
    CAST(total_cases_per_million AS DOUBLE),
    CAST(new_cases_per_million AS DOUBLE),
    CAST(new_cases_smoothed_per_million AS DOUBLE),

    CAST(total_deaths AS DOUBLE),
    CAST(new_deaths AS DOUBLE),
    CAST(new_deaths_smoothed AS DOUBLE),
    CAST(total_deaths_per_million AS DOUBLE),
    CAST(new_deaths_per_million AS DOUBLE),
    CAST(new_deaths_smoothed_per_million AS DOUBLE),

    CAST(excess_mortality AS DOUBLE),
    CAST(excess_mortality_cumulative AS DOUBLE),
    CAST(excess_mortality_cumulative_absolute AS DOUBLE),
    CAST(excess_mortality_cumulative_per_million AS DOUBLE),

    CAST(hosp_patients AS DOUBLE),
    CAST(hosp_patients_per_million AS DOUBLE),
    CAST(weekly_hosp_admissions AS DOUBLE),
    CAST(weekly_hosp_admissions_per_million AS DOUBLE),

    CAST(icu_patients AS DOUBLE),
    CAST(icu_patients_per_million AS DOUBLE),
    CAST(weekly_icu_admissions AS DOUBLE),
    CAST(weekly_icu_admissions_per_million AS DOUBLE),
```

```
CAST(stringency_index AS DOUBLE),
CAST(reproduction_rate AS DOUBLE),

CAST(total_tests AS DOUBLE),
CAST(new_tests AS DOUBLE),
CAST(total_tests_per_thousand AS DOUBLE),
CAST(new_tests_per_thousand AS DOUBLE),
CAST(new_tests_smoothed AS DOUBLE),
CAST(new_tests_smoothed_per_thousand AS DOUBLE),

CAST(positive_rate AS DOUBLE),
CAST(tests_per_case AS DOUBLE),

CAST(total_vaccinations AS DOUBLE),
CAST(people_vaccinated AS DOUBLE),
CAST(people_fully_vaccinated AS DOUBLE),
CAST(total_boosters AS DOUBLE),
CAST(new_vaccinations AS DOUBLE),
CAST(new_vaccinations_smoothed AS DOUBLE),

CAST(total_vaccinations_per_hundred AS DOUBLE),
CAST(people_vaccinated_per_hundred AS DOUBLE),
CAST(people_fully_vaccinated_per_hundred AS DOUBLE),
CAST(total_boosters_per_hundred AS DOUBLE),

CAST(new_vaccinations_smoothed_per_million AS DOUBLE),
CAST(new_people_vaccinated_smoothed AS DOUBLE),
CAST(new_people_vaccinated_smoothed_per_hundred AS DOUBLE),

code,

CAST(population AS DOUBLE),
CAST(population_density AS DOUBLE),
CAST(median_age AS DOUBLE),
CAST(life_expectancy AS DOUBLE),
CAST(gdp_per_capita AS DOUBLE),
CAST(extreme_poverty AS DOUBLE),
CAST(diabetes_prevalence AS DOUBLE),
CAST(handwashing_facilities AS DOUBLE),
CAST(hospital_beds_per_thousand AS DOUBLE),
CAST(human_development_index AS DOUBLE),

continent,
```

```
    substr(`date`, 1, 4) AS year
FROM ods_covid_raw
WHERE
    country IS NOT NULL
    AND `date` IS NOT NULL
    AND continent IS NOT NULL
    AND continent != '';
```

Here, we completed the DWD level database, next step we are going to work on DWS level table.

1.

```
CREATE TABLE dws_time_series_trend (
    `date` STRING,

    total_cases DOUBLE,
    total_deaths DOUBLE,
    total_vaccinations DOUBLE,

    new_cases DOUBLE,
    new_deaths DOUBLE,
    new_vaccinations DOUBLE,

    avg_new_cases_7d DOUBLE,
    avg_new_deaths_7d DOUBLE,

    continent STRING
)
PARTITIONED BY (year STRING)
STORED AS PARQUET;

INSERT OVERWRITE TABLE dws_time_series_trend
PARTITION (year)
SELECT
    `date`,

    SUM(total_cases) AS total_cases,
    SUM(total_deaths) AS total_deaths,
    SUM(total_vaccinations) AS total_vaccinations,

    SUM(new_cases) AS new_cases,
    SUM(new_deaths) AS new_deaths,
    SUM(new_vaccinations) AS new_vaccinations,
```

```
    AVG(SUM(new_cases)) OVER (
        PARTITION BY continent
        ORDER BY `date`
        ROWS BETWEEN 6 PRECEDING AND CURRENT ROW
    ) AS avg_new_cases_7d,

    AVG(SUM(new_deaths)) OVER (
        PARTITION BY continent
        ORDER BY `date`
        ROWS BETWEEN 6 PRECEDING AND CURRENT ROW
    ) AS avg_new_deaths_7d,

    continent,
    year
FROM dwd_covid_clean_full
GROUP BY continent, year, `date`;
```

2.

```
CREATE TABLE dws_country_risk_profile (
    country STRING,
    code STRING,

    total_cases DOUBLE,
    total_deaths DOUBLE,
    population DOUBLE,

    infection_rate DOUBLE,
    mortality_rate DOUBLE,

    gdp_per_capita DOUBLE,
    human_development_index DOUBLE
)
PARTITIONED BY (continent STRING, year STRING)
STORED AS PARQUET;

INSERT OVERWRITE TABLE dws_country_risk_profile
PARTITION (continent, year)
SELECT
    country,
    code,

    MAX(total_cases) AS total_cases,
    MAX(total_deaths) AS total_deaths,
```

```
    MAX(population) AS population,

    MAX(total_cases) / MAX(population)        AS infection_rate,
    MAX(total_deaths) / MAX(total_cases)      AS mortality_rate,

    MAX(gdp_per_capita) AS gdp_per_capita,
    MAX(human_development_index) AS human_development_index,

    continent,
    year
FROM dwd_covid_clean_full
GROUP BY country, code, continent, year;

3.
CREATE TABLE dws_vaccine_testing_effect (
    country STRING,
    `date` STRING,

    people_vaccinated_per_hundred DOUBLE,
    total_vaccinations DOUBLE,

    new_cases DOUBLE,
    new_deaths DOUBLE,

    total_tests DOUBLE,
    positive_rate DOUBLE,

    recovery_indicator DOUBLE
)
PARTITIONED BY (continent STRING, year STRING)
STORED AS PARQUET;

INSERT OVERWRITE TABLE dws_vaccine_testing_effect
PARTITION (continent, year)
SELECT
    country,
    `date`,

    people_vaccinated_per_hundred,
    total_vaccinations,

    new_cases,
    new_deaths,
```

```
    total_tests,
    positive_rate,

    (people_vaccinated_per_hundred / (new_cases + 1)) AS recovery_indicator,

    continent,
    year
FROM dwd_covid_clean_full;

4.
CREATE TABLE dws_policy_effectiveness (
    country STRING,
    `date` STRING,

    stringency_index DOUBLE,
    reproduction_rate DOUBLE,

    new_cases DOUBLE,
    new_deaths DOUBLE,

    lag_7d_stringency DOUBLE,
    lag_7d_new_cases DOUBLE
)
PARTITIONED BY (continent STRING, year STRING)
STORED AS PARQUET;

INSERT OVERWRITE TABLE dws_policy_effectiveness
PARTITION (continent, year)
SELECT
    country,
    `date`,

    stringency_index,
    reproduction_rate,

    new_cases,
    new_deaths,

    LAG(stringency_index, 7) OVER (
        PARTITION BY country
        ORDER BY `date`
    ) AS lag_7d_stringency,

    LAG(new_cases, 7) OVER (
```

```
        PARTITION BY country
        ORDER BY `date`
    ) AS lag_7d_new_cases,

    continent,
    year
FROM dwd_covid_clean_full;
```

Export from hive to local for visualization and analysis:

```
INSERT OVERWRITE DIRECTORY '/user/gn2279_nyu_edu/final/export/dwd_covid_clean_full'
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
SELECT * FROM dwd_covid_clean_full;

INSERT OVERWRITE DIRECTORY
'/user/gn2279_nyu_edu/final/export/dws_time_series_trend'
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
SELECT * FROM dws_time_series_trend;

INSERT OVERWRITE DIRECTORY
'/user/gn2279_nyu_edu/final/export/dws_country_risk_profile'
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
SELECT * FROM dws_country_risk_profile;

INSERT OVERWRITE DIRECTORY
'/user/gn2279_nyu_edu/final/export/dws_vaccine_testing_effect'
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
SELECT * FROM dws_vaccine_testing_effect;

INSERT OVERWRITE DIRECTORY
'/user/gn2279_nyu_edu/final/export/dws_policy_effectiveness'
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
SELECT * FROM dws_policy_effectiveness;
```