```
stop

***************************************************
* First steps
***************************************************

* Set working directory

cd "C:\Work\COVID19"

* Set log file
* Use option 'append', if you want to add to current log file
* For more info type: help log

log using today_date.log, replace

***************************************************
***************************************************
***************************************************
* Import data into Stata
***************************************************
***************************************************
***************************************************


***************************************************
* Import cases data
***************************************************

* Source of data
* https://data.humdata.org/dataset/novel-coronavirus-2019-ncov-cases

import delimited "time_series_covid19_confirmed_global_iso3_regions.csv", clear

* Browse data

browse

* Rename a variable
* For more info type: help rename

rename v4 longitude

* Convert numbers from string to numeric
* For more info type: help destring
* If you get the error: "contains nonnumeric characters; no replace"
* use the option ignore() to account for any non-numeric characters

destring lat longitude, replace ignore("#geo+lat" "#geo+long")

* Drop first case
* For more info type: help drop

drop in 1

* Replace variable names with variable labels
* For more info type: help foreach

foreach v of varlist v* {
   local z : variable label `v'  /* save variable label in local `z' */
   local w : subinstr local z "/" "_", all /* replace all "/" with "_" and saves it in local `w' */
   rename `v' x`w' /* renames the variable with local `w' and prefix 'x' */
}

* Saving data in Stata format
* Stata files have extension *.dta

save cases, replace

* Descriptive statistics
```

```
* For more info type: help summarize
* Check also commands: tabstat, table

summarize

* Frequencies
* For more info type: help tab

tab countryregion

* Aggregate at the country/region level (one observation per country/region)
* It creates a new dataset, you need to save it again (preferable with different name)
* For more info type: help collapse

collapse (sum) x*, by(countryregion)

* Check frequencies again

tab countryregion

save cases_agg, replace

* Reshape long to wide (create panel data)
* See here: https://www.princeton.edu/~otorres/DataPrep101.pdf#page=27
* It creates a new dataset, you need to save it again (preferable with different name)
* For more info type: help reshape

reshape long x, i(countryregion) j(date) string

browse

* Rename a variable

rename x cases

* Add a label variable

label variable cases "Cumulative cases"

* Format 'date' as date variable
* See here: https://www.princeton.edu/~otorres/TS101.pdf
* For more info type: help gen
* For more info type: help format
* For more info type: help date

gen datevar = date(date, "MDY", 2099)

format datevar %td

*format datevar %tdM_D,_YY

* Save the data

save cases_final, replace

************************************************
* Import death data
************************************************

import delimited "time_series_covid19_deaths_global_iso3_regions.csv", clear

* Browse data

browse

* Rename a variable
* For more info type: help rename

rename v4 longitude

```

```
* Convert numbers from string to numeric
* For more info type: help destring
* If you get the error: "contains nonnumeric characters; no replace"
* use the option ignore() to account for any non-numeric characters

destring lat longitude, replace ignore("#geo+lat" "#geo+long")

* Drop first case
* For more info type: help drop

drop in 1

* Replace variable names with variable labels
* For more info type: help foreach

foreach v of varlist v* {
   local z : variable label `v'  /* save variable label in local `z' */
   local w : subinstr local z "/" "_", all /* replace all "/" with "_" and saves it in local `w' */
   rename `v' x`w' /* renames the variable with local `w' and prefix 'x' */
}

* Saving data in Stata format
* Stata files have extension *.dta

save deaths, replace

* Descriptive statistics
* For more info type: help tab
* Check also commands: tabstat, table

summarize

* Frequencies
* For more info type: help tab

tab countryregion

* Aggregate at the country/region level (one observation per country/region)
* It creates a new dataset, you need to save it again (preferable with different name)
* For more info type: help collapse

collapse (sum) x*, by(countryregion)

* Check frequencies again

tab countryregion

save deaths_agg, replace

* Reshape long to wide (create panel data)
* It creates a new dataset, you need to save it again (preferable with different name)
* For more info type: help reshape

reshape long x, i(countryregion) j(date) string

browse

* Rename a variable

rename x deaths

* Add a label variable

label variable deaths "Cumulative deaths"

* Format 'date' as date variable
* For more info type: help gen
* For more info type: help format
* For more info type: help date

```

```
gen datevar = date(date, "MDY", 2099)

format datevar %td

*format datevar %tdM_D,_YY

* Save the data

save deaths_final, replace

****************************************
* Import WDI data
****************************************

* Source of data:
* https://databank.worldbank.org/source/world-development-indicators

import excel "WDI2021Select.xlsx", sheet("Data") firstrow clear

* Removing variables
* For more info type: help drop

drop Time TimeCode

* Renaming variables
* For more info type: help rename

rename CountryName country
rename CountryCode ccode
rename GDPpercapitaconstant2015US gdppc
rename PopulationtotalSPPOPTOTL pop
rename Populationdensitypeoplepers popdensity

tab country

* Replacing WDI country names to match the COVID19 dataset
* For more info type: help replace

replace country = "Bahamas" if country == "Bahamas, The"
replace country = "Brunei" if country == "Brunei Darussalam"
replace country = "Congo (Brazzaville)" if country == "Congo, Rep."
replace country = "Congo (Kinshasa)" if country == "Congo, Dem. Rep."
replace country = "Czechia" if country == "Czech Republic"
replace country = "Egypt" if country == "Egypt, Arab Rep."
replace country = "Gambia" if country == "Gambia, The"
replace country = "Iran" if country == "Iran, Islamic Rep."
replace country = "Korea, North" if country == "Korea, Dem. People's Rep."
replace country = "Korea, South" if country == "Korea, Rep."
replace country = "Kyrgyzstan" if country == "Kyrgyz Republic"
replace country = "Laos" if country == "Lao PDR"
replace country = "Micronesia" if country == "Micronesia, Fed. Sts."
replace country = "Russia" if country == "Russian Federation"
replace country = "Slovakia" if country == "Slovak Republic"
replace country = "Syria" if country == "Syrian Arab Republic"
replace country = "Turkey" if country == "Turkiye"
replace country = "US" if country == "United States"
replace country = "Venezuela" if country == "Venezuela, RB"
replace country = "Yemen" if country == "Yemen, Rep."

* Saving as Stata data

save wdi2021, replace

*******************************************************
* Import World Values Survey dataset
*******************************************************

* Source of data:
* https://www.worldvaluessurvey.org/WVSOnline.jsp

```

```
import excel "Confidence_The_Government20172022.xls", sheet("Sheet1") firstrow clear

tab country

* Replacing WDI country names to match the COVID19 dataset
* For more info type: help replace

replace country = "Bosnia and Herzegovina" if country == "Bosnia Herzegovina"
replace country = "Korea, South" if country == "South Korea"
replace country = "US" if country == "United States"

save wvs7, replace


**************************************************
**************************************************
**************************************************
* Merging data
**************************************************
**************************************************
**************************************************


**************************************************
* Merge cases and deaths dataset
**************************************************

* Open data in Stata format

use cases_final, clear

* For more info type: help merge

merge 1:1 countryregion datevar using deaths_final, gen(merge1)

tab merge1

* Save dataset

save covid, replace

**************************************************
* Merge COVID and WDI datasets
**************************************************

* Rename a variable

rename countryregion country

* Running the following will give an error, see below

merge m:1 country using wdi2021, gen(merge2)

* Getting the following error
* "variable country does not uniquely identify observations in the using data"
* We need to check the WDI data for duplicate values in variable 'country'

use wdi2021, clear

tab country, missing

* It looks like there are missing values in variable 'country'
* Drop missing values which could be:
*   - blanks in string variables (use double quotes)
*   - dots in numeric variables (use only a dot, no quotes)

drop if country == ""

save, replace

```

```
* Repeat

use covid, clear

rename countryregion country

merge m:1 country using wdi2021, gen(merge2)

* NOTE: If you want to keep only the perfect matches type:
*keep if merge2 == 3

save covid_final.dta, replace

*********************************************************
*Merge WVS with covid_final dataset
*********************************************************

merge m:1 country using wvs7, gen(merge3)

* Creating a new variable from others in the data

gen trust = Agreatdeal + Quitealot

* Add a label to the new variable

label variable trust "Trust in goverment (a great deal + quite a lot)"

* Save the data

save, replace

* Move id variables to top or begining of the dataset
* For more info type: help order

order country datevar date

*****************************************************
*****************************************************
*****************************************************
* Visualizing data
*****************************************************
*****************************************************
*****************************************************

* See here for a gallery of graphs in Stata
* https://www.stata.com/support/faqs/graphics/gph/stata-graphs/
* https://www.stata.com/features/publication-quality-graphics/

* Describe and summarize the data
* For more info type: help describe
* For more info type: help summarize

describe

summarize

* Set dataset as panel dataset
* Allow to use time series operators
* For more info type: help xtset

* Running the following will give an error, see below

xtset country datevar

* Gives error:
* "string variables not allowed in varlist;
*  country is a string variable"

* Country variable needs to have a number associated to each country.
* 'encode' assigns a number in alphabetical order
```

```
* Do not use 'encode' to conver numbers in string to numeric, (use 'destring' instead)

encode country, generate(country1)

order country1 country

browse

* Set dataset as panel dataset
* Allow to use time series operators

xtset country1 datevar

* For time series data type: tsset datevar

* Time series or line graphs
* For more info type: help twoway line

*xtline cases
*xtline cases, overlay legend(off)
* For more info type: help xtline

* Time series graph or line graphs
* Time variable always on x-axis (horizontal)
* For more info type: help twoway line


twoway line cases datevar if country == "US"

twoway line cases datevar if country == "US", title(Cumulative COVID-19 cases in the United States) ///
                        ylabel( , labsize(2) format(%11.0fc)) ///
                        ytitle(# of cases) ///
                        xtitle("") ///
                        tlabel(22jan2020(15)31aug2022 , labsize(1.5) angle(45) grid)

twoway line deaths datevar if country == "US", title(Cumulative COVID-19 deaths in the United States) ///
                        ylabel( , labsize(2) format(%11.0fc)) ///
                        ytitle(# of deaths) ///
                        xtitle("") ///
                        tlabel(22jan2020(15)31aug2022 , labsize(1.5) angle(45) grid)

* See the assigned number to variable country1
* For more info type: help labelbook

labelbook country1

twoway line cases datevar if country1 == 263, ///
                        title({bf:Cumulative} COVID-19 cases in the {it:United States}) ///
                        ylabel( , labsize(2) format(%11.0fc)) ///
                        ytitle(# of cases) ///
                        xtitle("") ///
                        tlabel(22jan2020(15)31aug2022 , labsize(1.5) angle(45) format(%tdm_d,_y) grid)

* For more info see:
*https://www.stata.com/bookstore/pdf/g_text.pdf
*https://www.stata.com/features/overview/symbols-and-multiple-fonts/

* Compare with another country/region

twoway line cases datevar if country == "US" || ///
       line cases datevar if country == "United Kingdom", title(Cumulative COVID-19 cases) ///
                        ylabel( , labsize(2) format(%11.0fc)) ///
                        ytitle(# of cases) ///
                        xtitle("")  ///
                        tlabel(22jan2020(15)31aug2022 , labsize(1.5) angle(45) grid) ///
                        legend(label(1 "United States") label(2 "United Kingdom"))

* Accounting for population (per 100k)

table country1 if datevar == td(22jan2020), statistic(sum pop)
```

```
491
492  gen cases100k = cases / (pop/100000)
493
494  gen deaths100k = deaths / (pop/100000)
495
496  twoway line cases100k datevar if country == "US", lcolor(black)  || ///
497         line cases100k datevar if country == "United Kingdom", lcolor(red) || ///
498         line cases100k datevar if country == "Brazil", lcolor(green) ///
499                           title("Cumulative COVID-19 cases per 100,000 people") ///
500                           ylabel( , labsize(2) format(%11.0fc)) ///
501                           ytitle(# of cases) ///
502                           xtitle("") ///
503                           tlabel(22jan2020(15)31aug2022 , labsize(1.5) angle(45) grid)  ///
504                           legend(label(1 "United States") label(2 "United Kingdom") label(3 "Brazil"))
505
506  twoway line deaths100k datevar if country == "US", lcolor(black)  || ///
507         line deaths100k datevar if country == "United Kingdom", lcolor(red) || ///
508         line deaths100k datevar if country == "Brazil", lcolor(green) ///
509                           title("Cumulative COVID-19 deaths per 100,000 people") ///
510                           ylabel( , labsize(2) format(%11.0fc)) ///
511                           ytitle(# of cases) ///
512                           xtitle("") ///
513                           tlabel(22jan2020(15)31aug2022 , labsize(1.5) angle(45) grid)  ///
514                           legend(label(1 "United States") label(2 "United Kingdom") label(3 "Brazil"))
515
516  * For some examples see:
517  * https://www.princeton.edu/~otorres/Visual101.pdf
518
519  * Get daily counts
520
521  * Using time series operators
522  * Accounts for the panel data structure
523
524  xtset country1 datevar
525
526  * Using time series operators
527  * See https://www.princeton.edu/~otorres/TS101.pdf
528  * See also option [_n-1] in: help _n
529
530  gen daily_cases = cases - L1.cases
531  gen daily_cases100k = cases100k - L1.cases100k
532
533  gen daily_deaths = deaths - L1.deaths
534  gen daily_deaths100k = deaths100k - L1.deaths100k
535
536  browse country1 datevar cases daily_cases
537
538  sum daily_cases daily_cases100k daily_deaths daily_deaths100k
539
540  replace daily_cases = 0 if daily_cases < 0
541  replace daily_cases100k = 0 if daily_cases100k <0
542
543  replace daily_deaths = 0 if daily_deaths <0
544  replace daily_deaths100k = 0 if daily_deaths100k <0
545
546  sum daily_cases daily_cases100k daily_deaths daily_deaths100k
547
548  twoway line daily_cases datevar if country == "US", title(Daily COVID-19 cases in the United States) ///
549                           ylabel( , labsize(3) format(%11.0fc)) ///
550                           ytitle(# of cases) ///
551                           xtitle("") ///
552                           tlabel(22jan2020(15)31aug2022 , labsize(1.5) angle(45) grid)
553
554  twoway line daily_deaths datevar if country == "US", title(Daily COVID-19 deaths in the United States) ///
555                           ylabel( , labsize(3) )
556                           ytitle(# of deaths) ///
557                           xtitle("") ///
558                           tlabel(22jan2020(15)31aug2022 , labsize(1.5) angle(45) grid)
559  * 7-day moving average
560
```

```
* Manual estimation
*gen daily7_deaths_m = (daily_deaths + l1.daily_deaths + l2.daily_deaths + l3.daily_deaths + l4.daily_deaths + l5.daily_deaths + l6.daily_deaths)/7

* Using user-written add-on egenmore
* For more info type: help egenmore
* See also: help egen

ssc install egenmore, replace

xtset country1 datevar

egen daily7_cases = filter(daily_cases), lags(0/6) normalise
egen daily7_cases100k = filter(daily_cases100k), lags(0/6) normalise

egen daily7_deaths = filter(daily_deaths), lags(0/6) normalise
egen daily7_deaths100k = filter(daily_deaths100k), lags(0/6) normalise

browse country1 datevar daily_deaths daily7_deaths

twoway line daily7_cases datevar if country == "US", ///
                        title(7-day moving average COVID-19 cases) ///
                        subtitle(United States) ///
                        ylabel( , labsize(3) format(%11.0fc)) ytitle(# of cases) ///
                        tlabel(22jan2020(15)31aug2022 , labsize(1.5) angle(45) grid) xtitle("") ///
                        tline(20jan2021) ///
                        ttext(700000 20jul2020 "Trump") ///
                        ttext(700000 20jul2021 "Biden")

twoway line daily7_deaths datevar if country == "US", ///
                        title(7-day moving average COVID-19 deaths) ///
                        subtitle(United States) ///
                        ylabel( , labsize(3) format(%11.0fc)) ytitle(# of deaths) ///
                        tlabel(22jan2020(15)31aug2022 , labsize(1.5) angle(45) grid) xtitle("") ///
                        tline(20jan2021) ///
                        ttext(3500 20jul2020 "Trump") ///
                        ttext(3500 20jul2021 "Biden")

twoway line daily7_deaths100k datevar if country == "US" || ///
       line daily7_deaths100k datevar if country == "United Kingdom", ///
                        title(7-day moving average COVID-19 deaths per 100k people) ///
                        subtitle(United States and United Kingdom) ///
                        ylabel( , labsize(3) format(%11.0fc)) ///
                        ytitle(# of deaths) ///
                        xtitle("") ///
                        tlabel(22jan2020(15)31aug2022 , labsize(1.5) angle(45) grid)  ///
                        legend(label(1 "United States") label(2 "United Kingdom" ))

* Percentage change

gen daily7_casesgr = ((daily7_cases - l1.daily7_cases) / l1.daily7_cases) * 100

gen daily7_deathsgr = ((daily7_deaths - l1.daily7_deaths) / l1.daily7_deaths) * 100

browse country1 datevar daily7_cases daily7_casesgr

twoway line daily7_deathsgr datevar if country == "US", ///
                        title(Daily percentage growth COVID-19 deaths) ///
                        subtitle(United States) ///
                        ylabel( , labsize(3) format(%11.0fc)) ///
                        ytitle(%) ///
                        xtitle("") ///
                        tlabel(22jan2020(15)31aug2022 , labsize(1.5) angle(45) grid)  ///
                        tline(20jan2021) ///
                        ttext(450 20jul2020 "Trump") ///
                        ttext(450 20jul2021 "Biden")

* Scatterplots

twoway scatter daily7_cases100k gdppc
```

```
630
631 twoway scatter daily7_cases100k gdppc, jitter(13) ///
632                                     xlabel(0(5000)112000 , labsize(2) angle(45) format(%11.0fc) grid) ///
633                                     ylabel(, labsize(2) angle(45) format(%11.0fc))
634
635 twoway scatter daily7_cases100k gdppc, jitter(13) ///
636                                     xlabel(0(5000)112000 , labsize(2) angle(45) format(%11.0fc) grid) ///
637                                     ylabel(, labsize(2) angle(45) format(%11.0fc)) || ///
638         lfit daily7_cases100k gdppc, lcolor(white)
639
640 twoway scatter daily7_deaths100k gdppc, jitter(13) ///
641                                   xlabel(0(5000)112000 , labsize(2) angle(45) format(%11.0fc) grid) || ///
642         lfit daily7_deaths100k gdppc, lcolor(white)
643
644 twoway scatter daily7_deaths100k gdppc, jitter(13) ///
645                           xlabel(0(5000)112000 , labsize(2) angle(45) format(%11.0fc) grid) ///
646                           msymbol(none) mlabel(country) mlabsize(1.5) mlabposition(0) || ///
647         lfit daily7_deaths100k gdppc, lcolor(white) legend(off)
648
649 * Scatterplot with trust in goverment
650
651 twoway scatter daily7_deaths100k trust, jitter(13)  || ///
652        lfit daily7_deaths100k trust, lcolor(white) ///
653                                     legend(off) ///
654                                     ytitle(7-day MA deaths per 100k) ///
655                                     xtitle("Trust in goverment (a great deal + quite a lot)")
656
657 * Correlation
658 * For more info type: help pwcorr
659 * See also command: corr
660
661 pwcorr daily7_cases100k daily7_deaths100k gdppc trust, star(0.05) sig
662
663
664 * Recoding a variable
665
666 * Get detailed summary statistics
667
668 sum gdppc
669
670 sum gdppc if datevar == td(31aug2022), detail
671
672 * Recoding a variable into a new one
673 * For more info type: help recode
674
675 recode gdppc (0/2000 = 1 "Less 2k") ///
676              (2000.001/16000 = 2 ">2k-16k") ///
677              (16000.001/43000 = 3 ">16k-43k") ///
678              (43000.001/56000 = 4 ">43k-56k") ///
679              (56000.001/111000 = 5 ">56k") ///
680              (else = .), gen(income_group) label(income_group)
681
682 label variable income_group "GDPPC recoded"
683
684 tab income_group
685
686 tab income_group, nolabel
687
688 twoway scatter daily7_cases100k income_group, jitter(13)  xlabel( , valuelabel) ///
689                                               xtitle(Income groups) mlabel(country)  ///
690                                               mlabsize(1.5) msymbol(none) mlabposition(0)
691
692 twoway scatter daily7_deaths100k income_group, jitter(13) xlabel( , valuelabel) ///
693                                                xtitle(Income groups) mlabel(country)  ///
694                                                mlabsize(1.5) msymbol(none) mlabposition(0)
695
696 * Box plots. This takes longer
697 * For more info type: help graph box
698 *graph box daily7_deaths100k, over(income_group) marker(1, mlabel(country) msize(1.5))
699
```

```
* Is there a significant difference across groups?
* One-way Anova
* Null hypothesis: all means are equal
* Alternative hypothesis: at least one is not
* For more info type: help oneway

oneway daily7_cases100k income_group, tabulate bonferroni

oneway daily7_deaths100k income_group, tabulate bonferroni

table income_group, statistic(mean daily7_cases100k) statistic(mean daily7_deaths100k)

* ttest
* Null hypothesis: mean1 = mean2
* Alternative hypothesis: mean1 not equal to mean2 (two-sided)
* For more info type: help ttest

ttest daily7_cases100k if income_group == 3 | income_group == 4, by(income_group)

ttest daily7_deaths100k if income_group == 4 | income_group == 5, by(income_group)


* Regression
* reg output independent(s), robust
* For more info type: help reg

reg daily7_cases100k gdppc, robust

* Histograms
* For more info type: help hist

hist daily7_deaths100k

hist gdppc

hist trust

twoway scatter daily7_deaths100k gdppc

gen lngdppc = ln(gdppc)

gen lndaily7_deaths100k = ln( daily7_deaths100k + 1)

gen lndaily7_cases100k = ln( daily7_cases100k + 1)

twoway scatter lndaily7_deaths100k lngdppc

reg lndaily7_deaths100k lngdppc, robust

reg lndaily7_deaths100k lngdppc trust, robust

reg lndaily7_deaths100k i.income_group, robust

reg lndaily7_deaths100k i.income_group trust, robust


* Export regression results to a nice table using outreg2
* For more info type: help outreg2

ssc install outreg2, replace

reg lndaily7_deaths100k lngdppc, robust

outreg2 using myreg.doc, replace

reg lndaily7_deaths100k lngdppc trust, robust

outreg2 using myreg.doc, append

```

```
* If when using outreg2 you get the following error, make sure to close the word or excel document firstrow
* "file using "myreg.doc" is read-only; cannot be modified or erased
*  The file needs to be closed if being used by another software such as Word."

* For all other procedures, you can use the user-created command "asdoc"
* For more info type: help asdoc

ssc install asdoc, replace

asdoc ttest daily7_cases100k if income_group == 3 | income_group == 4, by(income_group) save(myresults.doc) replace

asdoc ttest daily7_deaths100k if income_group == 4 | income_group == 5, by(income_group) save(myresults.doc)

asdoc pwcorr daily7_cases100k daily7_deaths100k gdppc income_group trust, star(0.05) sig save(myresults.doc)


* If when using asdoc you get the following error, make sure to close the word document firstrow
* "file ttest.doc could not be opened
*                 fopen():   603  file could not be opened
*            asdoctable():     -  function returned error
*                <istmt>:     -  function returned error"


* Close the log

log close


*********************************************
* ONLINE RESOURCES
*********************************************
/*
Stata documentation - https://www.stata.com/features/documentation/
Stata tutorials - http://www.princeton.edu/~otorres/
*/
```