```
/*******************************************************************************
1. Master File Read
*******************************************************************************/
clear all
set more off
cd  "<<insert input directory>>"
import excel  "Master_Procurement_Data", sheet("all data") firstrow clear
cd  "<<insert output directory>>"


*Configuration
ssc install aaplot
ssc install estout
ssc install extremes

*Creation of Transparency Variables
foreach var of varlist Q20AwardLaw Q20ContractLaw Q20DocumentsLaw Q20ModelsLaw
Q20NoticeLaw Q20PlansLaw Q45FeaturesDisclosureonly Q20AwardPractice
Q20ContractPractice Q20Documentspractice Q20ModelsPractice Q20NoticePractice
Q20PlansPractice Q50RenegPublicity {
        replace `var' = "0" if `var' == "No"
    replace `var' = "1" if `var' == "Yes"
        destring `var', replace
}

*Creation of Competition Variables
foreach var of varlist Q14Open Q15Prequalification Q18Dividing Q21Preparation
Q28BidOpening Q30Standstill Q15aPrequalification Bid_opening
complaints_award_first_suspensio {
        replace `var' = "0" if `var' == "No"
    replace `var' = "1" if `var' == "Yes"
        destring `var', replace
}

replace Q16Openpractice = "1" if Q16Openpractice == "Open is default" |
Q16Openpractice == "Open is most common"
replace Q16Openpractice = "0" if Q16Openpractice == "Other"
destring Q16Openpractice, replace

foreach var of varlist abuse_PE_ad Q18a {
        replace `var' = "0" if `var' == "Occasionally" | `var' == "Often" | `var' ==
"Very Often"
    replace `var' = "1" if `var' == "Very Rarely" | `var' == "Rarely"
        destring `var', replace
}

* Creation of Limits to Exclusion Variables
foreach var of varlist Q22ContentDocs Q38Abnormallylowbids Q39Errors
Q39cErrorspractice {
        replace `var' = "0" if `var' == "No"
```

```
    replace `var' = "1" if `var' == "Yes"
        destring `var', replace
}

replace Q25Clarifications = "1" if Q25Clarifications == "The procuring entity
addresses all clarifications in a public meeting."
replace Q25Clarifications = "0" if Q25Clarifications == "The procuring entity will
answer, and it is always required to communicate the answer to all other bidders
too." | Q25Clarifications == "The procuring entity will answer, but it is not always
required to communicate the answer to all other bidders." | Q25Clarifications ==
"The procuring entity will only answer to the relevant bidder." | Q25Clarifications
== "Other"
destring Q25Clarifications, replace

replace Q35AwardCriteria = "1" if Q35AwardCriteria == "Price"
replace Q35AwardCriteria = "0" if Q35AwardCriteria == "Price and qualitative
elements" | Q35AwardCriteria == "Discretion of the Procuring Entity"
destring Q35AwardCriteria, replace

foreach var of varlist abuse_PE_tech_specifications clarification_circumvent_informa
abuse_COMP_low_bids10 {
        replace `var' = "0" if `var' == "Occasionally" | `var' == "Often" | `var' ==
"Very Often"
    replace `var' = "1" if `var' == "Very Rarely" | `var' == "Rarely"
        destring `var', replace
}

replace eval_price_only_frequency = "0" if eval_price_only_frequency == "Very
Rarely" | eval_price_only_frequency == "Rarely" | eval_price_only_frequency ==
"Occasionally"
replace eval_price_only_frequency = "1" if eval_price_only_frequency == "Often" |
eval_price_only_frequency == "Very Often"
destring eval_price_only_frequency, replace

*Creation of Integrity Variables
foreach var of varlist Q45Additional10 Q58Paymentbylaw Q58eLatePaymentinterest {
        replace `var' = "0" if `var' == "No"
    replace `var' = "1" if `var' == "Yes"
        destring `var', replace
}

replace Q12Certificate = "0" if Q12Certificate == "No"
replace Q12Certificate = "1" if Q12Certificate == "Yes" | Q12Certificate == "Yes,
there is a specific budget allocation" | Q12Certificate == "Yes, a certificate is
required"
destring Q12Certificate, replace

replace Q23Subcontracting = "0" if Q23Subcontracting == "No" | Q23Subcontracting ==
"Liability." | Q23Subcontracting == "Features." | Q23Subcontracting == "Features,
liability." | Q23Subcontracting == "Features, disclosure." | Q23Subcontracting ==
```

```
"Disclosure." | Q23Subcontracting == "Disclosure, liability."
replace Q23Subcontracting = "1" if Q23Subcontracting == "Features, disclosure,
liability."
destring Q23Subcontracting, replace

replace Q45Renegotiation10 = "0" if Q45Renegotiation10 == "No"
replace Q45Renegotiation10 = "1" if Q45Renegotiation10 == "Features, disclosure,
liability." | Q45Renegotiation10 == "Liability." | Q45Renegotiation10 == "Features."
| Q45Renegotiation10 == "Features, liability." | Q45Renegotiation10 == "Features,
disclosure." | Q45Renegotiation10 == "Disclosure." | Q45Renegotiation10 ==
"Disclosure, liability."
destring Q45Renegotiation10, replace

foreach var of varlist Q13 abuse_COMP_subcontractors abuse_COMP_renegotiation{
        replace `var' = "0" if `var' == "Occasionally" | `var' == "Often" | `var' ==
"Very Often"
    replace `var' = "1" if `var' == "Very Rarely" | `var' == "Rarely"
        destring `var', replace
}

foreach var of varlist Q58c Q58f{
        replace `var' = "1" if `var' == "Often" | `var' == "Very Often"
    replace `var' = "0" if `var' == "Very Rarely" | `var' == "Rarely" | `var' ==
"Occasionally"
        destring `var', replace
}

*Creation of Quality of Product and Integrity of Process Variables
foreach var of varlist Overrun {
        replace `var' = "0.05" if `var' == "Very Often"
        replace `var' = "0.3" if `var' == "Often"
    replace `var' = "0.625" if `var' == "Occasionally"
        replace `var' = "0.825" if `var' == "Rarely"
        replace `var' = "0.95" if `var' == "Very Rarely"
        destring `var', replace
}

foreach var of varlist Low_quality Favoritism Corruption Collusion No_competition {
        replace `var' = "0.95" if `var' == "Very Often"
        replace `var' = "0.7" if `var' == "Often"
    replace `var' = "0.375" if `var' == "Occasionally"
        replace `var' = "0.175" if `var' == "Rarely"
        replace `var' = "0.05" if `var' == "Very Rarely"
        destring `var', replace
}

*Construct Laws and Pratcies
egen Transparency_l = rowmean(Q20AwardLaw Q20ContractLaw Q20DocumentsLaw
Q20ModelsLaw Q20NoticeLaw Q20PlansLaw Q45FeaturesDisclosureonly)
egen Transparency_p = rowmean(Q20AwardPractice Q20ContractPractice
```

```
Q20Documentspractice Q20ModelsPractice Q20NoticePractice Q20PlansPractice
Q50RenegPublicity)
egen Competition_l = rowmean(Q14Open Q15Prequalification Q18Dividing Q21Preparation
Q28BidOpening Q30Standstill)
egen Competition_p = rowmean(Q16Openpractice Q15aPrequalification Q18a abuse_PE_ad
Bid_opening complaints_award_first_suspensio)
egen Limits_to_Exclusion_l = rowmean(Q22ContentDocs Q25Clarifications
Q35AwardCriteria Q38Abnormallylowbids Q39Errors)
egen Limits_to_Exclusion_p = rowmean(abuse_PE_tech_specifications
clarification_circumvent_informa eval_price_only_frequency abuse_COMP_low_bids10
Q39cErrorspractice)
egen Integrity_of_contract_l = rowmean(Q12Certificate Q23Subcontracting
Q45Additional10 Q45Renegotiation10 Q58Paymentbylaw Q58eLatePaymentinterest)
egen Integrity_of_contract_p = rowmean(Q13 abuse_COMP_subcontractors
Q55AdditionalWorksP abuse_COMP_renegotiation Q58c Q58f)

egen laws=rowtotal(Transparency_l Competition_l Limits_to_Exclusion_l
Integrity_of_contract_l)
egen practices=rowtotal(Transparency_p Competition_p Limits_to_Exclusion_p
Integrity_of_contract_p)

*Quality Outcome
foreach var in Time Overrun Low_quality {
        egen `var'_z=std(`var')
}
egen quality=rowmean(Time_z Overrun_z Low_quality_z )
replace quality=-1*quality //So that for quality, high is good.

*Integrity Outcome
foreach var in Favoritism  Corruption  Collusion No_competition {
        replace `var' =-1* `var'   //So that high is good.
}
foreach var in Favoritism Corruption  Collusion  No_competition {
        egen `var'_z=std(`var')
}
egen integrity=rowmean(Favoritism_z Corruption_z Collusion_z  No_competition_z)

foreach var in BribesGTI favoritismGTI{
        egen `var'_z=std(`var')
}
egen GTI_integrity=rowmean(BribesGTI_z favoritismGTI_z)

drop  BribesGTI_z favoritismGTI_z Favoritism_z Corruption_z Collusion_z
No_competition_z

drop if Economy == ""


/*******************************************************************************
2. Charts and Tables
```

```
*********************************************************************************/

*HCI -> split
centile HCI2020,centile(50)
gen edu=1 if HCI2020<=r(c_1)  & HCI2020!=.  //Low education .
replace edu=2 if HCI2020>r(c_1) & HCI2020!=.   //More education

gen labelclock = 3

/* Summary Stats */
extremes law Economy
extremes prac Economy

extremes quality Economy
extremes integrity Economy, n(20)

reg prac law loggdp
corr law prac

/*T-test for HCI high and low: Coefficient on laws predicting Integrity*/
/* "The coefficient on laws predicting Integrity is not different between high and
low human capital samples, with a t-statistics of -1.82; the coefficient on laws
predicting Quality is not different between the two samples either, with a
t-statistic of -1.44." */
gen HCI_high = 1 if edu == 2
replace HCI_high = 0 if edu == 1
regress inte laws HCI_high i.HCI_high#c.laws
regress quality laws HCI_high i.HCI_high#c.laws
drop HCI_high

/* Table 3 */
est drop _all
estpost correlate integrity quality Rquality Aspeed loggdp law practice, matrix
est store c1
esttab c1 using "Table 3.csv", replace postfoot("") nomtitle nonumbers nodepvars
nonotes unstack  not  compress  star(* 0.1 ** 0.05 *** 0.01)

/* Figures 1- 8 */
*1.
aaplot laws loggdp   , mlabel(countrycode) mlabvpos(labelclock) mlabsize(vsmall)
ytitle("Laws")  xtitle("Log GDP")  lopts(lc(blue)) aformat(%7.0g) bformat(%7.0g)
rsqformat(%4.0g) rmseformat(%4.3f) subtitle("", size(vsmall))
gr export "Figure 1.png",replace

*2.
aaplot practice loggdp   , mlabel(countrycode) mlabvpos(labelclock) mlabsize(vsmall)
ytitle("Practice")  xtitle("Log GDP")  lopts(lc(blue)) aformat(%7.0g) bformat(%7.0g)
rsqformat(%4.0g) rmseformat(%4.3f) subtitle("", size(vsmall))
gr save "Figure 2",replace
gr export "Figure 2.png",replace
```

```
*3. Practice
aaplot integrity practice , mlabel(countrycode) mlabvpos(labelclock)
mlabsize(vsmall) xtitle("Practice")  ytitle("Integrity")  abbrev(11) lopts(lc(blue))
aformat(%7.0g) bformat(%7.0g) rsqformat(%4.0g) rmseformat(%4.3f) subtitle("",
size(vsmall))  yla(-3(1)2)
gr save "Figure 3",replace
gr export "Figure 3.png",replace

aaplot  quality practice  , mlabel(countrycode) mlabvpos(labelclock)
mlabsize(vsmall) xtitle("Practice")  ytitle("Quality")   abbrev(11) lopts(lc(blue))
aformat(%7.0g) bformat(%7.0g) rsqformat(%4.0g) rmseformat(%4.3f) subtitle("",
size(vsmall)) yla(-3(1)2)
gr save "Figure 4",replace
gr export "Figure 4.png",replace

*4. Law
aaplot  integrity law  , mlabel(countrycode) mlabvpos(labelclock) mlabsize(vsmall)
xtitle("Laws")  ytitle("Integrity")  abbrev(11)  lopts(lc(blue)) aformat(%7.0g)
bformat(%7.0g) rsqformat(%4.0g) rmseformat(%4.3f) subtitle("", size(vsmall))
yla(-3(1)2)
gr save "Figure 5",replace
gr export "Figure 5.png",replace

aaplot  quality law  , mlabel(countrycode) mlabvpos(labelclock) mlabsize(vsmall)
xtitle("Laws")  ytitle("Quality") abbrev(11) lopts(lc(blue)) aformat(%7.0g)
bformat(%7.0g) rsqformat(%4.0g) rmseformat(%4.3f) subtitle("", size(vsmall))
yla(-3(1)2)
gr save "Figure 6",replace
gr export "Figure 6.png",replace

*Quality:
reg quality laws Rquality Aspeed loggdp
est store product1
*Process:
reg integrity laws Rquality Aspeed loggdp
est store process1

esttab process1 product1  using "item 4.csv", se r2 replace nogap nocons nonumbers
unstack   compress  star(* 0.1 ** 0.05 *** 0.01)

*5. Splitting the sample: We measure the level of public sector capacity with either
human capital or a survey measure of government efficiency
aaplot  quality law if edu==1 , mlabel(countrycode) mlabvpos(labelclock)
mlabsize(vsmall) xtitle("Laws")  ytitle("Quality (Less Education)") abbrev(11)
lopts(lc(blue)) aformat(%7.0g) bformat(%7.0g) rmseformat(%4.3f) subtitle("",
size(vsmall)) yla(-2(1)2)
gr save "Figure 7 ", replace
gr export "Figure 7.png",replace
aaplot  quality law if edu==2 , mlabel(countrycode) mlabvpos(labelclock)
```

```
mlabsize(vsmall) xtitle("Laws")  ytitle("Quality (More Education)") abbrev(11)
lopts(lc(blue)) aformat(%7.0g) bformat(%7.0g) rmseformat(%4.3f) subtitle("",
size(vsmall)) yla(-2(1)2)
gr save "Figure 7more", replace
gr export "Figure 7more.png",replace

aaplot  integrity law if edu==1 , mlabel(countrycode) mlabvpos(labelclock)
mlabsize(vsmall) xtitle("Laws")  ytitle("Integrity (Less Education)") abbrev(11)
lopts(lc(blue)) aformat(%7.0g) bformat(%7.0g) rmseformat(%4.3f) subtitle("",
size(vsmall))
gr save "Figure 8", replace
gr export "Figure 8.png",replace
aaplot  integrity law if edu==2 , mlabel(countrycode) mlabvpos(labelclock)
mlabsize(vsmall) xtitle("Laws")  ytitle("Integrity (More Education)") abbrev(11)
lopts(lc(blue)) aformat(%7.0g) bformat(%7.0g) rmseformat(%4.3f) subtitle("",
size(vsmall))
gr save "Figure 8more", replace
gr export "Figure 8more.png",replace


/* Appendix A */
* Table  A1
preserve
est drop _all
drop if AvgSizeProcuredProjects == .
estpost tabstat AvgSizeProcuredProjects, by(income_wb) statistics(count mean p50 sd
min max) columns(statistics) listwise
esttab using "Table A1.csv", cells("count mean p50 sd min max") nomtitle nonumber
noobs replace
est drop _all
restore

* Figure A1: See Excel file in the package.

* Figure A2-A3
preserve
drop if Economy == "Australia" |  Economy == "Bulgaria" |  Economy == "Cyprus" |
Economy == "Czech Republic" |  Economy == "Finland" |  Economy == "France" |
Economy == "Germany" |  Economy == "Iceland" |  Economy == "Ireland" |  Economy ==
"Italy" |  Economy == "Luxembourg" |  Economy == "Netherlands" |  Economy ==
"Norway" |  Economy == "Switzerland" |  Economy == "United Kingdom" |  Economy ==
"United States"

aaplot  quality law if edu==2 , mlabel(countrycode) mlabvpos(labelclock)
mlabsize(vsmall) xtitle("Laws")  ytitle("Quality (More Education)") abbrev(11)
lopts(lc(blue)) aformat(%7.0g) bformat(%7.0g) rmseformat(%4.3f) subtitle("",
size(vsmall))
gr save "Figure A2_more", replace
gr export "Figure A2_more.png",replace
```

```
aaplot  integrity law if edu==2 , mlabel(countrycode) mlabvpos(labelclock)
mlabsize(vsmall) xtitle("Laws")  ytitle("Integrity (More Education)") abbrev(11)
lopts(lc(blue)) aformat(%7.0g) bformat(%7.0g) rmseformat(%4.3f) subtitle("",
size(vsmall))
gr save "Figure A3_more", replace
gr export "Figure A3_more.png",replace
restore

* Figure A4: See Excel file in the package.

* Figure A5-A6
preserve

replace laws =  nonbinding_laws if !mi(nonbinding_laws)

aaplot  quality laws if edu==2 , mlabel(countrycode) mlabvpos(labelclock)
mlabsize(vsmall) xtitle("Laws")  ytitle("Quality (More Education)") abbrev(11)
lopts(lc(blue)) aformat(%7.0g) bformat(%7.0g) rmseformat(%4.3f) subtitle("",
size(vsmall))
gr save "Figure A5_more", replace
gr export "Figure A5_more.png",replace

aaplot integrity laws if edu==2 , mlabel(countrycode) mlabvpos(labelclock)
mlabsize(vsmall) xtitle("Laws")  ytitle("Integrity (More Education)") abbrev(11)
lopts(lc(blue)) aformat(%7.0g) bformat(%7.0g) rmseformat(%4.3f) subtitle("",
size(vsmall))
gr save "Figure A6_more", replace
gr export "Figure A6_more.png",replace

restore

*Figure A7. Road Quality and Practice;
aaplot  Aspeed practice, mlabel(countrycode) mlabvpos(labelclock) mlabsize(vsmall)
xtitle("Practice")  ytitle("Average Speed") abbrev(11) lopts(lc(blue))
aformat(%7.0g) bformat(%7.0g) rmseformat(%4.3f) subtitle("", size(vsmall))
gr save "Figure A7_1", replace
gr export "Figure A7_1.png",replace

aaplot  Rquality practice, mlabel(countrycode) mlabvpos(labelclock) mlabsize(vsmall)
xtitle("Practice")  ytitle("Road Quality") abbrev(11) lopts(lc(blue)) aformat(%7.0g)
bformat(%7.0g) rmseformat(%4.3f) subtitle("", size(vsmall))
gr save "Figure A7_2", replace
gr export "Figure A7_2.png",replace

aaplot  GTI_integrity practice, mlabel(countrycode) mlabvpos(labelclock)
mlabsize(vsmall) xtitle("Practice")  ytitle("Integrity GTI") abbrev(11)
lopts(lc(blue)) aformat(%7.0g) bformat(%7.0g) rmseformat(%4.3f) subtitle("",
size(vsmall))
gr save "Figure A7_3", replace
gr export "Figure A7_3.png",replace
```

```
*Figure A8. Road Quality and Laws
aaplot  Aspeed law, mlabel(countrycode) mlabvpos(labelclock) mlabsize(vsmall)
xtitle("Laws")  ytitle("Average Speed") abbrev(11) lopts(lc(blue)) aformat(%7.0g)
bformat(%7.0g) rmseformat(%4.3f) subtitle("", size(vsmall))
gr save "Figure A8_1", replace
gr export "Figure A8_1.png",replace

aaplot  Rquality law, mlabel(countrycode) mlabvpos(labelclock) mlabsize(vsmall)
xtitle("Laws")  ytitle("Road Quality") abbrev(11) lopts(lc(blue)) aformat(%7.0g)
bformat(%7.0g) rmseformat(%4.3f) subtitle("", size(vsmalllaw ))
gr save "Figure A8_2", replace
gr export "Figure A8_2.png",replace

aaplot  GTI_integrity law, mlabel(countrycode) mlabvpos(labelclock) mlabsize(vsmall)
xtitle("Laws")  ytitle("Integrity GTI") abbrev(11) lopts(lc(blue)) aformat(%7.0g)
bformat(%7.0g) rmseformat(%4.3f) subtitle("", size(vsmalllaw ))
gr save "Figure A8_3", replace
gr export "Figure A8_3.png",replace

/* Appendix B Tables.*/
* B1
reg integrity laws HCI2020 if edu==1  //Low education .
est store process1
reg quality laws HCI2020 if edu==1
est store product1

reg integrity laws HCI2020 if edu==2
est store process2
reg quality laws HCI2020 if edu==2
est store product2

esttab process1 product1 process2   product2   using "Table B1.csv", se r2 replace
nogap nocons nonumbers  unstack   compress  star(* 0.1 ** 0.05 *** 0.01)

est drop _all

* B2
preserve
bys edu: sum law
bys edu: sum prac
ttest law==practi if edu == 1
        gen m_l_edu1=r(mu_1)
        gen se_l_edu1=r(sd_1)/sqrt(r(N_1))
        gen m_p_edu1=r(mu_2)
        gen se_p_edu1=r(sd_2)/sqrt(r(N_2))
        gen t_edu1=r(t)
        gen p_edu1=r(p)

ttest law==practi if edu == 2
```

```
        gen m_l_edu2=r(mu_1)
        gen se_l_edu2=r(sd_1)/sqrt(r(N_1))
        gen m_p_edu2=r(mu_2)
        gen se_p_edu2=r(sd_2)/sqrt(r(N_2))
        gen t_edu2=r(t)
        gen p_edu2=r(p)

gen diff=practices-laws
ttest diff ,by(edu)
        gen t_diff=-1*r(t)
        gen p_diff=r(p_u)

keep m_* se_* t_* p_*
duplicates drop
export excel using "Table B2.xlsx", firstrow(variables) replace

restore

* B3
reg integrity laws HCI
est store process1
reg integrity laws  GovEff
est store process2
reg integrity laws HCI GovEff
est store process3
reg quality laws HCI
est store product1
reg quality laws  GovEff
est store product2
reg quality laws HCI GovEff
est store product3

esttab process1 process2 process3 product1 product2 product3  using "Table B3.csv",
se r2 replace nogap nocons nonumbers  unstack   compress  star(* 0.1 ** 0.05 ***
0.01)
est drop _all

* B4
qui sum HCI
gen HCI_demeaned=HCI-r(mean)

qui sum laws
gen laws_demeaned=laws-r(mean)

qui sum GovEff
gen GovEff_demeaned=GovEff-r(mean)

gen HCIlaws=HCI_demeaned*laws_demeaned
gen WGIlaws=GovEff_demeaned*laws_demeaned
```

```
reg integrity laws HCI2020 HCIlaws
est store process1
reg integrity laws HCI2020 HCIlaws GovEffWB loggdp
est store process2
reg integrity laws HCI2020 GovEffWB WGIlaws loggdp
est store process3
reg quality laws HCI2020 HCIlaws
est store product1
reg quality laws HCI2020 HCIlaws GovEffWB  loggdp
est store product2
reg quality laws HCI2020 GovEffWB WGIlaws loggdp
est store product3

esttab process1 process2 process3 product1 product2 product3 using "Table B4.csv",
se r2 replace nogap nocons nonumbers  unstack   compress  star(* 0.1 ** 0.05 ***
0.01)
est drop _all
drop HCIlaws HCI_demeaned laws_demeaned GovEff_demeaned

/* Appendix C Tables.*/
* C1 and C2
corr law loggdp
corr prac loggdp

est drop _all
estpost correlate Transparency_l Competition_l Integrity_of_contract_l
Limits_to_Exclusion_l  loggdp law, matrix
est store c1
esttab c1 using "Table C1.csv", replace postfoot("") nomtitle nonumbers nodepvars
nonotes unstack  not  compress  star(* 0.1 ** 0.05 *** 0.01)
est drop _all
estpost correlate Transparency_p Competition_p Integrity_of_contract_p
Limits_to_Exclusion_p  loggdp prac, matrix
est store c1
esttab c1 using "Table C2.csv", replace postfoot("") nomtitle nonumbers nodepvars
nonotes unstack  not  compress  star(* 0.1 ** 0.05 *** 0.01)

* C3
global Theme_l "Transparency_l Competition_l Limits_to_Exclusion_l
Integrity_of_contract_l"
global Theme_p "Transparency_p Competition_p Limits_to_Exclusion_p
Integrity_of_contract_p"

est drop _all
local i=0
foreach outcome in Corruption integrity quality   {
        local i=`i'+1
        estpost correlate `outcome' laws  $Theme_l
        est store c`i'_l
        estpost correlate `outcome' practice $Theme_p
```

```
        est store c`i'_p
}
esttab c1_l c1_p c2_l c2_p c3_l c3_p using "Table C3.csv",replace nogap noobs nocons
nonumbers  unstack not   compress  star(* 0.1 ** 0.05 *** 0.01)

* C4
rename HCI2020 HCI
centile HCI,centile(50)
gen HCI_group=1 if HCI>r(c_1) &!mi(HCI)  //HIGH  education
replace HCI_group=2 if HCI<=r(c_1) & HCI!=.  //LOW education

local iclass `" "==1" "==2" "'
local iclasswords `""Above median HCI" "Below median HCI" "'
tokenize `"`iclasswords'"'

global theme "Transparency Competition Limits_to_Exclusion Integrity_of_contract"

preserve
{
rename laws laws_l
rename practices laws_p

local i=0
qui foreach class of local iclass {
        local i=`i'+1
        qui foreach var in laws $theme {
                ttest `var'_l==`var'_p if HCI_group`class'
                ret list
                local t_`var'_`i'=r(t)
                local m_l_`var'_`i'=r(mu_1)
                local se_l_`var'_`i'=r(sd_1)/sqrt(r(N_1))
                local m_p_`var'_`i'=r(mu_2)
                local se_p_`var'_`i'=r(sd_2)/sqrt(r(N_2))
                local diff_`var'_`i'=r(mu_1)-r(mu_2)
                local p_`var'_`i'=r(p)
                local n_`var'_`i'=r(N_1)
                noi disp as result "`var' ``i'' ," _newline "Mean `var'_l="
`m_l_`var'_`i'' ", Mean `var'_p=" `m_p_`var'_`i'' ", Diff = " `diff_`var'_`i'' " t="
`t_`var'_`i''
        }
}

bys HCI_group: keep if _n==1
keep HCI_group
expand 5
sort HCI_group
bys HCI_group: gen n=_n
list

gen Mean_l=.
```

```
gen Se_l=.
gen Mean_p=.
gen Se_p=.
gen Difference=.
gen p=.
gen t=.
gen N=.

list

gen Variable=""
local l=0
qui foreach varloop in laws $theme {
        local l=`l'+1
        replace Variable="`varloop'" if n==`l'
}

list
local i=0
qui foreach class of local iclass {
        local i=`i'+1
        qui foreach var in laws $theme {
                replace Mean_l=`m_l_`var'_`i'' if Variable=="`var'" &
HCI_group`class'
                replace Se_l=`se_l_`var'_`i'' if Variable=="`var'" &
HCI_group`class'
                replace Mean_p=`m_p_`var'_`i'' if Variable=="`var'" &
HCI_group`class'
                replace Se_p=`se_p_`var'_`i'' if Variable=="`var'" &
HCI_group`class'
                replace Difference=`diff_`var'_`i'' if Variable=="`var'" &
HCI_group`class'
                replace p=`p_`var'_`i'' if Variable=="`var'" & HCI_group`class'
                replace t=`t_`var'_`i'' if Variable=="`var'" & HCI_group`class'
                replace N=`n_`var'_`i'' if Variable=="`var'" & HCI_group`class'
        }
}
drop if HCI_group==.
}
export excel HCI_group Variable Mean_l Se_l Mean_p Se_p Difference p t N using
"Table C4.xlsx",replace firstrow(variables)

restore

preserve
{
gen diff=practices-laws

local iclass `" "==1" "==2" "'
local iclasswords `""Below median HCI" "Above median HCI" "'
```

```
tokenize `"`iclasswords'"'

global theme "Transparency Competition Limits_to_Exclusion Integrity_of_contract"

ttest diff ,by(HCI_group)
local t_diff=r(t)
local p_diff=r(p_l)

qui foreach var in $theme {
        gen diff_`var'=`var'_p-`var'_l
        ttest diff_`var' ,by(HCI_group)
        local t_`var'=r(t)
        local p_`var'=r(p_l)
}

keep in 1
set obs 5
gen p=.
gen t=.
gen n=_n
keep n p t
list

gen Variable=""
local l=0
qui foreach varloop in diff $theme {
        local l=`l'+1
        replace Variable="`varloop'" if n==`l'
}

qui foreach var in diff $theme {
        replace p=`p_`var'' if Variable=="`var'"
        replace t=`t_`var'' if Variable=="`var'"
}
}
export excel Variable p t using "Table C4.xlsx", firstrow(variables) sheet("by HCI")

restore

*C5
foreach var in Time Overrun Low_quality  {
        replace `var' =-1* `var'
}
est drop _all

qui sum HCI
gen HCI_demeaned=HCI-r(mean)

qui sum laws
gen laws_demeaned=laws-r(mean)
```

```
gen HCIlaws=HCI_demeaned*laws_demeaned

reg Time laws HCI HCIlaws
est store c1
reg Overrun laws HCI HCIlaws
est store c2
reg Low_quality laws HCI HCIlaws
est store c3

reg Corruption  laws HCI HCIlaws
est store c4
reg Favoritism laws HCI HCIlaws
est store c5
reg Collusion laws HCI HCIlaws
est store c6
reg No_competition laws HCI HCIlaws
est store c7

esttab _all using "Table C5.csv", se r2 replace nogap nocons nonumbers  unstack
compress  star(* 0.1 ** 0.05 *** 0.01)
est drop _all
```