



User: Calendar
Project: Tables

```

name: <unnamed>
log: P:\Zofar\Schulze\esra23_calendar\out\nls_responseTime.smcl
log type: smcl
opened on: 16 Jul 2023, 10:14:56

```

```

1 .
2 .
3 . keep if calendarRange==2
   (524,858 observations deleted)

4 . keep if epiFinish==1
   (26,230 observations deleted)

5 .
6 .
7 . * _____ Boxplot Bearbeitungsdauer nach Modul_____
8 .
9 . ** Labels für Module vorbereiten
10 .
11 . cap drop wavestrng

12 . gen wavestrng=""
    (1,182,772 missing values generated)

13 . replace wavestrng="03" if wave==201803
    variable wavestrng was str1 now str2
    (637,953 real changes made)

14 . replace wavestrng="04" if wave==201804
    (544,819 real changes made)

15 .
16 . qui levelsof(wave)

17 . local wavelist `r(levels)'

18 .
19 . qui levelsof(modul)

20 . local levelslist `r(levels)'

21 .
22 . levelsof(wavestrng)
    `03' `04'

23 . local wavelb `r(levels)'

24 .
25 .
26 . cap drop modul_labeled

27 . gen modul_labeled = ""
    (1,182,772 missing values generated)

28 .
29 . foreach modul_name in ``levelslist'' {
      2. foreach wave in ``wavelb'' {
          3.     qui levelsof pid if wavestrng=="`wave'" & modul=="`modul_name'" // Anzahl Befragte im Modul
          4.     local part_count : word count `r(levels)'
          5.

```

```

30 .      qui levelsof page if wavestrng=="`wave'" & modul=="`modul_name'" // Anzahl pages pro Modul (aus den hi
    6.      local page_count : word count `r(levels)'
    7.
31 .      replace modul_labeled=="`wave', `modul_name' (n=`part_count', pages: `page_count')"' if wavestrng=="`wave
> "" /* & epiFinish==1 & calendarRange==2*/
    8.      di "`modul_name' `page_count' `part_count'"
    9.  }
   10. }
variable modul_labeled was str1 now str36
(338,915 real changes made)
conventional 89 6066
(257,490 real changes made)
conventional 69 4999
(299,038 real changes made)
episodes 690 6066
(287,329 real changes made)
episodes 196 4999

32 .
33 .
34 .
35 . * _____ Box Plot: Bearbeitungszeit nach Modul (umfangreiche Labels) _____
36 . /// over: Modul und View (Seiten pro Modul+Personen in Labels)
> /// eingeschränkt: nur 2 Jahre, Episodenmodul beendet
>
37 . table (wave modul) /*if epiFinish==1 & calendarRange==2 */ , stat(freq) ///
>      stat(mean moduldauer_minutes) ///
>      stat(median moduldauer_minutes) ///
>      stat(min moduldauer_minutes) ///
>      stat(max moduldauer_minutes) ///
>      nformat(%9.2f) nototals

```

	Frequency	Mean	Median	Minimum value	Maximum value
calendar type					
static calendar					
Fragenmodul					
conventional	338915.00	27.03	19.41	1.53	298.49
episodes	299038.00	21.42	15.92	0.23	306.88
dynamic calendar					
Fragenmodul					
conventional	257490.00	23.70	15.28	1.97	339.54
episodes	287329.00	23.48	17.44	0.69	344.91

```

38 .
39 . set scheme s1color

40 . graph hbox moduldauer_minutes /*if epiFinish==1 & calendarRange==2 */, ///
>      over(modul_labeled, label(labsize(small))) ///
>      nooutsides ///
>      box(1, fcolor("10 125 148") ///
>      fintensity(inten100) ///
>      lcolor(black) ///
>      lwidth(vthin) ///
>      ytitle("Response Time (Minutes)", size(small)) ///
>      ylabel( , labsize(vsmall))

41 .

```

```

42 . graph save Graph "${out}ResponseTime_nachModul_extLabel.gph", replace
    file P:\Zofar\Schulze\esra23_calendar\out\ResponseTime_nachModul_extLabel.gph saved

43 . graph export "${out}ResponseTime_nachModul_extLabel.svg", as(svg) replace
    file P:\Zofar\Schulze\esra23_calendar\out\ResponseTime_nachModul_extLabel.svg saved as SVG format

44 .
45 .
46 .
47 . * _____ Datensatz reduzieren _____
48 . // zwei Zeilen pro Wellenteilnehmer (person + modul)
49 . keep moduldauer_minutes modul_st token wave pid modul mobile_view

50 . collapse (first) moduldauer_minutes modul_st token mobile_view, by(pid modul wave)

51 .
52 .
53 . * _____ Box Plot: Bearbeitungszeit nach Modul & Bildschirmgröße _____
54 . /// over: Modul und View
    > /// eingeschränkt: nur 2 Jahre, Episodenmodul beendet, keine SessionTimeout im jeweiligen Modul
    > table (wave modul mobile_view) if /*epiFinish==1 & calendarRange==2 & */ modul_st==0, stat(freq) ///
    >     stat(mean moduldauer_minutes) ///
    >     stat(median moduldauer_minutes) ///
    >     stat(min moduldauer_minutes) ///
    >     stat(max moduldauer_minutes) ///
    >     nformat(%9.2f) nototals

```

	Frequency	Mean	Median	Minimum value	Maximum value
calendar type					
static calendar					
Fragenmodul					
conventional					
(first) mobile_view					
0	2449.00	21.18	18.11	1.76	246.01
1	2605.00	16.09	14.33	1.53	152.59
episodes					
(first) mobile_view					
0	2601.00	15.32	13.51	1.09	90.06
1	2379.00	13.45	11.47	1.21	91.53
dynamic calendar					
Fragenmodul					
conventional					
(first) mobile_view					
0	1941.00	16.85	14.17	3.47	162.95
1	1949.00	15.44	12.85	2.53	90.24
episodes					
(first) mobile_view					
0	2277.00	15.37	13.13	0.90	108.41
1	2310.00	13.85	12.00	0.69	71.38

```

55 .
56 . quiet: tabout modul wave mobile_view using "${out}nls_respTimeModule.xls" if modul_st==0, ///
    >     c(count wave mean moduldauer_minutes ) ///
    >     clab(N repsTimePerModule ) ///
    >     replace sum ///
    >     f(0 4 0) ///
    >     style(xlsx) ///
    >     font(bold) dpcomma

```

```

57 .
58 .
59 .
60 . set scheme s1color

61 . graph hbox moduldauer_minutes if /*epiFinish==1 & calendarRange==2 & */ modul_st==0, ///
>     over(modul, label(labsize(small))) ///
>     over(mobile_view, label(labsize(vsmall))) ///
>     /// over(wave, label(labsize(small))) ///
>     by(wave) ///
>     subtitle(, size(small) color("black") nobox fcolor()) ///
>     nooutsides ///
>     box(1, fcolor("10 125 148")) ///
>     fintensity(inten100) ///
>     lcolor(black) ///
>     lwidth(vthin) ///
>     ytitle("Response Time (Minutes)", size(small)) ///
>     ylabel( , labsize(vsmall))

62 .
63 .
64 . graph save Graph "${out}ResponseTime_nachModul.gph", replace
    file P:\Zofar\Schulze\esra23_calendar\out\ResponseTime_nachModul.gph saved

65 . graph export "${out}ResponseTime_nachModul.svg", as(svg) replace
    file P:\Zofar\Schulze\esra23_calendar\out\ResponseTime_nachModul.svg saved as SVG format

66 .
67 .
68 .
69 . * _____ Box Plot: Bearbeitungszeit nach Modul & Bildschirmgröße _____
70 . /// over: View
> /// eingeschränkt: nur 2 Jahre, nur Episodenmodul,
> ///                               Episodenmodul beendet, keine SessionTimeout im Modul
> table (wave mobile_view) if /*epiFinish==1 & calendarRange==2 & */ modul=="episodes" & modul_st==0, stat(freq) /
>     stat(mean moduldauer_minutes) ///
>     stat(median moduldauer_minutes) ///
>     stat(min moduldauer_minutes) ///
>     stat(max moduldauer_minutes) ///
>     nformat(%9.2f) nototals

```

	Frequency	Mean	Median	Minimum value	Maximum value
calendar type					
static calendar					
(first) mobile_view					
0	2601.00	15.32	13.51	1.09	90.06
1	2379.00	13.45	11.47	1.21	91.53
dynamic calendar					
(first) mobile_view					
0	2277.00	15.37	13.13	0.90	108.41
1	2310.00	13.85	12.00	0.69	71.38

```

71 .
72 .
73 . graph hbox moduldauer_minutes if /*epiFinish==1 & calendarRange==2 & */ modul=="episodes" & modul_st==0, ///
>     /// over(modul, label(labsize(small))) ///
>     over(mobile_view, label(labsize(vsmall))) ///
>     /// over(wave, label(labsize(small))) ///
>     by(wave) ///
>     subtitle(, size(small) color("black") nobox fcolor()) ///
>     nooutsides ///
>     box(1, fcolor("10 125 148")) ///
>     fintensity(inten100) ///
>     lcolor(black) ///
>     lwidth(vthin) ///
>     ytitle("Response Time (Minutes)", size(small)) ///
>     ylabel( , labsize(vsmall))

```

```
74 .
75 . graph save Graph "${out}ResponseTime_Episodes_nachView.gph", replace
    file P:\Zofar\Schulze\esra23_calendar\out\ResponseTime_Episodes_nachView.gph saved
76 . graph export "${out}ResponseTime_Episodes_nachView.svg", as(svg) replace
    (file P:\Zofar\Schulze\esra23_calendar\out\ResponseTime_Episodes_nachView.svg not found)
    file P:\Zofar\Schulze\esra23_calendar\out\ResponseTime_Episodes_nachView.svg saved as SVG format
77 .
78 .
79 . cap log close
```