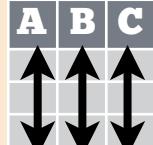


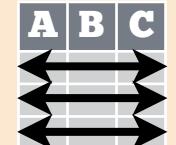
# dplyr yordamida Ma'lumotlar Transformatsiyasi bo'yicha qo'llanma



**dplyr** funksiyalari trubka(pipe)lar bilan ishlaydi va toza ma'lumotni qabul qiladi.

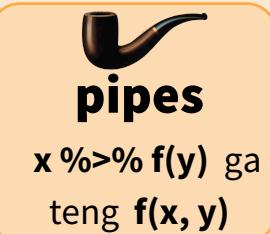


Har bir o'zgaruvchi tegishli ustunda



&

Har bir **kuzatuv**, yoki **holat**, tegishli satrda



## Jamlash

Quyidagilar **jamlash funksiyalarini** ustunlarga qo'llab, yangi jadval yaratadi. Jamlash funksiyalari vektor qabul qiladi va bir qiymatni qaytaradi. (orqa sahfaga qarang).



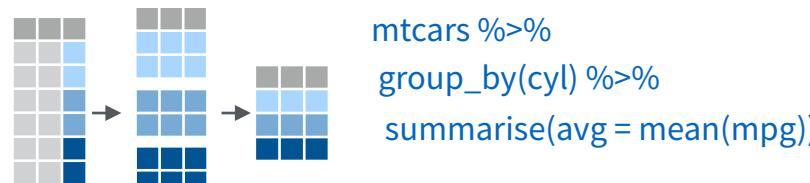
**count(x, ..., wt = NULL, sort = FALSE)**  
Har bir guruhdagi satrlar sonini hisoblash.  
Shuningdek, **tally()**.  
`count(iris, Species)`

### Turli shakllari

- **summarise\_all()** - funksiyani har bir ustunga qo'llash.
- **summarise\_at()** - funksiyani ma'lum bir ustunga qo'llash
- **summarise\_if()** - funksiyani bir toifadagi ustunlarga qo'llash

## Guruhash

**group\_by()** jadvalning guruhlangan nusxasini yaratishda qo'llaniladi. dplyr funksiyalari har bir guruh ustida alohida ishlab, so'ng natijalarni birlashtiradi.



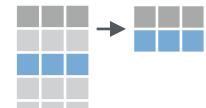
**group\_by(.data, ..., add = FALSE)**  
... ga ko'rta guruhlangan jadval nusxasini qaytaradi.  
`g_iris <- group_by(iris, Species)`

**ungroup(x, ...)**  
Jadvalning guruhlanmagan nusxasini qaytaradi.  
`ungroup(g_iris)`

## Manipulyatsiyalash

### Ajratish

Satr funksiyalari jadval ko'rinishidagi satrlar to'plamini qaytaradi. Nostandard kod uchun \_ bilan yakunlanuvchi variantidan foydalaning.



**filter(.data, ...)**

Shartni qanoatlantiruvchi satrlarni ajratish.  
Shuningdek, **filter\_()**.  
`filter(iris, Sepal.Length > 7)`



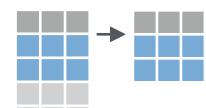
**distinct(.data, ..., .keep\_all = FALSE)**

Bir xil qiymatli satrlarni o'chirish.  
Shuningdek, **distinct\_()**.  
`distinct(iris, Species)`



**sample\_frac(tbl, size = 1, replace = FALSE, weight = NULL, .env = parent.frame())**

Berilgan nisbatda satrlarni tasodify ajratib olish.  
`sample_frac(iris, 0.5, replace = TRUE)`



**slice(.data, ...)**

Joylashuviga ko'rta satrni ajratib olish.  
Shuningdek, **slice\_()**. `slice(iris, 10:15)`

**top\_n(x, n, wt)**

Top n ta yozuvni tanlash va tartiblash(agar guruhlangan bo'lsa, guruh bo'yicha).  
`top_n(iris, 5, Sepal.Width)`

**filter() bilan ishlatishtum mumkin bo'lgan mantiqiy operatorlar**

<	<=	is.na()	%in%		xor()
>	>=	!is.na()	!	&	

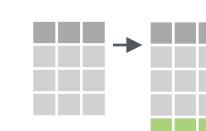
Yordam uchun **?base::logic** va **?Comparison** ga qarang.

### Tartibga keltirish

**arrange(.data, ...)**

Ustundagi qiymatlarga ko'rta satrlarni tartiblash (o'sish tartibida), kamayish tartibida esa **desc()** yordamida tartiblash mumkin.

`arrange(mtcars, mpg)`  
`arrange(mtcars, desc(mpg))`



### Qo'shish

**add\_row(.data, ..., .before = NULL, .after = NULL)**

Jadvalga bir yoki bir nechta satrni qo'shish.  
`add_row(faithful, eruptions = 1, waiting = 1)`

## O'zgaruvchilar manipulyatsiyasi

### O'zgaruvchilarni ajratish

Ustun funksiyalari ustunlar to'plamini yangi jadval shaklida taqdim qiladi. Nostandard kod uchun \_ bilan tugaydiganlardan foydalaning.



**select(.data, ...)**

Nomiga ko'rta ustunni ajratib olish.  
Shuningdek, **select\_if()**.  
`select(iris, Sepal.Length, Species)`

**select() bilan quyidagi yordamchi funksiyalardan foydalaning,**

Masalan: `select(iris, starts_with("Sepal"))`

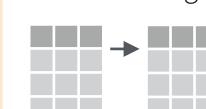
**contains(match)**  
**ends\_with(match)**  
**matches(match)**

**num\_range(prefix, range)**  
**one\_of(...)**  
**starts\_with(match)**

: e.g. mpg:cyl  
-, e.g., -Species  
**everything()**

### Yangi o'zgaruvchilarni yaratish

Quyidagilar, **vektorli funksiyalarini** ustunlarga tatbiq qiladi. Vektorli funksiyalar vektorni qabul qiladi va bir xil o'lchamdagи vektorni qaytaradi.



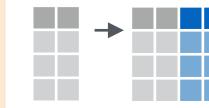
**mutate(.data, ...)**

Yangi ustun(lar)ni hisoblash.  
`mutate(mtcars, gpm = 1/mpg)`



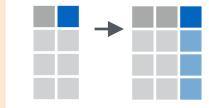
**transmute(.data, ...)**

Yangi ustun(lar)ni hisoblash, qolganlarini tashlab yuborish.  
`transmute(mtcars, gpm = 1/mpg)`



**mutate\_all(.tbl, .funs, ...)**

Funksiyalarni har bir ustunga tatbiq qilish.  
**funs()** bilan ishlatishtum lozim.  
`mutate_all(faithful, funs(log(), log2()))`



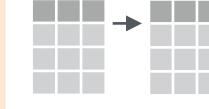
**mutate\_at(.tbl, .cols, .funs, ...)**

Funksiyalarni faqat ayrim ustunlarga tatbiq qilish. **funs()**, **vars()** va **select()** uchun yordamchi funksiyalardan foydalanshtum lozim.  
`mutate_at(iris, vars(-Species), funs(log(.)))`



**mutate\_if(.tbl, .predicate, .funs, ...)**

Funksiyalarni bir xil toifadagi ustunlarga qo'llash. **funs()** bilan qo'llang.  
`mutate_if(iris, is.numeric, funs(log(.)))`



**add\_column(.data, ..., .before = NULL, .after = NULL)**

Yangi ustun(lar)ni qo'shish.  
`add_column(mtcars, new = 1:32)`



**rename(.data, ...)**

Ustunlarni qayta nomlash.  
`rename(iris, Length = Sepal.Length)`

## Vektorli funksiyalar

### mutate() bilan qo'llash uchun

**mutate()** va **transmute()** vektorli funksiyalarni ustunlarga qo'llab, yangi ustun hosil qiladi. Vektorli funksiyalar vektorlarni qabul qilib, shu o'lchamli vektorni qaytaradi.



#### Siljitchish

dplyr::lag() - Elementlarni 1 ta oldinga siljitchish.

dplyr::lead() - Elementlarni 1 ta orqaga siljitchish.

#### Yig'uvchi funksiyalar

dplyr::cumall() - yig'uvchi all()

dplyr::cumany() - yig'uvchi any()

cummax() - yig'uvchi max()

dplyr::cummean() - yig'uvchi mean()

cummin() - yig'uvchi min()

cumprod() - yig'uvchi prod()

cumsum() - yig'uvchi sum()

#### Reytinglash

dplyr::cume\_dist() - barcha qiymatlar nisbati <=

dplyr::dense\_rank() - ties = min ga ko'ra reytinglash, bo'sh joylarsiz.

dplyr::min\_rank() - ties = min ga ko'ra reytinglash

dplyr::ntile() - bins into n bins

dplyr::percent\_rank() - [0,1] oraliq'iga keltirilgan min\_rank

dplyr::row\_number() - ties = "first" ga ko'ra reytinglash

#### Matematika

+, -, \*, /, ^, %/, %% - arifmetik amallar

log(), log2(), log10() - logarifmlar

<, <=, >, >=, !=, == - mantiqiy taqqoslash

#### Boshqalar

dplyr::between() - x >= left & x <= right

dplyr::case\_when() - ko'p holatlari if\_else()

dplyr::coalesce() - vektorlar to'plami

elementlarining, birinchi mavjud qiymatlari

dplyr::if\_else() - elementma-element if() + else()

dplyr::na\_if() - ma'lum qiymatlarni NAga almashtirish

pmax() - elementma-element max()

pmin() - elementma-element min()

dplyr::recode() - vektorli switch()

dplyr::recode\_factor() - faktorlar uchun vektorli switch()

## Jamlash funksiyalari

### summarise() bilan qo'llash uchun

**summarise()** jamlash funksiyalarini ustunlarga qo'llab yangi ustun hosil qiladi. Jamlash funksiyalar, vektorlarni qabul qilib, yagona qiymat qaytaradi.



#### Soni

dplyr::n() - qiymatlar/satrlar soni

dplyr::n\_distinct() - unikal satrlar soni

sum(!is.na()) - qiymati mavjud satrlar soni

#### Joylashuvi

mean() - o'rtacha qiymat, mean(!is.na())

median() - mediana

#### Mantiqiy qiymatlar

mean() - ROST qiymatlar nisbati

sum() - ROST (TRUE) qiymatlari soni

#### Joylashuv/tartib

dplyr::first() - birinchi qiymat

dplyr::last() - oxirgi qiymat

dplyr::nth() - vektorning n-qiyamti

#### Rang

quantile() - n-kvartil

min() - minimum qiymat

max() - maksimum qiymat

#### Yoyilishi

IQR() - Inter-kvartil oraliq'i

mad() - chetlanishning o'rtacha absolyut qiymati

sd() - o'rtacha kvadratik chetlanish

var() - dispersiya

## Satr nomlari

Toza ma'lumotlar ustunlardan alohida saqlanuvchi satr nomlaridan foydalanmaydi. Ular bilan ishlash uchun avval ustunga ko'chirish lozim.

A	B	C
1	a t	1 a t
2	b u	2 b u
3	c v	3 c v

#### rownames\_to\_column()

Satr nomlarini ustunga ko'chirish.

a <- rownames\_to\_column(iris,

var = "C")

A	B	C
1	a t	1 a t
2	b u	2 b u
3	c v	3 c v

#### column\_to\_rownames()

Ustunni satr nomlariga ko'chirish.

column\_to\_rownames(a,

var = "C")

Hamda has\_rownames(), remove\_rownames()

## Jadvallarni birlashtirish

### O'zgaruvchilarni birlashtirish

x	A	B	C
	a	t	1
	b	u	2
	c	v	3

y	A	B	D
	a	t	3
	b	u	2
	d	w	1

=
---

**bind\_cols()** dan jadvallarni qanday bo'lsa shundayligicha bir-biriga yonma-yon qo'yishda foydalaning.

A	B	C	A	B	D
a	t	1	a	t	3
b	u	2	b	u	2
c	v	3	d	w	1

#### bind\_cols(...)

Jadvallarni yonma-yon joylashtirib, yagona jadval shaklida qaytaradi. SATRLAR MOS KELISHI LOZIM.

"O'zgartirib birlashtirish" funksiyalaridan bir jadvalga boshqasining ustunlaridan foydalananib mos satrlarga birlashtirishda foydalanish mumkin. Har bir funksiya jadvaldan turli qiymatlar jadvalini saqlab qoladi.

A	B	C	D
a	t	1	3
b	u	2	2
c	v	3	NA

**left\_join(x, y, by = NULL, copy=FALSE, suffix=c(".x",".y"),...)**  
Mos qiymatlarni y dan x ga joylash.

A	B	C	D
a	t	1	3
b	u	2	2
d	w	NA	1

**right\_join(x, y, by = NULL, copy = FALSE, suffix=c(".x",".y"),...)**  
Mos qiymatlarni x dan y ga joylash.

A	B	C	D
a	t	1	3
b	u	2	2

**inner\_join(x, y, by = NULL, copy = FALSE, suffix=c(".x",".y"),...)**  
Ma'lumotlarni birlashtirish. Faqat mos keluvchi satrlar saqlanib qolinadi.

A	B	C	D
a	t	1	3
b	u	2	2
c	v	3	NA

**full\_join(x, y, by = NULL, copy=FALSE, suffix=c(".x",".y"),...)**  
Ma'lumotlarni birlashtirish. Barcha qiymatlari va satrlar saqlanib qolinadi.

A.x	B.x	C	A.y	B.y	D
a	t	1	d	w	
b	u	2	b	u	
c	v	3	a		