R Notebook

Code ▾

Hide

```
head(iris)
```

	Sepal.Length <dbl>	Sepal.Width <dbl>	Petal.Length <dbl>	Petal.Width <dbl>	Species <fctr>
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa

6 rows

Hide

```
dim(iris)
```

```
[1] 150   5
```

Hide

```
str(iris)
```

```
'data.frame':	150 obs. of  5 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

MANIPULATE ROWS

EXTRACT ROWS

Hide

```
library(dplyr)
```

Hide

```
#Extract rows that meet logical criteria
filter(iris,is.na(Petal.Length))
```

```
0 rows
```

Hide

```
# <,>,<=,>=,is.na(),!is.na(),!,%in%,&,|,xor() these operators can be used with filter() functio
n.
```

Hide

```
#Remove rows with duplicate values. Display only unique values of a column
distinct(iris,Sepal.Length)
```

Sepal.Length <dbl>
5.1
4.9
4.7
4.6
5.0
5.4
4.4
4.8
4.3
5.8

1-10 of 35 rows Previous **1** 2 3 4 Next

Hide

```
#Randomly select fraction of rows.
sample_frac(iris,0.5,replace=TRUE)
```

	Sepal.Length <dbl>	Sepal.Width <dbl>	Petal.Length <dbl>	Petal.Width <dbl>	Species <fctr>
16	5.7	4.4	1.5	0.4	setosa
107	4.9	2.5	4.5	1.7	virginica
138	6.4	3.1	5.5	1.8	virginica
95	5.6	2.7	4.2	1.3	versicolor

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
	<dbl>	<dbl>	<dbl>	<dbl>	<fctr>
87	6.7	3.1	4.7	1.5	versicolor
94	5.0	2.3	3.3	1.0	versicolor
94.1	5.0	2.3	3.3	1.0	versicolor
125	6.7	3.3	5.7	2.1	virginica
42	4.5	2.3	1.3	0.3	setosa
57	6.3	3.3	4.7	1.6	versicolor

1-10 of 75 rows Previous **1** 2 3 4 5 6 ... 8 Next

Hide

```
#Randomly select size rows
sample_n(iris, 10, replace = TRUE)
```

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
	<dbl>	<dbl>	<dbl>	<dbl>	<fctr>
86	6.0	3.4	4.5	1.6	versicolor
76	6.6	3.0	4.4	1.4	versicolor
96	5.7	3.0	4.2	1.2	versicolor
95	5.6	2.7	4.2	1.3	versicolor
8	5.0	3.4	1.5	0.2	setosa
101	6.3	3.3	6.0	2.5	virginica
57	6.3	3.3	4.7	1.6	versicolor
87	6.7	3.1	4.7	1.5	versicolor
16	5.7	4.4	1.5	0.4	setosa
70	5.6	2.5	3.9	1.1	versicolor

1-10 of 10 rows

Hide

```
#Select rows by position.
slice(iris, 10:15)
```

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
<dbl>	<dbl>	<dbl>	<dbl>	<fctr>
4.9	3.1	1.5	0.1	setosa

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
<dbl>	<dbl>	<dbl>	<dbl>	<fctr>
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3.0	1.4	0.1	setosa
4.3	3.0	1.1	0.1	setosa
5.8	4.0	1.2	0.2	setosa

6 rows

Hide

```
#Select and order top n entries (by group if grouped data).
top_n(iris, 5, Sepal.Width)
```

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
<dbl>	<dbl>	<dbl>	<dbl>	<fctr>
5.4	3.9	1.7	0.4	setosa
5.8	4.0	1.2	0.2	setosa
5.7	4.4	1.5	0.4	setosa
5.4	3.9	1.3	0.4	setosa
5.2	4.1	1.5	0.1	setosa
5.5	4.2	1.4	0.2	setosa

6 rows

ARRANGE ROWS

Hide

```
#Order rows by values of a column or columns (low to high), use with desc() to order from high t
o low.
arrange(mtcars, mpg)
```

mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear ▶
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
10.4	8	472.0	205	2.93	5.250	17.98	0	0	3
10.4	8	460.0	215	3.00	5.424	17.82	0	0	3
13.3	8	350.0	245	3.73	3.840	15.41	0	0	3
14.3	8	360.0	245	3.21	3.570	15.84	0	0	3
14.7	8	440.0	230	3.23	5.345	17.42	0	0	3

mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear ▶
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
15.0	8	301.0	335	3.54	3.570	14.60	0	1	5
15.2	8	275.8	180	3.07	3.780	18.00	0	0	3
15.2	8	304.0	150	3.15	3.435	17.30	0	0	3
15.5	8	318.0	150	2.76	3.520	16.87	0	0	3
15.8	8	351.0	264	4.22	3.170	14.50	0	1	5

1-10 of 32 rows | 1-10 of 11 columns Previous **1** 2 3 4 Next

Hide

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

mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear ▶
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
33.9	4	71.1	65	4.22	1.835	19.90	1	1	4
32.4	4	78.7	66	4.08	2.200	19.47	1	1	4
30.4	4	75.7	52	4.93	1.615	18.52	1	1	4
30.4	4	95.1	113	3.77	1.513	16.90	1	1	5
27.3	4	79.0	66	4.08	1.935	18.90	1	1	4
26.0	4	120.3	91	4.43	2.140	16.70	0	1	5
24.4	4	146.7	62	3.69	3.190	20.00	1	0	4
22.8	4	108.0	93	3.85	2.320	18.61	1	1	4
22.8	4	140.8	95	3.92	3.150	22.90	1	0	4
21.5	4	120.1	97	3.70	2.465	20.01	1	0	3

1-10 of 32 rows | 1-10 of 11 columns Previous **1** 2 3 4 Next

ADD ROWS

Hide

```
#Add one or more rows to a table.
add_row(mtcars, hp = 200, cyl = 5)
```

mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear ▶
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
21.0	6	160.0	110	3.90	2.620	16.46	0	1	4
21.0	6	160.0	110	3.90	2.875	17.02	0	1	4

mpg <dbl>	cyl <dbl>	disp <dbl>	hp <dbl>	drat <dbl>	wt <dbl>	qsec <dbl>	vs <dbl>	am <dbl>	gear <dbl>
22.8	4	108.0	93	3.85	2.320	18.61	1	1	4
21.4	6	258.0	110	3.08	3.215	19.44	1	0	3
18.7	8	360.0	175	3.15	3.440	17.02	0	0	3
18.1	6	225.0	105	2.76	3.460	20.22	1	0	3
14.3	8	360.0	245	3.21	3.570	15.84	0	0	3
24.4	4	146.7	62	3.69	3.190	20.00	1	0	4
22.8	4	140.8	95	3.92	3.150	22.90	1	0	4
19.2	6	167.6	123	3.92	3.440	18.30	1	0	4

1-10 of 33 rows | 1-10 of 11 columns Previous **1** 2 3 4 Next

MANIPULATE COLUMNS/VARIABLES

EXTRACT VARIABLES

Hide

```
#Extract column values as a vector. Choose by name or index.
pull(iris, Sepal.Length)
```

```
  [1] 5.1 4.9 4.7 4.6 5.0 5.4 4.6 5.0 4.4 4.9 5.4 4.8 4.8 4.3 5.8 5.7 5.4 5.1 5.7 5.1
 [21] 5.4 5.1 4.6 5.1 4.8 5.0 5.0 5.2 5.2 4.7 4.8 5.4 5.2 5.5 4.9 5.0 5.5 4.9 4.4 5.1
 [41] 5.0 4.5 4.4 5.0 5.1 4.8 5.1 4.6 5.3 5.0 7.0 6.4 6.9 5.5 6.5 5.7 6.3 4.9 6.6 5.2
 [61] 5.0 5.9 6.0 6.1 5.6 6.7 5.6 5.8 6.2 5.6 5.9 6.1 6.3 6.1 6.4 6.6 6.8 6.7 6.0 5.7
 [81] 5.5 5.5 5.8 6.0 5.4 6.0 6.7 6.3 5.6 5.5 5.5 6.1 5.8 5.0 5.6 5.7 5.7 6.2 5.1 5.7
[101] 6.3 5.8 7.1 6.3 6.5 7.6 4.9 7.3 6.7 7.2 6.5 6.4 6.8 5.7 5.8 6.4 6.5 7.7 7.7 6.0
[121] 6.9 5.6 7.7 6.3 6.7 7.2 6.2 6.1 6.4 7.2 7.4 7.9 6.4 6.3 6.1 7.7 6.3 6.4 6.0 6.9
[141] 6.7 6.9 5.8 6.8 6.7 6.7 6.3 6.5 6.2 5.9
```

Hide

```
#Extract columns as a table. Also select_if().
select(iris, Sepal.Length, Species)
```

Sepal.Length <dbl>	Species <fctr>
5.1	setosa
4.9	setosa
4.7	setosa
4.6	setosa
5.0	setosa

Sepal.Length	Species
<dbl>	<fctr>
5.4	setosa
4.6	setosa
5.0	setosa
4.4	setosa
4.9	setosa

1-10 of 150 rows Previous **1** 2 3 4 5 6 ... 15 Next

We can use these helpers with select ():- e.g. select(iris, starts_with("Sepal"))

contains(match) ends_with(match) matches(match) :, e.g. mpg:cyl -, e.g, -Species num_range(prefix, range) one_of(.) starts_with(match)

MAKE NEW VARIABLES

Hide

```
#Compute new column(s).
mutate(mtcars, gpm = 1/mpg)
```

mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
21.0	6	160.0	110	3.90	2.620	16.46	0	1	4
21.0	6	160.0	110	3.90	2.875	17.02	0	1	4
22.8	4	108.0	93	3.85	2.320	18.61	1	1	4
21.4	6	258.0	110	3.08	3.215	19.44	1	0	3
18.7	8	360.0	175	3.15	3.440	17.02	0	0	3
18.1	6	225.0	105	2.76	3.460	20.22	1	0	3
14.3	8	360.0	245	3.21	3.570	15.84	0	0	3
24.4	4	146.7	62	3.69	3.190	20.00	1	0	4
22.8	4	140.8	95	3.92	3.150	22.90	1	0	4
19.2	6	167.6	123	3.92	3.440	18.30	1	0	4

1-10 of 32 rows | 1-10 of 12 columns Previous **1** 2 3 4 Next

Hide

```
#Compute new column(s), drop others.
transmute(mtcars, gpm = 1/mpg)
```

gpm
<dbl>

gpm
<dbl>
0.04761905
0.04761905
0.04385965
0.04672897
0.05347594
0.05524862
0.06993007
0.04098361
0.04385965
0.05208333

1-10 of 32 rows Previous **1** 2 3 4 Next

Hide

```
#Apply funs to every column. Use with funs(). Also mutate_if().
mutate_all(mtcars, funs(log(.), log2(.)))
```

mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear ▸
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
21.0	6	160.0	110	3.90	2.620	16.46	0	1	4
21.0	6	160.0	110	3.90	2.875	17.02	0	1	4
22.8	4	108.0	93	3.85	2.320	18.61	1	1	4
21.4	6	258.0	110	3.08	3.215	19.44	1	0	3
18.7	8	360.0	175	3.15	3.440	17.02	0	0	3
18.1	6	225.0	105	2.76	3.460	20.22	1	0	3
14.3	8	360.0	245	3.21	3.570	15.84	0	0	3
24.4	4	146.7	62	3.69	3.190	20.00	1	0	4
22.8	4	140.8	95	3.92	3.150	22.90	1	0	4
19.2	6	167.6	123	3.92	3.440	18.30	1	0	4

1-10 of 32 rows | 1-10 of 33 columns Previous **1** 2 3 4 Next

Hide

```
mutate_if(iris, is.numeric, funs(log(.)))
```

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
<dbl>	<dbl>	<dbl>	<dbl>	<fctr>
1.629241	1.2527630	0.33647224	-1.60943791	setosa
1.589235	1.0986123	0.33647224	-1.60943791	setosa
1.547563	1.1631508	0.26236426	-1.60943791	setosa
1.526056	1.1314021	0.40546511	-1.60943791	setosa
1.609438	1.2809338	0.33647224	-1.60943791	setosa
1.686399	1.3609766	0.53062825	-0.91629073	setosa
1.526056	1.2237754	0.33647224	-1.20397280	setosa
1.609438	1.2237754	0.40546511	-1.60943791	setosa
1.481605	1.0647107	0.33647224	-1.60943791	setosa
1.589235	1.1314021	0.40546511	-2.30258509	setosa

1-10 of 150 rows Previous **1** 2 3 4 5 6 ... 15 Next

Hide

```
#Apply funs to specific columns. Use with funs(), vars() and the helper functions for select().
mutate_at(iris, vars( -Species), funs(log(.)))
```

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
<dbl>	<dbl>	<dbl>	<dbl>	<fctr>
1.629241	1.2527630	0.33647224	-1.60943791	setosa
1.589235	1.0986123	0.33647224	-1.60943791	setosa
1.547563	1.1631508	0.26236426	-1.60943791	setosa
1.526056	1.1314021	0.40546511	-1.60943791	setosa
1.609438	1.2809338	0.33647224	-1.60943791	setosa
1.686399	1.3609766	0.53062825	-0.91629073	setosa
1.526056	1.2237754	0.33647224	-1.20397280	setosa
1.609438	1.2237754	0.40546511	-1.60943791	setosa
1.481605	1.0647107	0.33647224	-1.60943791	setosa
1.589235	1.1314021	0.40546511	-2.30258509	setosa

1-10 of 150 rows Previous **1** 2 3 4 5 6 ... 15 Next

Hide

```
#Add new column(s). Also add_count(), add_tally().
add_column(mtcars, new = 1:32)
```

```
Error: could not find function "add_column"
```

Hide

```
#Rename columns.
rename(iris, Length = Sepal.Length)
```

Length	Sepal.Width	Petal.Length	Petal.Width	Species
<dbl>	<dbl>	<dbl>	<dbl>	<fctr>
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa

1-10 of 150 rows Previous **1** 2 3 4 5 6 ... 15 Next

SUMMARISE ROWS

Hide

```
#Compute table of summaries.
summarise(mtcars, avg = mean(mpg))
```

avg
<dbl>
20.09062

1 row

Hide

```
#Count number of rows in each group defined by the variables in … Also tally().
count(iris, Species)
```

Species <fctr>	n <int>
setosa	50
versicolor	50
virginica	50

3 rows

Variations:

summarise_all() - Apply funs to every column. summarise_at() - Apply funs to specific columns. summarise_if() - Apply funs to all cols of one type.

GROUP ROWS

Hide

```
#Returns copy of table grouped by …
g_iris <- group_by(iris, Species)
g_iris
```

Sepal.Length <dbl>	Sepal.Width <dbl>	Petal.Length <dbl>	Petal.Width <dbl>	Species <fctr>
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa

1-10 of 150 rows Previous **1** 2 3 4 5 6 ... 15 Next

Hide

```
#Returns ungrouped copy of table.
ungroup(g_iris)
```

Sepal.Length <dbl>	Sepal.Width <dbl>	Petal.Length <dbl>	Petal.Width <dbl>	Species <fctr>

Sepal.Length <dbl>	Sepal.Width <dbl>	Petal.Length <dbl>	Petal.Width <dbl>	Species <fctr>
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa

1-10 of 150 rows Previous **1** 2 3 4 5 6 ... 15 Next

Hide

```
mtcars %>%
group_by(cyl) %>%
summarise(avg = mean(mpg))
```

cyl <dbl>	avg <dbl>
4	26.66364
6	19.74286
8	15.10000

3 rows

Summary Functions:-

TO USE WITH SUMMARISE ()-

COUNTS

dplyr::n() - number of values/rows dplyr::n_distinct() - # of uniques sum(!is.na()) - # of non-NA’s

LOCATION mean() - mean, also mean(!is.na()) median() - median

LOGICALS mean() - Proportion of TRUE’s sum() - # of TRUE’s

POSITION/ORDER dplyr::first() - first value dplyr::last() - last value dplyr::nth() - value in nth location of vector

RANK quantile() - nth quantile min() - minimum value max() - maximum value

SPREAD IQR() - Inter-Quartile Range mad() - median absolute deviation sd() - standard deviation var() - variance

MUTATE FUNCTIONS :-

TO USE WITH MUTATE ()

OFFSETS

dplyr::lag() - Offset elements by 1 dplyr::lead() - Offset elements by -1

CUMULATIVE AGGREGATES

dplyr::cumall() - Cumulative all() dplyr::cumany() - Cumulative any() cummax() - Cumulative max() dplyr::cummean() - Cumulative mean() cummin() - Cumulative min() cumprod() - Cumulative prod() cumsum() - Cumulative sum()

RANKINGS

dplyr::cume_dist() - Proportion of all values <= dplyr::dense_rank() - rank with ties = min, no gaps dplyr::min_rank() - rank with ties = min dplyr::ntile() - bins into n bins dplyr::percent_rank() - min_rank scaled to [0,1] dplyr::row_number() - rank with ties = “first”

MATH

+, - , *, /, ^, %/%, %% - arithmetic ops log(), log2(), log10() - logs <, <=, >, >=, !=, == - logical comparisons dplyr::between() - x >= left & x <= right dplyr::near() - safe == for floating point numbers

MISC

dplyr::case_when() - multi-case if_else() dplyr::coalesce() - first non-NA values by element across a set of vectors dplyr::if_else() - element-wise if() + else() dplyr::na_if() - replace specific values with NA pmax() - element-wise max() pmin() - element-wise min() dplyr::recode() - Vectorized switch() dplyr::recode_factor() - Vectorized switch() for factors