Becoming a tidy ninja with tidyr

tidyr

A package that ***reshapes*** the layout of tables and **tidies/cleans** existing data

Verb	Usage
gather	collapses multiple columns in key-value pairs
spread	spreads a key-value pair across multiple columns
replace_na	replace missing values
fill	fills missing values by using previous entry
separate	turns a single character column into multiple columns
extract	turns each regex capture group into a new column
unite	paste together multiple columns into one
complete	explicitly completes missing data combinations

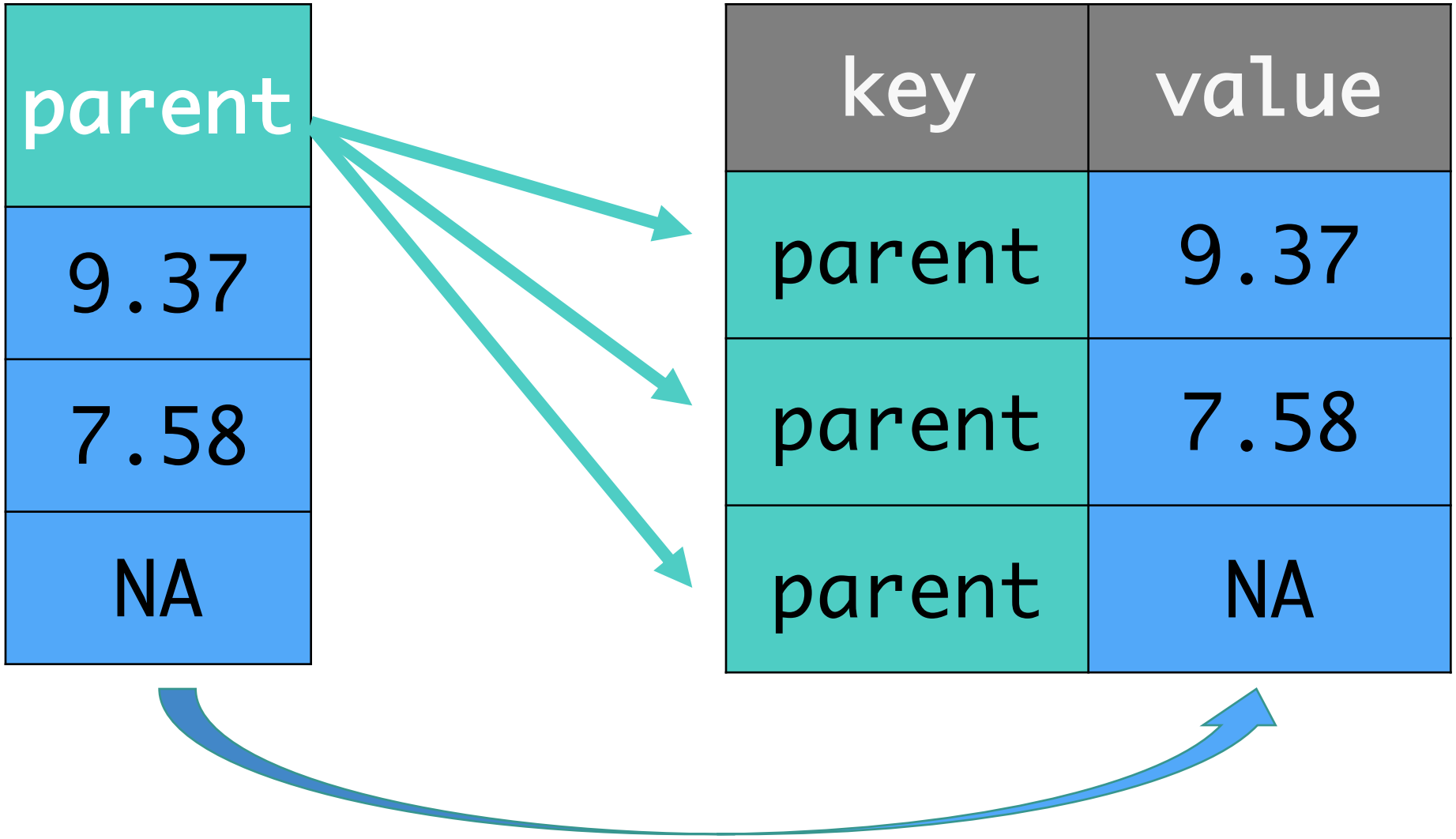
```
df <- data.frame(id = 1,
date = paste0(seq(as.Date("2015/10/20"), by = "day", length.out =
3),"T","9:00:00"),
parent = c(round(sort(10 - rnorm(2,1,1),decreasing = TRUE),2),NA),
met1 = c(round(sort(8 - rnorm(2,1,1),decreasing = TRUE),2),NA),
wt = c(70,rep(NA,2)),
age = c(50,rep(".”,2)))
```

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

Verb	Usage
gather	collapses multiple columns in key-value pairs
spread	spreads a key-value pair across multiple columns
replace_na	replace missing values
fill	fills missing values by using previous entry
separate	turns a single character column into multiple columns
extract	turns each regex capture group into a new column
unite	paste together multiple columns into one
complete	explicitly completes missing data combinations

```
gather(df,
       <key_col_name>,
       <value_col_name>,
       <column(s)_to_pivot>)
```

gather(df, key, value, parent)



id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

```
gather(df,
       analyte
       dv,
       parent, met1)
```

id	date	wt	age	analyte	dv
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA

id	date	wt	age	analyte	dv
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

Stack back together

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

```
gather(df,
       analyte
       dv,
       parent, met1)
```

id	date	wt	age	analyte	DV

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

```
gather(df,
       analyte
       dv,
       parent, met1)
```

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

```
gather(df,
       analyte
       dv,
       parent, met1)
```

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

```
gather(df,
       analyte
       dv,
       parent, met1)
```

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

```
gather(df,
       analyte
       dv,
       parent, met1)
```

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

```
gather(df,
       analyte
       dv,
       parent, met1)
```

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

```
gather(df,
       analyte
       dv,
       parent, met1)
```

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

```
gather(df,
       analyte,
       dv,
       parent, met1)
```

gather()

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

(former column names)

key

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

(former cells)

key values

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

gather()

Collapses multiple columns into two columns:

1. a key column that contains the former column names
2. a value column that contains the former column cell values

```
gather(df, analyte,DV,parent:met1)
```

data frame to reshape

name of the new key column

name of the new value column

name or numeric indices of columns to collapse

Verb	Usage
gather	collapses multiple columns in key-value pairs
spread	spreads a key-value pair across multiple columns
replace_na	replace missing values
fill	fills missing values by using previous entry
separate	turns a single character column into multiple columns
extract	turns each regex capture group into a new column
unite	paste together multiple columns into one
complete	explicitly completes missing data combinations

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

Active Moiety Concentration
= Parent + Met1

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

```
gdf %>%
spread(analyte,DV)
```

id	date	wt	age	parent	met1

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

```
gdf %>%
spread(analyte,DV)
```

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

```
gdf %>%
spread(analyte,DV)
```

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

```
gdf %>%
spread(analyte,DV)
```

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

```
gdf %>%
spread(analyte,DV)
```

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

```
gdf %>%
spread(analyte,DV)
```

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

```
gdf %>%
spread(analyte,DV)
```

id	date	wt	age	parent	met1	AM
1	2015-10-20T9:00:00	70	50	9.37	8.05	
1	2015-10-21T9:00:00	NA	.	7.58	6.41	
1	2015-10-22T9:00:00	NA	.	NA	NA	

mutate

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

```
gdf %>%
spread(analyte,DV)
```

spread()

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

key (new column names)

```
gdf %>%
spread(analyte,DV)
```

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

key **value** (new cells)

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	70	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

```
gdf %>%
spread(analyte,DV)
```

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

spread()

Generates multiple columns from two columns:

1. each unique value in the key column becomes a column name
2. each value in the **value** column becomes a cell in the new columns

```
spread(df, analyte,DV)
```

spread()

Generates multiple columns from two columns:

1. each unique value in the key column becomes a column name
2. each value in the **value** column becomes a cell in the new columns

```
spread(df, analyte,DV)
```

data frame to reshape

column to use for keys

column to use for values

```
gdf <- df %>% gather(analyte,DV,parent:met1)
```

id	date	wt	age	parent	met1
1	2015-10-20T9:00:00	70	50	9.37	8.05
1	2015-10-21T9:00:00	NA	.	7.58	6.41
1	2015-10-22T9:00:00	NA	.	NA	NA

```
gdf %>%
spread(analyte,DV)
```

id	date	wt	age	analyte	DV
1	2015-10-20T9:00:00	70	50	parent	9.37
1	2015-10-21T9:00:00	NA	.	parent	7.58
1	2015-10-22T9:00:00	NA	.	parent	NA
1	2015-10-20T9:00:00	NA	50	met1	8.05
1	2015-10-21T9:00:00	NA	.	met1	6.41
1	2015-10-22T9:00:00	NA	.	met1	NA

Verb	Usage
gather	collapses multiple columns in key-value pairs
spread	spreads a key-value pair across multiple columns
replace_na	replace missing values
fill	fills missing values by using previous entry
separate	turns a single character column into multiple columns
extract	turns each regex capture group into a new column
unite	paste together multiple columns into one
complete	explicitly completes missing data combinations

replace_na()

- replaces missing values with specific value of choice
- replaces many missing values at once to any “type”

```
df %>%
  replace_na(
    list(parent="BQL",
         met1="unknown",AM=-99)
)
```

id	date	wt	age	parent	met1	AM
1	2015-10-20T9:00:00	70	50	9.37	8.05	17.42
1	2015-10-21T9:00:00	NA	.	7.58	6.41	14.26
1	2015-10-22T9:00:00	NA	.	NA	NA	NA

id	date	wt	age	parent	met1	AM
1	2015-10-20T9:00:00	70	50	9.37	8.05	17.42
1	2015-10-21T9:00:00	NA	.	7.58	6.41	14.26
1	2015-10-22T9:00:00	NA	.	NA	NA	NA

BQL	unknown	-99

```
df %>% replace_na(list(
        parent="BQL",
        met1="unknown",
        AM=-99)
    )
```

id	date	wt	age	parent	met1	AM
1	2015-10-20T9:00:00	70	50	9.37	8.05	17.42
1	2015-10-21T9:00:00	NA	.	7.58	6.41	14.26
1	2015-10-22T9:00:00	NA	.	BQL	unknown	-99

```
Replacements <- list(
    parent="BQL",
    met1="unknown",
    AM=-99)
df %>% replace_na(replacements)
```

Verb	Usage
gather	collapses multiple columns in key-value pairs
spread	spreads a key-value pair across multiple columns
replace_na	replace missing values
fill	fills missing values by using previous entry
separate	turns a single character column into multiple columns
extract	turns each regex capture group into a new column
unite	paste together multiple columns into one
complete	explicitly completes missing data combinations

id	date	wt	age	parent	met1	AM
1	2015-10-20T9:00:00	70	50	9.37	8.05	17.42
1	2015-10-21T9:00:00	NA	.	7.58	6.41	14.26
1	2015-10-22T9:00:00	NA	.	BQL	unknown	-99

id	date	wt	age	parent	met1	AM
1	2015-10-20T9:00:00	70	50	9.37	8.05	17.42
1	2015-10-21T9:00:00	70	.	7.58	6.41	14.26
1	2015-10-22T9:00:00	70	.	BQL	unknown	-99

What will be outcome of fill()'ing age

id	date	wt	age	parent	met1	AM
1	2015-10-20T9:00:00	70	50	9.37	8.05	17.42
1	2015-10-21T9:00:00	70	.	7.58	6.41	14.26
1	2015-10-22T9:00:00	70	.	BQL	unknown	-99

tidyr verbs are select() aware

```
df %>% fill(everything())
df %>% fill(contains(“OCC”))
df %>% fill(-USUBJID)
```

Verb	Usage
gather	collapses multiple columns in key-value pairs
spread	spreads a key-value pair across multiple columns
replace_na	replace missing values
fill	fills missing values by using previous entry
separate	turns a single character column into multiple columns
extract	turns each regex capture group into a new column
unite	paste together multiple columns into one
complete	explicitly completes missing data combinations

id	date	wt	age	parent	met1	AM
1	2015-10-20T9:00:00	70	50	9.37	8.05	17.42
1	2015-10-21T9:00:00	70	.	7.58	6.41	14.26
1	2015-10-22T9:00:00	70	.	BQL	unknown	-99

id	date	time	wt	age	parent	met1	AM

id	date	wt	age	parent	met1	AM
1	2015-10-20T9:00:00	70	50	9.37	8.05	17.42
1	2015-10-21T9:00:00	70	.	7.58	6.41	14.26
1	2015-10-22T9:00:00	70	.	BQL	unknown	-99

```
df %>% separate(date,into=c("date","time"),sep="T")
```

id	date	time	wt	age	parent	met1	AM
1	2015-10-20	9:00:00	70	50	9.37	8.05	17.42
1	2015-10-21	9:00:00	70	.	7.58	6.41	14.26
1	2015-10-22	9:00:00	70	.	BQL	unknown	-99

Verb	Usage
gather	collapses multiple columns in key-value pairs
spread	spreads a key-value pair across multiple columns
replace_na	replace missing values
fill	fills missing values by using previous entry
separate	turns a single character column into multiple columns
extract	turns each regex capture group into a new column
unite	paste together multiple columns into one
complete	explicitly completes missing data combinations

Verb	Usage
gather	collapses multiple columns in key-value pairs
spread	spreads a key-value pair across multiple columns
replace_na	replace missing values
fill	fills missing values by using previous entry
separate	turns a single character column into multiple columns
extract	turns each regex capture group into a new column
unite	paste together multiple columns into one
complete	explicitly completes missing data combinations

id	date	time	wt	age	parent	met1	AM
1	2015-10-20	9:00:00	70	50	9.37	8.05	17.42
1	2015-10-21	9:00:00	70	.	7.58	6.41	14.26
1	2015-10-22	9:00:00	70	.	BQL	unknown	-99

```
df %>% unite(datetime,date,time,sep="T")
```

id	datetime	wt	age	parent	met1	AM
1	2015-10-20T9:00:00	70	50	9.37	8.05	17.42
1	2015-10-21T9:00:00	70	.	7.58	6.41	14.26
1	2015-10-22T9:00:00	70	.	BQL	unknown	-99

Verb	Usage
gather	collapses multiple columns in key-value pairs
spread	spreads a key-value pair across multiple columns
replace_na	replace missing values
fill	fills missing values by using previous entry
separate	turns a single character column into multiple columns
extract	turns each regex capture group into a new column
unite	paste together multiple columns into one
complete	explicitly completes missing data combinations

```
df_comp <- dplyr::data_frame(id = c(1,1,1,2,2),
                             time = c(0,1,2,0,1))
```

id	time
1	0
1	1
1	2
2	0
2	1

```
df_comp %>% complete(  id, time)
```

id	time
1	0
1	1
1	2
2	0
2	1
2	2

complete()

- explicitly completes missing data combinations

```
df_comp %>%
    complete(id, time, fill = list())
```

columns to be expanded

what to fill in case of missing combinations