Урок 4. Группировка и агрегирование данных

```
# тот же файл для работы
dat <- read.csv("firms.csv")

# и библиотека
library(tidyverse)
```

В этом уроке речь пойдет, пожалуй, о самых полезных функциях `dplyr` или `tidyverse`.

При работе с данными мы часто сталкиваемся с тем, что нам нужно получить какую-то сводную информацию по переменным. Для этого существует функция `summarise()`. Попробуем пока получить общее число строк в базе данных:

```
dat %>% summarise(total = n())
```

```
##   total
## 1   548
```

Функция `n()` универсальна, она используется для подсчета элементов. К ней мы еще вернемся чуть позже.

Теперь сделаем что-нибудь более интересное. Определим минимальное, максимальное и среднее значение числа продаж в этом датафрейме.

```
dat %>% summarise(avg_sales = mean(SalesInThousands),
          min_sales = min(SalesInThousands),
          max_sales = max(SalesInThousands))
```

```
##   avg_sales min_sales max_sales
## 1        NA        NA        NA
```

Почему R не хочет ничего считать? Потому что в переменно age есть пропущенные значения! Как справиться с этой проблемой? Самое простое и очевидное – удалить `NA` из базы. Но это необязательно. У многих функций в R, работающих с переменными, есть параметр na.rm, который позволяет зафиксировать, исключать ли пропущенные значения (`rm` - от *remove*) при подсчете или нет.

```
dat %>% summarise(avg_sales = mean(SalesInThousands, na.rm = TRUE),
          min_sales = min(SalesInThousands, na.rm = TRUE),
          max_sales = max(SalesInThousands, na.rm = TRUE))
```

```
##   avg_sales min_sales max_sales
## 1  53.46596     17.34     99.65
```

Часто необходимо получить сводную информацию не по всем наблюдениям в базе, а по определенной группе. Для этого сначала нужно сгруппировать данные, основываясь на значениях какой-нибудь переменной. Воспользуемся функцией `group_by()` и посмотрим, сколько в базе фирм разных размеров:

```
dat %>% group_by(MarketSize) %>% summarise(count = n())
```

```
## # A tibble: 4 x 2
##   MarketSize count
##   <fct>      <int>
## 1 ""             1
## 2 Large        167
## 3 Medium       320
## 4 Small         60
```

Так как у одной фирмы не указан размер и значение не является полноценным пропущенным (NA), вместо трех групп фирм мы получили четыре. Поправим:

```
dat <- filter(dat, MarketSize != "") # удалить строки с "" в MarketSize
```

```
dat %>% group_by(MarketSize) %>% summarise(count = n())
```

```
## # A tibble: 3 x 2
##   MarketSize count
##   <fct>      <int>
## 1 Large        167
## 2 Medium       320
## 3 Small         60
```

А теперь посмотрим на среднее число продаж разных типов фирм:

```
dat %>% group_by(MarketSize) %>% summarise(avg_sales = mean(SalesInThousands, na.rm = TRUE))
```

```
## # A tibble: 3 x 2
##   MarketSize avg_sales
##   <fct>          <dbl>
## 1 Large           70.0
## 2 Medium          44.0
## 3 Small           57.4
```

Число наблюдений можно посчитать и по-другому – с помощью функции `tally()`:

```
dat %>% group_by(MarketSize) %>% tally()
```

```
## # A tibble: 3 x 2
##   MarketSize     n
##   <fct>      <int>
## 1 Large        167
## 2 Medium       320
## 3 Small         60
```