String Manipulation

Ao realizar a limpeza e tratamento de dados é muito comum a necessidade de manipular strings. Em alguns casos para remover inconsistências, ou derivar uma informação em uma nova, separar ou juntar informações, e assim por diante.

Base que utilizaremos para o teste

```
base <- read_delim("~/Documentos/base.csv", delim=";")
```

```
## Rows: 10 Columns: 3
## -- Column specification --------------------------------------------------------
## Delimiter: ";"
## chr (2): digito, endereco
## dbl (1): cpf
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
glimpse(base)
```

```
## Rows: 10
## Columns: 3
## $ cpf      <dbl> 368, 6751, 746283915, 12384957, 749603, 6480, 52769814, 68792~
## $ digito   <chr> "#32", "#56", "#48", "#25", "#62", "#38", "#48", "#56", "#25"~
## $ endereco <chr> "Rua Joaquim Nabuco, 123, (-22.9067/-43.1729)", "Rua Visconde~
```

parse_number

Em alguns casos pode ser necessário converter um tipo de dado para numérico, mas antes pode ser necessário remover algum caracter indesejado. Uma forma simples de realizar isso é com a função `parse_number`

```
base$digito %>% head()
```

```
## [1] "#32" "#56" "#48" "#25" "#62" "#38"
```

```
base <- base %>% mutate(digito = parse_number(digito))
base$digito %>% head()
```

```
## [1] 32 56 48 25 62 38
```

separate

Caso um campo contenha muitas informações distintas, é possivel usar essa função para separar de acordo com um caracter separador.

```
base <- base %>% separate(col="endereco",
                          into= c("logradouro","numero","lat_long"),
                          sep= ",")
base
```

```
## # A tibble: 10 x 5
##          cpf digito logradouro                    numero lat_long
##        <dbl>  <dbl> <chr>                         <chr>  <chr>
##  1       368     32 Rua Joaquim Nabuco            " 123" " (-22.9067/-43.1729~
##  2      6751     56 Rua Visconde de Inhauma       " 456" " (-22.9867/-43.1249~
##  3 746283915     48 Rua Barão de Ipanema          " 789" " (-22.9457/-43.1939~
##  4  12384957     25 Rua Santa Luzia               " 101" " (-22.8667/-43.1359~
##  5    749603     62 Rua Senador Dantas            " 202" " (-22.9167/-43.1569~
##  6      6480     38 Rua Marechal Deodoro          " 303" " (-22.9267/-43.1679~
##  7  52769814     48 Rua Conde de Bonfim           " 404" " (-22.9367/-43.1789~
##  8   6879253     56 Rua Nossa Senhora de Copacabana " 505" " (-22.9467/-43.1899~
##  9 847961320     25 Rua Santa Clara               " 606" " (-22.9567/-43.2009~
## 10      9461     32 Rua Dona Mariana              " 707" " (-22.9667/-43.2119~
```

str_sub

Nem sempre tem apenas um único caracter atrapalhando o número para conseguir resolver apenas com o `parse_number`. Uma alternativa é cortar a string para retornar apenas o conteúdo de interesse.

```
base <- base %>%
  mutate(lat  = as.numeric(str_sub(lat_long,3,9)),
         long = as.numeric(str_sub(lat_long,12,-2)))
base
```

```
## # A tibble: 10 x 7
##          cpf digito logradouro                    numero lat_long    lat  long
##        <dbl>  <dbl> <chr>                         <chr>  <chr>     <dbl> <dbl>
##  1       368     32 Rua Joaquim Nabuco            " 123" " (-22.9~ -22.9 -43.2
##  2      6751     56 Rua Visconde de Inhauma       " 456" " (-22.9~ -23.0 -43.1
##  3 746283915     48 Rua Barão de Ipanema          " 789" " (-22.9~ -22.9 -43.2
##  4  12384957     25 Rua Santa Luzia               " 101" " (-22.8~ -22.9 -43.1
##  5    749603     62 Rua Senador Dantas            " 202" " (-22.9~ -22.9 -43.2
##  6      6480     38 Rua Marechal Deodoro          " 303" " (-22.9~ -22.9 -43.2
##  7  52769814     48 Rua Conde de Bonfim           " 404" " (-22.9~ -22.9 -43.2
##  8   6879253     56 Rua Nossa Senhora de Copacabana " 505" " (-22.9~ -22.9 -43.2
##  9 847961320     25 Rua Santa Clara               " 606" " (-22.9~ -23.0 -43.2
## 10      9461     32 Rua Dona Mariana              " 707" " (-22.9~ -23.0 -43.2
```

str_c e str_pad

Quando trabalhamos com um CPF por exemplo, pode acontecer de o dígito vir num campo separado, e/ou o CPF vir numérico e perder os zeros à esquerda. O `str_c` junta os campos enquanto que o `str_pad` completa os zeros.

```
base <- base %>%
  mutate(cpf_completo = str_c(cpf,digito) %>% str_pad(11, pad="0"))

base %>% select(cpf_completo, logradouro, numero, lat, long)
```

```
## # A tibble: 5 x 5
##    cpf_completo logradouro                numero   lat  long
##    <chr>        <chr>                     <chr>  <dbl> <dbl>
##  1 00000036832  Rua Joaquim Nabuco        " 123" -22.9 -43.2
##  2 00000675156  Rua Visconde de Inhauma   " 456" -23.0 -43.1
##  3 74628391548  Rua Barão de Ipanema      " 789" -22.9 -43.2
##  4 01238495725  Rua Santa Luzia           " 101" -22.9 -43.1
##  5 00074960362  Rua Senador Dantas        " 202" -22.9 -43.2
##  6 00000648038  Rua Marechal Deodoro      " 303" -22.9 -43.2
##  7 05276981448  Rua Conde de Bonfim       " 404" -22.9 -43.2
```

```
##  8 00687925356  Rua Nossa Senhora de Copacabana " 505" -22.9 -43.2
##  9 84796132025  Rua Santa Clara                  " 606" -23.0 -43.2
## 10 00000946132  Rua Dona Mariana                 " 707" -23.0 -43.2
```

Uma alternativa ao `str_c` é o `paste0`, o `paste` por padrão insere um espaço, enquanto o `paste0` não faz nenhuma inserção nessa junção, apesar de ambos ter um parâmetro para personalizar esse comportamento.

```
base <- base %>%
  mutate(cpf_completo2 = paste0(cpf,digito) %>% str_pad(11, pad="0"))

base %>% select(cpf_completo2, logradouro, numero, lat, long)
```

```
## # A tibble: 10 x 5
##    cpf_completo2 logradouro                       numero   lat  long
##    <chr>         <chr>                            <chr>  <dbl> <dbl>
##  1 00000036832   Rua Joaquim Nabuco               " 123" -22.9 -43.2
##  2 00000675156   Rua Visconde de Inhauma          " 456" -23.0 -43.1
##  3 74628391548   Rua Barão de Ipanema             " 789" -22.9 -43.2
##  4 01238495725   Rua Santa Luzia                  " 101" -22.9 -43.1
##  5 00074960362   Rua Senador Dantas               " 202" -22.9 -43.2
##  6 00000648038   Rua Marechal Deodoro             " 303" -22.9 -43.2
##  7 05276981448   Rua Conde de Bonfim              " 404" -22.9 -43.2
##  8 00687925356   Rua Nossa Senhora de Copacabana  " 505" -22.9 -43.2
##  9 84796132025   Rua Santa Clara                  " 606" -23.0 -43.2
## 10 00000946132   Rua Dona Mariana                 " 707" -23.0 -43.2
```