

R - Practice 01

Sangkon Han(sangkon@pusan.ac.kr)

2023-08-03

Contents

dplyr & tidyr	2
Manipulate variables(columns)	2
select(), rename()	2
mutate() / transmute()	4
Manipulate variables(row)	5
filter(), slice()	5
arrange	8
distinct	9
Sample rows	10
summarise	11
group_by()	11
count()	13
pipe operator %>%	13
pivoting()	14
separating and uniting	16
separate()	16
unite()	30

```
df <- mpg
str(df)

## tibble [234 x 11] (S3: tbl_df/tbl/data.frame)
## $ manufacturer: chr [1:234] "audi" "audi" "audi" "audi" ...
## $ model       : chr [1:234] "a4" "a4" "a4" "a4" ...
## $ displ       : num [1:234] 1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
## $ year        : int [1:234] 1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ...
## $ cyl         : int [1:234] 4 4 4 4 6 6 6 4 4 4 ...
## $ trans       : chr [1:234] "auto(l5)" "manual(m5)" "manual(m6)" "auto(av)" ...
## $ drv         : chr [1:234] "f" "f" "f" "f" ...
## $ cty         : int [1:234] 18 21 20 21 16 18 18 16 20 ...
## $ hwy         : int [1:234] 29 29 31 30 26 26 27 26 25 28 ...
## $ fl          : chr [1:234] "p" "p" "p" "p" ...
## $ class       : chr [1:234] "compact" "compact" "compact" "compact" ...

nrow(df); ncol(df)

## [1] 234
## [1] 11
```

dplyr & tidyr

Manipulate variables(columns)

`select()`, `rename()`

```
df.car.info <- select(df, manufacturer, model, year)
```

```
select(df, starts_with(match = "m"))
```

```
## # A tibble: 234 x 2
##   manufacturer model
##   <chr>         <chr>
## 1 audi         a4
## 2 audi         a4
## 3 audi         a4
## 4 audi         a4
## 5 audi         a4
## 6 audi         a4
## 7 audi         a4
## 8 audi         a4 quattro
## 9 audi         a4 quattro
## 10 audi        a4 quattro
## # i 224 more rows
```

```
select(df, contains(match = "r"))
```

```
## # A tibble: 234 x 4
##   manufacturer year trans      drv
##   <chr>         <int> <chr>   <chr>
## 1 audi         1999 auto(l5)  f
## 2 audi         1999 manual(m5) f
## 3 audi         2008 manual(m6) f
## 4 audi         2008 auto(av)   f
## 5 audi         1999 auto(l5)  f
## 6 audi         1999 manual(m5) f
## 7 audi         2008 auto(av)   f
## 8 audi         1999 manual(m5) 4
## 9 audi         1999 auto(l5)  4
## 10 audi        2008 manual(m6) 4
## # i 224 more rows
```

```
select(df, ends_with(match = "y"))
```

```
## # A tibble: 234 x 2
##   cty    hwy
##   <int> <int>
## 1    18    29
## 2    21    29
## 3    20    31
## 4    21    30
## 5    16    26
## 6    18    26
## 7    18    27
## 8    18    26
## 9    16    25
```

```
## 10    20    28
## # i 224 more rows
```

```
select(df, 1:3)
```

```
## # A tibble: 234 x 3
##   manufacturer model      displ
##   <chr>         <chr>    <dbl>
## 1 audi         a4        1.8
## 2 audi         a4        1.8
## 3 audi         a4         2
## 4 audi         a4         2
## 5 audi         a4        2.8
## 6 audi         a4        2.8
## 7 audi         a4        3.1
## 8 audi         a4 quattro  1.8
## 9 audi         a4 quattro  1.8
## 10 audi        a4 quattro   2
## # i 224 more rows
```

```
select(df, c(2,5,7))
```

```
## # A tibble: 234 x 3
##   model      cyl drv
##   <chr>    <int> <chr>
## 1 a4         4 f
## 2 a4         4 f
## 3 a4         4 f
## 4 a4         4 f
## 5 a4         6 f
## 6 a4         6 f
## 7 a4         6 f
## 8 a4 quattro  4 4
## 9 a4 quattro  4 4
## 10 a4 quattro  4 4
## # i 224 more rows
```

```
select(df, 9:11)
```

```
## # A tibble: 234 x 3
##   hwy fl      class
##   <int> <chr> <chr>
## 1    29 p    compact
## 2    29 p    compact
## 3    31 p    compact
## 4    30 p    compact
## 5    26 p    compact
## 6    26 p    compact
## 7    27 p    compact
## 8    26 p    compact
## 9    25 p    compact
## 10   28 p    compact
## # i 224 more rows
```

```
select(df, (ncol(df)-2):ncol(df))
```

```
## # A tibble: 234 x 3
```

```
##      hwy fl      class
##      <int> <chr> <chr>
## 1      29 p      compact
## 2      29 p      compact
## 3      31 p      compact
## 4      30 p      compact
## 5      26 p      compact
## 6      26 p      compact
## 7      27 p      compact
## 8      26 p      compact
## 9      25 p      compact
## 10     28 p      compact
## # i 224 more rows
```

```
df1 <- rename(df, mnfc = manufacturer, mod = model)
df1 <- select(df, mnfc = manufacturer, mod = model, everything())
```

mutate() / transmute()

```
df <- mutate(df, `avg miles per gallon` = (cty + hwy) / 2)
df <- mutate(df,
  car = paste(manufacturer, model, sep = " "),
  `cyl / trans` = paste(cyl, " cylinders", " / ", trans, " transmission", sep = ""))
```

```
df1 <- transmute(df,
  `avg miles per gallon` = (cty + hwy) / 2)
df1
```

```
## # A tibble: 234 x 1
##   `avg miles per gallon`
##               <dbl>
## 1                23.5
## 2                25
## 3                25.5
## 4                25.5
## 5                21
## 6                22
## 7                22.5
## 8                22
## 9                20.5
## 10               24
## # i 224 more rows
```

```
df2 <- mutate(df,
  car = paste(manufacturer, model, sep = " "),
  `cyl / trans` = paste(cyl, " cylinders", " / ", trans, " transmission", sep = ""))
df2
```

```
## # A tibble: 234 x 14
##   manufacturer model      displ  year   cyl trans drv      cty  hwy fl      class
##   <chr>          <chr>    <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 audi          a4        1.8  1999    4 auto~ f      18   29 p      comp~
## 2 audi          a4        1.8  1999    4 manu~ f      21   29 p      comp~
## 3 audi          a4         2    2008    4 manu~ f      20   31 p      comp~
## 4 audi          a4         2    2008    4 auto~ f      21   30 p      comp~
## 5 audi          a4        2.8  1999    6 auto~ f      16   26 p      comp~
```

```
## 6 audi      a4      2.8 1999    6 manu~ f      18    26 p    comp~
## 7 audi      a4      3.1 2008    6 auto~ f      18    27 p    comp~
## 8 audi      a4 quattro 1.8 1999    4 manu~ 4      18    26 p    comp~
## 9 audi      a4 quattro 1.8 1999    4 auto~ 4      16    25 p    comp~
## 10 audi     a4 quattro 2    2008    4 manu~ 4      20    28 p    comp~
## # i 224 more rows
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
df2 <- transmute(df,
  car = paste(manufacturer, model, sep = " "),
  `cyl / trans` = paste(cyl, " cylinders", " / ", trans, " transmission", sep = ""))
df2
```

```
## # A tibble: 234 x 2
##   car      `cyl / trans`
##   <chr>      <chr>
## 1 audi a4      4 cylinders / auto(l5) transmission
## 2 audi a4      4 cylinders / manual(m5) transmission
## 3 audi a4      4 cylinders / manual(m6) transmission
## 4 audi a4      4 cylinders / auto(av) transmission
## 5 audi a4      6 cylinders / auto(l5) transmission
## 6 audi a4      6 cylinders / manual(m5) transmission
## 7 audi a4      6 cylinders / auto(av) transmission
## 8 audi a4 quattro 4 cylinders / manual(m5) transmission
## 9 audi a4 quattro 4 cylinders / auto(l5) transmission
## 10 audi a4 quattro 4 cylinders / manual(m6) transmission
## # i 224 more rows
```

Manipulate variables(row)

filter(), slice()

```
filter(df, manufacturer == "audi")
```

```
## # A tibble: 18 x 14
##   manufacturer model   displ  year  cyl trans drv   cty   hwy fl   class
##   <chr>          <chr>   <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 audi          a4       1.8  1999    4 auto~ f    18    29 p    comp~
## 2 audi          a4       1.8  1999    4 manu~ f    21    29 p    comp~
## 3 audi          a4       2    2008    4 manu~ f    20    31 p    comp~
## 4 audi          a4       2    2008    4 auto~ f    21    30 p    comp~
## 5 audi          a4       2.8  1999    6 auto~ f    16    26 p    comp~
## 6 audi          a4       2.8  1999    6 manu~ f    18    26 p    comp~
## 7 audi          a4       3.1  2008    6 auto~ f    18    27 p    comp~
## 8 audi          a4 quattro 1.8  1999    4 manu~ 4    18    26 p    comp~
## 9 audi          a4 quattro 1.8  1999    4 auto~ 4    16    25 p    comp~
## 10 audi         a4 quattro 2    2008    4 manu~ 4    20    28 p    comp~
## 11 audi         a4 quattro 2    2008    4 auto~ 4    19    27 p    comp~
## 12 audi         a4 quattro 2.8  1999    6 auto~ 4    15    25 p    comp~
## 13 audi         a4 quattro 2.8  1999    6 manu~ 4    17    25 p    comp~
## 14 audi         a4 quattro 3.1  2008    6 auto~ 4    17    25 p    comp~
## 15 audi         a4 quattro 3.1  2008    6 manu~ 4    15    25 p    comp~
## 16 audi         a6 quattro 2.8  1999    6 auto~ 4    15    24 p    mids~
## 17 audi         a6 quattro 3.1  2008    6 auto~ 4    17    25 p    mids~
```

```
## 18 audi          a6 quattro    4.2  2008      8 auto~ 4          16      23 p      mids~
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
```

```
filter(df, manufacturer == "audi" & year == 1999)
```

```
## # A tibble: 9 x 14
##   manufacturer model      displ year   cyl trans  drv      cty   hwy fl      class
##   <chr>          <chr>    <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 audi          a4          1.8  1999     4 auto(~ f      18    29 p      comp~
## 2 audi          a4          1.8  1999     4 manua~ f      21    29 p      comp~
## 3 audi          a4          2.8  1999     6 auto(~ f      16    26 p      comp~
## 4 audi          a4          2.8  1999     6 manua~ f      18    26 p      comp~
## 5 audi          a4 quattro  1.8  1999     4 manua~ 4      18    26 p      comp~
## 6 audi          a4 quattro  1.8  1999     4 auto(~ 4      16    25 p      comp~
## 7 audi          a4 quattro  2.8  1999     6 auto(~ 4      15    25 p      comp~
## 8 audi          a4 quattro  2.8  1999     6 manua~ 4      17    25 p      comp~
## 9 audi          a6 quattro  2.8  1999     6 auto(~ 4      15    24 p      mids~
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
```

```
df1 <- filter(df, manufacturer == "audi" | manufacturer == "dodge")
df2 <- filter(df, manufacturer %in% c("audi", "dodge"))
```

```
filter(df, hwy >= 30)
```

```
## # A tibble: 26 x 14
##   manufacturer model      displ year   cyl trans  drv      cty   hwy fl      class
##   <chr>          <chr>    <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 audi          a4          2      2008     4 manual(m~ f      20    31 p      comp~
## 2 audi          a4          2      2008     4 auto(av) f      21    30 p      comp~
## 3 chevrolet     malibu    2.4      2008     4 auto(14) f      22    30 r      mids~
## 4 honda          civic    1.6      1999     4 manual(m~ f      28    33 r      subc~
## 5 honda          civic    1.6      1999     4 auto(14) f      24    32 r      subc~
## 6 honda          civic    1.6      1999     4 manual(m~ f      25    32 r      subc~
## 7 honda          civic    1.6      1999     4 auto(14) f      24    32 r      subc~
## 8 honda          civic    1.8      2008     4 manual(m~ f      26    34 r      subc~
## 9 honda          civic    1.8      2008     4 auto(15) f      25    36 r      subc~
## 10 honda         civic    1.8      2008     4 auto(15) f      24    36 c      subc~
## # i 16 more rows
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
```

```
filter(df, year != 1999)
```

```
## # A tibble: 117 x 14
##   manufacturer model      displ year   cyl trans  drv      cty   hwy fl      class
##   <chr>          <chr>    <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 audi          a4          2      2008     4 manu~ f      20    31 p      comp~
## 2 audi          a4          2      2008     4 auto~ f      21    30 p      comp~
## 3 audi          a4          3.1      2008     6 auto~ f      18    27 p      comp~
## 4 audi          a4 quattro  2      2008     4 manu~ 4      20    28 p      comp~
## 5 audi          a4 quattro  2      2008     4 auto~ 4      19    27 p      comp~
## 6 audi          a4 quattro  3.1      2008     6 auto~ 4      17    25 p      comp~
## 7 audi          a4 quattro  3.1      2008     6 manu~ 4      15    25 p      comp~
## 8 audi          a6 quattro  3.1      2008     6 auto~ 4      17    25 p      mids~
```

```
## 9 audi          a6 quattro  4.2 2008    8 auto~ 4      16    23 p    mids~
## 10 chevrolet    c1500 sub~  5.3 2008    8 auto~ r      14    20 r    suv
## # i 107 more rows
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
```

```
slice(df, 1:5)
```

```
## # A tibble: 5 x 14
##   manufacturer model displ  year   cyl trans      drv   cty   hwy fl   class
##   <chr>          <chr> <dbl> <int> <int> <chr>    <chr> <int> <int> <chr> <chr>
## 1 audi          a4      1.8 1999     4 auto(l5) f      18    29 p    compa~
## 2 audi          a4      1.8 1999     4 manual(m5) f      21    29 p    compa~
## 3 audi          a4      2    2008     4 manual(m6) f      20    31 p    compa~
## 4 audi          a4      2    2008     4 auto(av) f      21    30 p    compa~
## 5 audi          a4      2.8 1999     6 auto(l5) f      16    26 p    compa~
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
```

```
slice(df, 20:30)
```

```
## # A tibble: 11 x 14
##   manufacturer model      displ  year   cyl trans drv   cty   hwy fl   class
##   <chr>          <chr>    <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 chevrolet    c1500 sub~  5.3 2008    8 auto~ r      11    15 e    suv
## 2 chevrolet    c1500 sub~  5.3 2008    8 auto~ r      14    20 r    suv
## 3 chevrolet    c1500 sub~  5.7 1999    8 auto~ r      13    17 r    suv
## 4 chevrolet    c1500 sub~  6    2008    8 auto~ r      12    17 r    suv
## 5 chevrolet    corvette  5.7 1999    8 manu~ r      16    26 p    2sea~
## 6 chevrolet    corvette  5.7 1999    8 auto~ r      15    23 p    2sea~
## 7 chevrolet    corvette  6.2 2008    8 manu~ r      16    26 p    2sea~
## 8 chevrolet    corvette  6.2 2008    8 auto~ r      15    25 p    2sea~
## 9 chevrolet    corvette  7    2008    8 manu~ r      15    24 p    2sea~
## 10 chevrolet   k1500 tah~  5.3 2008    8 auto~ 4      14    19 r    suv
## 11 chevrolet   k1500 tah~  5.3 2008    8 auto~ 4      11    14 e    suv
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
```

```
slice(df, (nrow(df)-9):nrow(df))
```

```
## # A tibble: 10 x 14
##   manufacturer model      displ  year   cyl trans drv   cty   hwy fl   class
##   <chr>          <chr>    <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 volkswagen    new beetle  2    1999     4 auto~ f      19    26 r    subc~
## 2 volkswagen    new beetle  2.5 2008     5 manu~ f      20    28 r    subc~
## 3 volkswagen    new beetle  2.5 2008     5 auto~ f      20    29 r    subc~
## 4 volkswagen    passat     1.8 1999     4 manu~ f      21    29 p    mids~
## 5 volkswagen    passat     1.8 1999     4 auto~ f      18    29 p    mids~
## 6 volkswagen    passat     2    2008     4 auto~ f      19    28 p    mids~
## 7 volkswagen    passat     2    2008     4 manu~ f      21    29 p    mids~
## 8 volkswagen    passat     2.8 1999     6 auto~ f      16    26 p    mids~
## 9 volkswagen    passat     2.8 1999     6 manu~ f      18    26 p    mids~
## 10 volkswagen    passat     3.6 2008     6 auto~ f      17    26 p    mids~
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
```

arrange

Sort rows by year (ascending order)

```
arrange(df, year)
```

```
## # A tibble: 234 x 14
##   manufacturer model      displ  year   cyl trans drv      cty   hwy fl      class
##   <chr>          <chr>    <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 audi          a4          1.8  1999     4 auto~ f      18    29 p      comp~
## 2 audi          a4          1.8  1999     4 manu~ f      21    29 p      comp~
## 3 audi          a4          2.8  1999     6 auto~ f      16    26 p      comp~
## 4 audi          a4          2.8  1999     6 manu~ f      18    26 p      comp~
## 5 audi          a4 quattro  1.8  1999     4 manu~ 4      18    26 p      comp~
## 6 audi          a4 quattro  1.8  1999     4 auto~ 4      16    25 p      comp~
## 7 audi          a4 quattro  2.8  1999     6 auto~ 4      15    25 p      comp~
## 8 audi          a4 quattro  2.8  1999     6 manu~ 4      17    25 p      comp~
## 9 audi          a6 quattro  2.8  1999     6 auto~ 4      15    24 p      mids~
## 10 chevrolet    c1500 sub~  5.7  1999     8 auto~ r      13    17 r      suv
## # i 224 more rows
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
```

Sort rows by year (descending order)

```
arrange(df, desc(year))
```

```
## # A tibble: 234 x 14
##   manufacturer model      displ  year   cyl trans drv      cty   hwy fl      class
##   <chr>          <chr>    <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 audi          a4          2    2008     4 manu~ f      20    31 p      comp~
## 2 audi          a4          2    2008     4 auto~ f      21    30 p      comp~
## 3 audi          a4          3.1  2008     6 auto~ f      18    27 p      comp~
## 4 audi          a4 quattro  2    2008     4 manu~ 4      20    28 p      comp~
## 5 audi          a4 quattro  2    2008     4 auto~ 4      19    27 p      comp~
## 6 audi          a4 quattro  3.1  2008     6 auto~ 4      17    25 p      comp~
## 7 audi          a4 quattro  3.1  2008     6 manu~ 4      15    25 p      comp~
## 8 audi          a6 quattro  3.1  2008     6 auto~ 4      17    25 p      mids~
## 9 audi          a6 quattro  4.2  2008     8 auto~ 4      16    23 p      mids~
## 10 chevrolet    c1500 sub~  5.3  2008     8 auto~ r      14    20 r      suv
## # i 224 more rows
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
```

Sort rows by year (ascending order), cyl and displ

```
df.sort <- arrange(df, year, cyl, displ)
```

```
df.sort
```

```
## # A tibble: 234 x 14
##   manufacturer model      displ  year   cyl trans drv      cty   hwy fl      class
##   <chr>          <chr>    <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 honda          civic      1.6  1999     4 manu~ f      28    33 r      subc~
## 2 honda          civic      1.6  1999     4 auto~ f      24    32 r      subc~
## 3 honda          civic      1.6  1999     4 manu~ f      25    32 r      subc~
## 4 honda          civic      1.6  1999     4 manu~ f      23    29 p      subc~
## 5 honda          civic      1.6  1999     4 auto~ f      24    32 r      subc~
## 6 audi          a4          1.8  1999     4 auto~ f      18    29 p      comp~
## 7 audi          a4          1.8  1999     4 manu~ f      21    29 p      comp~
```



```
## 8 audi          a4 quattro  1.8 1999    4 manu~ 4        18    26 p    comp~
## 9 audi          a4 quattro  1.8 1999    4 auto~ 4        16    25 p    comp~
## 10 toyota       corolla    1.8 1999    4 auto~ f        24    30 r    comp~
## # i 224 more rows
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
```

distinct

```
df.example <- data.frame(id = 1:3, name = c("John", "Max", "Julia"))
df.example <- bind_rows(df.example, slice(df.example, 2)) # create duplicate of 2nd row
df.example <- arrange(df.example, id)
df.example
```

```
##   id  name
## 1  1  John
## 2  2   Max
## 3  2   Max
## 4  3 Julia
```

```
# show table without duplicates
distinct(df.example)
```

```
##   id  name
## 1  1  John
## 2  2   Max
## 3  3 Julia
```

```
# Back to mpg example - lets create a table with duplicates
df.dupl <- select(df, manufacturer, model)
df.dupl
```

```
## # A tibble: 234 x 2
##   manufacturer model
##   <chr>         <chr>
## 1 audi          a4
## 2 audi          a4
## 3 audi          a4
## 4 audi          a4
## 5 audi          a4
## 6 audi          a4
## 7 audi          a4
## 8 audi          a4 quattro
## 9 audi          a4 quattro
## 10 audi          a4 quattro
## # i 224 more rows
```

```
# Keep only unique rows without duplicates
df.nodupl <- distinct(df.dupl)
df.nodupl
```

```
## # A tibble: 38 x 2
##   manufacturer model
##   <chr>         <chr>
## 1 audi          a4
## 2 audi          a4 quattro
```

```
## 3 audi      a6 quattro
## 4 chevrolet c1500 suburban 2wd
## 5 chevrolet corvette
## 6 chevrolet k1500 tahoe 4wd
## 7 chevrolet malibu
## 8 dodge     caravan 2wd
## 9 dodge     dakota pickup 4wd
## 10 dodge    durango 4wd
## # i 28 more rows
```

Sample rows

```
# sample_n() - Filter n randomly selected rows
set.seed(42)
```

```
# 10 randomly selected rows without replacement
sample_n(df, size = 10, replace = F)
```

```
## # A tibble: 10 x 14
##   manufacturer model      displ  year  cyl trans drv   cty   hwy fl   class
##   <chr>          <chr>    <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 dodge         dakota pi~    3.7  2008    6 manu~ 4     15    19 r    pick~
## 2 volkswagen    passat      1.8  1999    4 auto~ f     18    29 p    mids~
## 3 dodge         ram 1500 ~    4.7  2008    8 manu~ 4     12    16 r    pick~
## 4 nissan        pathfinde~    4    2008    6 auto~ 4     14    20 p    suv
## 5 dodge         ram 1500 ~    5.9  1999    8 auto~ 4     11    15 r    pick~
## 6 volkswagen    passat      1.8  1999    4 manu~ f     21    29 p    mids~
## 7 nissan        altima      3.5  2008    6 manu~ f     19    27 p    mids~
## 8 hyundai       tiburon     2.7  2008    6 manu~ f     17    24 r    subc~
## 9 volkswagen    passat      3.6  2008    6 auto~ f     17    26 p    mids~
## 10 jeep         grand che~    4.7  2008    8 auto~ 4     14    19 r    suv
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
```

```
# 10 randomly selected rows with replacement
sample_n(df, size = 10, replace = T)
```

```
## # A tibble: 10 x 14
##   manufacturer model      displ  year  cyl trans drv   cty   hwy fl   class
##   <chr>          <chr>    <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 dodge         caravan 2~    3.8  2008    6 auto~ f     16    23 r    mini~
## 2 chevrolet     corvette    5.7  1999    8 manu~ r     16    26 p    2sea~
## 3 dodge         ram 1500 ~    5.2  1999    8 auto~ 4     11    15 r    pick~
## 4 honda         civic      1.6  1999    4 manu~ f     28    33 r    subc~
## 5 ford          f150 pick~    5.4  1999    8 auto~ 4     11    15 r    pick~
## 6 subaru        forester ~    2.5  2008    4 auto~ 4     18    23 p    suv
## 7 hyundai       sonata      2.4  1999    4 manu~ f     18    27 r    mids~
## 8 chevrolet     c1500 sub~    5.3  2008    8 auto~ r     11    15 e    suv
## 9 nissan        pathfinde~    5.6  2008    8 auto~ 4     12    18 p    suv
## 10 hyundai      sonata      2.5  1999    6 manu~ f     18    26 r    mids~
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
```

```
# sample_frac() - Filter a fraction of randomly selected rows
# 10% of table rows randomly selected
sample_frac(df, size = 0.1, replace = F)
```

```
## # A tibble: 23 x 14
##   manufacturer model      displ  year   cyl trans drv      cty   hwy fl      class
##   <chr>          <chr>    <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 hyundai      sonata        2.4  2008     4 auto~ f      21    30 r      mids~
## 2 land rover   range rov~     4   1999     8 auto~ 4      11    15 p      suv
## 3 dodge        caravan 2~    3.3  1999     6 auto~ f      16    22 r      mini~
## 4 ford         f150 pick~    5.4  1999     8 auto~ 4      11    15 r      pick~
## 5 chevrolet    corvette      6.2  2008     8 auto~ r      15    25 p      2sea~
## 6 subaru       forester ~    2.5  2008     4 auto~ 4      20    26 r      suv
## 7 volkswagen   new beetle    1.9  1999     4 auto~ f      29    41 d      subc~
## 8 hyundai      sonata        2.4  1999     4 auto~ f      18    26 r      mids~
## 9 audi         a4            2.8  1999     6 auto~ f      16    26 p      comp~
## 10 volkswagen  gti           2.8  1999     6 manu~ f      17    24 r      comp~
## # i 13 more rows
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
```

summarise

```
# Calculate average hwy
summarise(df, `mean hwy` = mean(hwy))
```

```
## # A tibble: 1 x 1
##   `mean hwy`
##   <dbl>
## 1      23.4
```

```
# Count table rows, and count distinct car models
summarise(df, rows = n(), `nr models` = n_distinct(model))
```

```
## # A tibble: 1 x 2
##   rows `nr models`
##   <int>     <int>
## 1   234         38
```

```
# Calculate min / max hwy & cty
summarise(df,
  `min hwy` = min(hwy),
  `min cty` = min(cty),
  `max hwy` = max(hwy),
  `max cty` = max(cty))
```

```
## # A tibble: 1 x 4
##   `min hwy` `min cty` `max hwy` `max cty`
##   <int>     <int>     <int>     <int>
## 1      12         9      44      35
```

group_by()

```
# Group cars by manufacturer
group_by(df, manufacturer)
```

```
## # A tibble: 234 x 14
## # Groups:   manufacturer [15]
```

```
##      manufacturer model      displ  year   cyl trans drv      cty   hwy fl      class
##      <chr>          <chr>      <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 audi            a4          1.8  1999    4 auto~ f      18    29 p    comp~
## 2 audi            a4          1.8  1999    4 manu~ f      21    29 p    comp~
## 3 audi            a4          2    2008    4 manu~ f      20    31 p    comp~
## 4 audi            a4          2    2008    4 auto~ f      21    30 p    comp~
## 5 audi            a4          2.8  1999    6 auto~ f      16    26 p    comp~
## 6 audi            a4          2.8  1999    6 manu~ f      18    26 p    comp~
## 7 audi            a4          3.1  2008    6 auto~ f      18    27 p    comp~
## 8 audi            a4 quattro  1.8  1999    4 manu~ 4      18    26 p    comp~
## 9 audi            a4 quattro  1.8  1999    4 auto~ 4      16    25 p    comp~
## 10 audi           a4 quattro  2    2008    4 manu~ 4      20    28 p    comp~
## # i 224 more rows
## # i 3 more variables: `avg miles per gallon` <dbl>, car <chr>,
## #   `cyl / trans` <chr>
```

```
# Combine summarise() & group_by() - summary statistics for grouped data
# Count number of cars for each manufacturer
summarise(group_by(df, manufacturer), cars = n())
```

```
## # A tibble: 15 x 2
##   manufacturer cars
##   <chr>         <int>
## 1 audi           18
## 2 chevrolet      19
## 3 dodge          37
## 4 ford           25
## 5 honda           9
## 6 hyundai        14
## 7 jeep            8
## 8 land rover      4
## 9 lincoln         3
## 10 mercury        4
## 11 nissan         13
## 12 pontiac        5
## 13 subaru         14
## 14 toyota         34
## 15 volkswagen     27
```

```
# Calculate mean / min / max hwy for each model
summarise(group_by(df, model),
  `mean hwy` = mean(hwy),
  `min hwy` = min(hwy),
  `max hwy` = max(hwy))
```

```
## # A tibble: 38 x 4
##   model      `mean hwy` `min hwy` `max hwy`
##   <chr>          <dbl>    <int>    <int>
## 1 4runner 4wd      18.8        17        20
## 2 a4              28.3        26        31
## 3 a4 quattro      25.8        25        28
## 4 a6 quattro      24         23        25
## 5 altima          28.7        26        32
## 6 c1500 suburban  17.8        15        20
## 7 camry           28.3        26        31
```

```
## 8 camry solara          28.1      26      31
## 9 caravan 2wd          22.4      17      24
## 10 civic                32.6      29      36
## # i 28 more rows
```

count()

```
# Count number of table rows
count(df)
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1   234
```

```
# Count number of cars per model
count(group_by(df, model))
```

```
## # A tibble: 38 x 2
## # Groups:   model [38]
##   model          n
##   <chr>        <int>
## 1 4runner 4wd         6
## 2 a4              7
## 3 a4 quattro        8
## 4 a6 quattro        3
## 5 altima           6
## 6 c1500 suburban 2wd  5
## 7 camry            7
## 8 camry solara       7
## 9 caravan 2wd       11
## 10 civic           9
## # i 28 more rows
```

pipe operator %>%

```
df %>%
  filter(manufacturer == "audi") %>%
  count()
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1    18
```

```
df %>%
  filter(manufacturer %in% c("dodge", "chevrolet")) %>%
  select(manufacturer, model, year, class)
```

```
## # A tibble: 56 x 4
##   manufacturer model          year class
##   <chr>         <chr>        <int> <chr>
## 1 chevrolet    c1500 suburban 2wd  2008 suv
## 2 chevrolet    c1500 suburban 2wd  2008 suv
## 3 chevrolet    c1500 suburban 2wd  2008 suv
## 4 chevrolet    c1500 suburban 2wd  1999 suv
```

```
## 5 chevrolet      c1500 suburban 2wd  2008 suv
## 6 chevrolet      corvette              1999 2seater
## 7 chevrolet      corvette              1999 2seater
## 8 chevrolet      corvette              2008 2seater
## 9 chevrolet      corvette              2008 2seater
## 10 chevrolet     corvette              2008 2seater
## # i 46 more rows
```

```
df %>%
  group_by(manufacturer, model, class, trans) %>%
  summarise(`mean hwy` = mean(hwy), cars = n()) %>%
  ungroup() %>%
  filter(`mean hwy` > 30) %>%
  arrange(desc(`mean hwy`))
```

```
## # A tibble: 10 x 6
##   manufacturer model      class      trans      `mean hwy` cars
##   <chr>          <chr>    <chr>    <chr>    <dbl> <int>
## 1 honda         civic    subcompact auto(15)    36     2
## 2 toyota        corolla  compact   manual(m5)   36     2
## 3 toyota        corolla  compact   auto(14)    34     2
## 4 volkswagen    new beetle subcompact manual(m5)   33.7    3
## 5 volkswagen    new beetle subcompact auto(14)    33.5    2
## 6 honda         civic    subcompact auto(14)    32     2
## 7 honda         civic    subcompact manual(m5)   32     4
## 8 volkswagen    jetta    compact   manual(m5)   31.5    4
## 9 audi          a4       compact   manual(m6)   31     1
## 10 toyota       camry    midsize    auto(15)    31     1
```

pivoting()

```
table.long <- data.frame(id = 1:6,
  type = c("a", "b", "a", "c", "c", "a"),
  count = c(20, 50, 45, 15, 12, 5))
```

```
table.long
```

```
##   id type count
## 1  1   a    20
## 2  2   b    50
## 3  3   a    45
## 4  4   c    15
## 5  5   c    12
## 6  6   a     5
```

```
table.wide <- pivot_wider(table.long,
  names_from = type,
  values_from = count)
```

```
table.wide
```

```
## # A tibble: 6 x 4
##   id      a      b      c
##   <int> <dbl> <dbl> <dbl>
## 1     1    20    NA    NA
## 2     2    NA    50    NA
## 3     3    45    NA    NA
```

```
## 4      4      NA      NA      15
## 5      5      NA      NA      12
## 6      6       5      NA      NA
```

```
table.long1 <- pivot_longer(table.wide,
                             cols = c("a", "b", "c"),
                             names_to = "type",
                             values_to = "count",
                             values_drop_na = T)

table.long1
```

```
## # A tibble: 6 x 3
##       id type  count
##   <int> <chr> <dbl>
## 1     1  a      20
## 2     2  b      50
## 3     3  a      45
## 4     4  c      15
## 5     5  c      12
## 6     6  a       5
```

```
df.long <- df %>%
  filter(manufacturer %in% c("jeep", "land rover", "hyundai")) %>%
  select(model, trans, hwy) %>%
  group_by(model, trans) %>%
  summarise(`mean hwy` = mean(hwy)) %>%
  ungroup()

df.long
```

```
## # A tibble: 10 x 3
##       model          trans    `mean hwy`
##   <chr>          <chr>      <dbl>
## 1 grand cherokee 4wd auto(14)      18.5
## 2 grand cherokee 4wd auto(15)      17.3
## 3 range rover    auto(14)       15
## 4 range rover    auto(s6)       18
## 5 sonata         auto(14)      27.3
## 6 sonata         auto(15)       28
## 7 sonata         manual(m5)     28
## 8 tiburon        auto(14)      25.7
## 9 tiburon        manual(m5)     27
## 10 tiburon       manual(m6)     24
```

```
df.wide <- df.long %>%
  pivot_wider(names_from = trans,
              values_from = `mean hwy`)

df.wide
```

```
## # A tibble: 4 x 6
##       model          `auto(14)` `auto(15)` `auto(s6)` `manual(m5)` `manual(m6)`
##   <chr>          <dbl>      <dbl>      <dbl>      <dbl>      <dbl>
## 1 grand cherokee 4wd      18.5        17.3         NA         NA         NA
## 2 range rover      15          NA          18          NA         NA
## 3 sonata          27.3         28          NA          28         NA
## 4 tiburon         25.7         NA          NA          27         24
```

```
df.long1 <- df.wide %>%
  pivot_longer(-model, # exclude column "model" and use all remaining columns!!!
               names_to = "trans",
               values_to = "mean hwy",
               values_drop_na = T)
df.long1
```

```
## # A tibble: 10 x 3
##   model      trans    `mean hwy`
##   <chr>      <chr>      <dbl>
## 1 grand cherokee 4wd auto(14)      18.5
## 2 grand cherokee 4wd auto(15)      17.3
## 3 range rover    auto(14)       15
## 4 range rover    auto(s6)       18
## 5 sonata         auto(14)      27.3
## 6 sonata         auto(15)       28
## 7 sonata         manual(m5)     28
## 8 tiburon        auto(14)      25.7
## 9 tiburon        manual(m5)     27
## 10 tiburon       manual(m6)     24
```

separating and uniting

```
dates <- seq.Date(from = as.Date("2021-01-01"), to = as.Date("2021-12-31"), by = "day") # generate dates
table <- data.frame(date = dates)
table %>% head(); table %>% tail()
```

```
##           date
## 1 2021-01-01
## 2 2021-01-02
## 3 2021-01-03
## 4 2021-01-04
## 5 2021-01-05
## 6 2021-01-06
##
##           date
## 360 2021-12-26
## 361 2021-12-27
## 362 2021-12-28
## 363 2021-12-29
## 364 2021-12-30
## 365 2021-12-31
```

separate()

```
table.sep <- table %>%
  separate(data = .,
           col = date,
           into = c("year", "month", "dayofmonth"),
           sep = "-") %>%
  mutate(month = as.numeric(month),
         dayofmonth = as.numeric(dayofmonth)) %>%
  arrange(year, month, dayofmonth)
table.sep
```


##	year	month	dayofmonth
## 1	2021	1	1
## 2	2021	1	2
## 3	2021	1	3
## 4	2021	1	4
## 5	2021	1	5
## 6	2021	1	6
## 7	2021	1	7
## 8	2021	1	8
## 9	2021	1	9
## 10	2021	1	10
## 11	2021	1	11
## 12	2021	1	12
## 13	2021	1	13
## 14	2021	1	14
## 15	2021	1	15
## 16	2021	1	16
## 17	2021	1	17
## 18	2021	1	18
## 19	2021	1	19
## 20	2021	1	20
## 21	2021	1	21
## 22	2021	1	22
## 23	2021	1	23
## 24	2021	1	24
## 25	2021	1	25
## 26	2021	1	26
## 27	2021	1	27
## 28	2021	1	28
## 29	2021	1	29
## 30	2021	1	30
## 31	2021	1	31
## 32	2021	2	1
## 33	2021	2	2
## 34	2021	2	3
## 35	2021	2	4
## 36	2021	2	5
## 37	2021	2	6
## 38	2021	2	7
## 39	2021	2	8
## 40	2021	2	9
## 41	2021	2	10
## 42	2021	2	11
## 43	2021	2	12
## 44	2021	2	13
## 45	2021	2	14
## 46	2021	2	15
## 47	2021	2	16
## 48	2021	2	17
## 49	2021	2	18
## 50	2021	2	19
## 51	2021	2	20
## 52	2021	2	21
## 53	2021	2	22

## 54	2021	2	23
## 55	2021	2	24
## 56	2021	2	25
## 57	2021	2	26
## 58	2021	2	27
## 59	2021	2	28
## 60	2021	3	1
## 61	2021	3	2
## 62	2021	3	3
## 63	2021	3	4
## 64	2021	3	5
## 65	2021	3	6
## 66	2021	3	7
## 67	2021	3	8
## 68	2021	3	9
## 69	2021	3	10
## 70	2021	3	11
## 71	2021	3	12
## 72	2021	3	13
## 73	2021	3	14
## 74	2021	3	15
## 75	2021	3	16
## 76	2021	3	17
## 77	2021	3	18
## 78	2021	3	19
## 79	2021	3	20
## 80	2021	3	21
## 81	2021	3	22
## 82	2021	3	23
## 83	2021	3	24
## 84	2021	3	25
## 85	2021	3	26
## 86	2021	3	27
## 87	2021	3	28
## 88	2021	3	29
## 89	2021	3	30
## 90	2021	3	31
## 91	2021	4	1
## 92	2021	4	2
## 93	2021	4	3
## 94	2021	4	4
## 95	2021	4	5
## 96	2021	4	6
## 97	2021	4	7
## 98	2021	4	8
## 99	2021	4	9
## 100	2021	4	10
## 101	2021	4	11
## 102	2021	4	12
## 103	2021	4	13
## 104	2021	4	14
## 105	2021	4	15
## 106	2021	4	16
## 107	2021	4	17

## 108 2021	4	18
## 109 2021	4	19
## 110 2021	4	20
## 111 2021	4	21
## 112 2021	4	22
## 113 2021	4	23
## 114 2021	4	24
## 115 2021	4	25
## 116 2021	4	26
## 117 2021	4	27
## 118 2021	4	28
## 119 2021	4	29
## 120 2021	4	30
## 121 2021	5	1
## 122 2021	5	2
## 123 2021	5	3
## 124 2021	5	4
## 125 2021	5	5
## 126 2021	5	6
## 127 2021	5	7
## 128 2021	5	8
## 129 2021	5	9
## 130 2021	5	10
## 131 2021	5	11
## 132 2021	5	12
## 133 2021	5	13
## 134 2021	5	14
## 135 2021	5	15
## 136 2021	5	16
## 137 2021	5	17
## 138 2021	5	18
## 139 2021	5	19
## 140 2021	5	20
## 141 2021	5	21
## 142 2021	5	22
## 143 2021	5	23
## 144 2021	5	24
## 145 2021	5	25
## 146 2021	5	26
## 147 2021	5	27
## 148 2021	5	28
## 149 2021	5	29
## 150 2021	5	30
## 151 2021	5	31
## 152 2021	6	1
## 153 2021	6	2
## 154 2021	6	3
## 155 2021	6	4
## 156 2021	6	5
## 157 2021	6	6
## 158 2021	6	7
## 159 2021	6	8
## 160 2021	6	9
## 161 2021	6	10

## 162 2021	6	11
## 163 2021	6	12
## 164 2021	6	13
## 165 2021	6	14
## 166 2021	6	15
## 167 2021	6	16
## 168 2021	6	17
## 169 2021	6	18
## 170 2021	6	19
## 171 2021	6	20
## 172 2021	6	21
## 173 2021	6	22
## 174 2021	6	23
## 175 2021	6	24
## 176 2021	6	25
## 177 2021	6	26
## 178 2021	6	27
## 179 2021	6	28
## 180 2021	6	29
## 181 2021	6	30
## 182 2021	7	1
## 183 2021	7	2
## 184 2021	7	3
## 185 2021	7	4
## 186 2021	7	5
## 187 2021	7	6
## 188 2021	7	7
## 189 2021	7	8
## 190 2021	7	9
## 191 2021	7	10
## 192 2021	7	11
## 193 2021	7	12
## 194 2021	7	13
## 195 2021	7	14
## 196 2021	7	15
## 197 2021	7	16
## 198 2021	7	17
## 199 2021	7	18
## 200 2021	7	19
## 201 2021	7	20
## 202 2021	7	21
## 203 2021	7	22
## 204 2021	7	23
## 205 2021	7	24
## 206 2021	7	25
## 207 2021	7	26
## 208 2021	7	27
## 209 2021	7	28
## 210 2021	7	29
## 211 2021	7	30
## 212 2021	7	31
## 213 2021	8	1
## 214 2021	8	2
## 215 2021	8	3

##	216	2021	8	4
##	217	2021	8	5
##	218	2021	8	6
##	219	2021	8	7
##	220	2021	8	8
##	221	2021	8	9
##	222	2021	8	10
##	223	2021	8	11
##	224	2021	8	12
##	225	2021	8	13
##	226	2021	8	14
##	227	2021	8	15
##	228	2021	8	16
##	229	2021	8	17
##	230	2021	8	18
##	231	2021	8	19
##	232	2021	8	20
##	233	2021	8	21
##	234	2021	8	22
##	235	2021	8	23
##	236	2021	8	24
##	237	2021	8	25
##	238	2021	8	26
##	239	2021	8	27
##	240	2021	8	28
##	241	2021	8	29
##	242	2021	8	30
##	243	2021	8	31
##	244	2021	9	1
##	245	2021	9	2
##	246	2021	9	3
##	247	2021	9	4
##	248	2021	9	5
##	249	2021	9	6
##	250	2021	9	7
##	251	2021	9	8
##	252	2021	9	9
##	253	2021	9	10
##	254	2021	9	11
##	255	2021	9	12
##	256	2021	9	13
##	257	2021	9	14
##	258	2021	9	15
##	259	2021	9	16
##	260	2021	9	17
##	261	2021	9	18
##	262	2021	9	19
##	263	2021	9	20
##	264	2021	9	21
##	265	2021	9	22
##	266	2021	9	23
##	267	2021	9	24
##	268	2021	9	25
##	269	2021	9	26

##	270	2021	9	27
##	271	2021	9	28
##	272	2021	9	29
##	273	2021	9	30
##	274	2021	10	1
##	275	2021	10	2
##	276	2021	10	3
##	277	2021	10	4
##	278	2021	10	5
##	279	2021	10	6
##	280	2021	10	7
##	281	2021	10	8
##	282	2021	10	9
##	283	2021	10	10
##	284	2021	10	11
##	285	2021	10	12
##	286	2021	10	13
##	287	2021	10	14
##	288	2021	10	15
##	289	2021	10	16
##	290	2021	10	17
##	291	2021	10	18
##	292	2021	10	19
##	293	2021	10	20
##	294	2021	10	21
##	295	2021	10	22
##	296	2021	10	23
##	297	2021	10	24
##	298	2021	10	25
##	299	2021	10	26
##	300	2021	10	27
##	301	2021	10	28
##	302	2021	10	29
##	303	2021	10	30
##	304	2021	10	31
##	305	2021	11	1
##	306	2021	11	2
##	307	2021	11	3
##	308	2021	11	4
##	309	2021	11	5
##	310	2021	11	6
##	311	2021	11	7
##	312	2021	11	8
##	313	2021	11	9
##	314	2021	11	10
##	315	2021	11	11
##	316	2021	11	12
##	317	2021	11	13
##	318	2021	11	14
##	319	2021	11	15
##	320	2021	11	16
##	321	2021	11	17
##	322	2021	11	18
##	323	2021	11	19

```
## 324 2021    11      20
## 325 2021    11      21
## 326 2021    11      22
## 327 2021    11      23
## 328 2021    11      24
## 329 2021    11      25
## 330 2021    11      26
## 331 2021    11      27
## 332 2021    11      28
## 333 2021    11      29
## 334 2021    11      30
## 335 2021    12       1
## 336 2021    12       2
## 337 2021    12       3
## 338 2021    12       4
## 339 2021    12       5
## 340 2021    12       6
## 341 2021    12       7
## 342 2021    12       8
## 343 2021    12       9
## 344 2021    12      10
## 345 2021    12      11
## 346 2021    12      12
## 347 2021    12      13
## 348 2021    12      14
## 349 2021    12      15
## 350 2021    12      16
## 351 2021    12      17
## 352 2021    12      18
## 353 2021    12      19
## 354 2021    12      20
## 355 2021    12      21
## 356 2021    12      22
## 357 2021    12      23
## 358 2021    12      24
## 359 2021    12      25
## 360 2021    12      26
## 361 2021    12      27
## 362 2021    12      28
## 363 2021    12      29
## 364 2021    12      30
## 365 2021    12      31
```

```
table.sep_ <- table %>%
  separate(data = .,
           col = date,
           into = c("year", "month", "dayofmonth"),
           sep = "-") %>%
  mutate_at(.tbl = .,                                # which table? - . stands for table in the pipe line!
            .vars = c("month", "dayofmonth"),         # which variables are mutated?
            .funs = as.numeric) %>%                 # which functions is applied?
  arrange(year, month, dayofmonth)
table.sep_
```

```
##      year month dayofmonth
```

## 1	2021	1	1
## 2	2021	1	2
## 3	2021	1	3
## 4	2021	1	4
## 5	2021	1	5
## 6	2021	1	6
## 7	2021	1	7
## 8	2021	1	8
## 9	2021	1	9
## 10	2021	1	10
## 11	2021	1	11
## 12	2021	1	12
## 13	2021	1	13
## 14	2021	1	14
## 15	2021	1	15
## 16	2021	1	16
## 17	2021	1	17
## 18	2021	1	18
## 19	2021	1	19
## 20	2021	1	20
## 21	2021	1	21
## 22	2021	1	22
## 23	2021	1	23
## 24	2021	1	24
## 25	2021	1	25
## 26	2021	1	26
## 27	2021	1	27
## 28	2021	1	28
## 29	2021	1	29
## 30	2021	1	30
## 31	2021	1	31
## 32	2021	2	1
## 33	2021	2	2
## 34	2021	2	3
## 35	2021	2	4
## 36	2021	2	5
## 37	2021	2	6
## 38	2021	2	7
## 39	2021	2	8
## 40	2021	2	9
## 41	2021	2	10
## 42	2021	2	11
## 43	2021	2	12
## 44	2021	2	13
## 45	2021	2	14
## 46	2021	2	15
## 47	2021	2	16
## 48	2021	2	17
## 49	2021	2	18
## 50	2021	2	19
## 51	2021	2	20
## 52	2021	2	21
## 53	2021	2	22
## 54	2021	2	23

## 55	2021	2	24
## 56	2021	2	25
## 57	2021	2	26
## 58	2021	2	27
## 59	2021	2	28
## 60	2021	3	1
## 61	2021	3	2
## 62	2021	3	3
## 63	2021	3	4
## 64	2021	3	5
## 65	2021	3	6
## 66	2021	3	7
## 67	2021	3	8
## 68	2021	3	9
## 69	2021	3	10
## 70	2021	3	11
## 71	2021	3	12
## 72	2021	3	13
## 73	2021	3	14
## 74	2021	3	15
## 75	2021	3	16
## 76	2021	3	17
## 77	2021	3	18
## 78	2021	3	19
## 79	2021	3	20
## 80	2021	3	21
## 81	2021	3	22
## 82	2021	3	23
## 83	2021	3	24
## 84	2021	3	25
## 85	2021	3	26
## 86	2021	3	27
## 87	2021	3	28
## 88	2021	3	29
## 89	2021	3	30
## 90	2021	3	31
## 91	2021	4	1
## 92	2021	4	2
## 93	2021	4	3
## 94	2021	4	4
## 95	2021	4	5
## 96	2021	4	6
## 97	2021	4	7
## 98	2021	4	8
## 99	2021	4	9
## 100	2021	4	10
## 101	2021	4	11
## 102	2021	4	12
## 103	2021	4	13
## 104	2021	4	14
## 105	2021	4	15
## 106	2021	4	16
## 107	2021	4	17
## 108	2021	4	18

## 109 2021	4	19
## 110 2021	4	20
## 111 2021	4	21
## 112 2021	4	22
## 113 2021	4	23
## 114 2021	4	24
## 115 2021	4	25
## 116 2021	4	26
## 117 2021	4	27
## 118 2021	4	28
## 119 2021	4	29
## 120 2021	4	30
## 121 2021	5	1
## 122 2021	5	2
## 123 2021	5	3
## 124 2021	5	4
## 125 2021	5	5
## 126 2021	5	6
## 127 2021	5	7
## 128 2021	5	8
## 129 2021	5	9
## 130 2021	5	10
## 131 2021	5	11
## 132 2021	5	12
## 133 2021	5	13
## 134 2021	5	14
## 135 2021	5	15
## 136 2021	5	16
## 137 2021	5	17
## 138 2021	5	18
## 139 2021	5	19
## 140 2021	5	20
## 141 2021	5	21
## 142 2021	5	22
## 143 2021	5	23
## 144 2021	5	24
## 145 2021	5	25
## 146 2021	5	26
## 147 2021	5	27
## 148 2021	5	28
## 149 2021	5	29
## 150 2021	5	30
## 151 2021	5	31
## 152 2021	6	1
## 153 2021	6	2
## 154 2021	6	3
## 155 2021	6	4
## 156 2021	6	5
## 157 2021	6	6
## 158 2021	6	7
## 159 2021	6	8
## 160 2021	6	9
## 161 2021	6	10
## 162 2021	6	11

## 163 2021	6	12
## 164 2021	6	13
## 165 2021	6	14
## 166 2021	6	15
## 167 2021	6	16
## 168 2021	6	17
## 169 2021	6	18
## 170 2021	6	19
## 171 2021	6	20
## 172 2021	6	21
## 173 2021	6	22
## 174 2021	6	23
## 175 2021	6	24
## 176 2021	6	25
## 177 2021	6	26
## 178 2021	6	27
## 179 2021	6	28
## 180 2021	6	29
## 181 2021	6	30
## 182 2021	7	1
## 183 2021	7	2
## 184 2021	7	3
## 185 2021	7	4
## 186 2021	7	5
## 187 2021	7	6
## 188 2021	7	7
## 189 2021	7	8
## 190 2021	7	9
## 191 2021	7	10
## 192 2021	7	11
## 193 2021	7	12
## 194 2021	7	13
## 195 2021	7	14
## 196 2021	7	15
## 197 2021	7	16
## 198 2021	7	17
## 199 2021	7	18
## 200 2021	7	19
## 201 2021	7	20
## 202 2021	7	21
## 203 2021	7	22
## 204 2021	7	23
## 205 2021	7	24
## 206 2021	7	25
## 207 2021	7	26
## 208 2021	7	27
## 209 2021	7	28
## 210 2021	7	29
## 211 2021	7	30
## 212 2021	7	31
## 213 2021	8	1
## 214 2021	8	2
## 215 2021	8	3
## 216 2021	8	4

##	217	2021	8	5
##	218	2021	8	6
##	219	2021	8	7
##	220	2021	8	8
##	221	2021	8	9
##	222	2021	8	10
##	223	2021	8	11
##	224	2021	8	12
##	225	2021	8	13
##	226	2021	8	14
##	227	2021	8	15
##	228	2021	8	16
##	229	2021	8	17
##	230	2021	8	18
##	231	2021	8	19
##	232	2021	8	20
##	233	2021	8	21
##	234	2021	8	22
##	235	2021	8	23
##	236	2021	8	24
##	237	2021	8	25
##	238	2021	8	26
##	239	2021	8	27
##	240	2021	8	28
##	241	2021	8	29
##	242	2021	8	30
##	243	2021	8	31
##	244	2021	9	1
##	245	2021	9	2
##	246	2021	9	3
##	247	2021	9	4
##	248	2021	9	5
##	249	2021	9	6
##	250	2021	9	7
##	251	2021	9	8
##	252	2021	9	9
##	253	2021	9	10
##	254	2021	9	11
##	255	2021	9	12
##	256	2021	9	13
##	257	2021	9	14
##	258	2021	9	15
##	259	2021	9	16
##	260	2021	9	17
##	261	2021	9	18
##	262	2021	9	19
##	263	2021	9	20
##	264	2021	9	21
##	265	2021	9	22
##	266	2021	9	23
##	267	2021	9	24
##	268	2021	9	25
##	269	2021	9	26
##	270	2021	9	27

##	271	2021	9	28
##	272	2021	9	29
##	273	2021	9	30
##	274	2021	10	1
##	275	2021	10	2
##	276	2021	10	3
##	277	2021	10	4
##	278	2021	10	5
##	279	2021	10	6
##	280	2021	10	7
##	281	2021	10	8
##	282	2021	10	9
##	283	2021	10	10
##	284	2021	10	11
##	285	2021	10	12
##	286	2021	10	13
##	287	2021	10	14
##	288	2021	10	15
##	289	2021	10	16
##	290	2021	10	17
##	291	2021	10	18
##	292	2021	10	19
##	293	2021	10	20
##	294	2021	10	21
##	295	2021	10	22
##	296	2021	10	23
##	297	2021	10	24
##	298	2021	10	25
##	299	2021	10	26
##	300	2021	10	27
##	301	2021	10	28
##	302	2021	10	29
##	303	2021	10	30
##	304	2021	10	31
##	305	2021	11	1
##	306	2021	11	2
##	307	2021	11	3
##	308	2021	11	4
##	309	2021	11	5
##	310	2021	11	6
##	311	2021	11	7
##	312	2021	11	8
##	313	2021	11	9
##	314	2021	11	10
##	315	2021	11	11
##	316	2021	11	12
##	317	2021	11	13
##	318	2021	11	14
##	319	2021	11	15
##	320	2021	11	16
##	321	2021	11	17
##	322	2021	11	18
##	323	2021	11	19
##	324	2021	11	20

```
## 325 2021    11      21
## 326 2021    11      22
## 327 2021    11      23
## 328 2021    11      24
## 329 2021    11      25
## 330 2021    11      26
## 331 2021    11      27
## 332 2021    11      28
## 333 2021    11      29
## 334 2021    11      30
## 335 2021    12       1
## 336 2021    12       2
## 337 2021    12       3
## 338 2021    12       4
## 339 2021    12       5
## 340 2021    12       6
## 341 2021    12       7
## 342 2021    12       8
## 343 2021    12       9
## 344 2021    12      10
## 345 2021    12      11
## 346 2021    12      12
## 347 2021    12      13
## 348 2021    12      14
## 349 2021    12      15
## 350 2021    12      16
## 351 2021    12      17
## 352 2021    12      18
## 353 2021    12      19
## 354 2021    12      20
## 355 2021    12      21
## 356 2021    12      22
## 357 2021    12      23
## 358 2021    12      24
## 359 2021    12      25
## 360 2021    12      26
## 361 2021    12      27
## 362 2021    12      28
## 363 2021    12      29
## 364 2021    12      30
## 365 2021    12      31
```

`unite()`

```
table.unite <- table.sep %>%
  # add leading zeros
  mutate(month = str_pad(month, width = 2, side = "left", pad = "0"), # add leading zeros to month
         dayofmonth = str_pad(dayofmonth, width = 2, side = "left", pad = "0")) %>% # add leading zeros
  unite(data = .,
        col = "date",
        year, month, dayofmonth,
        sep = "-") %>%
  arrange(date)
table.unite
```

##	date
## 1	2021-01-01
## 2	2021-01-02
## 3	2021-01-03
## 4	2021-01-04
## 5	2021-01-05
## 6	2021-01-06
## 7	2021-01-07
## 8	2021-01-08
## 9	2021-01-09
## 10	2021-01-10
## 11	2021-01-11
## 12	2021-01-12
## 13	2021-01-13
## 14	2021-01-14
## 15	2021-01-15
## 16	2021-01-16
## 17	2021-01-17
## 18	2021-01-18
## 19	2021-01-19
## 20	2021-01-20
## 21	2021-01-21
## 22	2021-01-22
## 23	2021-01-23
## 24	2021-01-24
## 25	2021-01-25
## 26	2021-01-26
## 27	2021-01-27
## 28	2021-01-28
## 29	2021-01-29
## 30	2021-01-30
## 31	2021-01-31
## 32	2021-02-01
## 33	2021-02-02
## 34	2021-02-03
## 35	2021-02-04
## 36	2021-02-05
## 37	2021-02-06
## 38	2021-02-07
## 39	2021-02-08
## 40	2021-02-09
## 41	2021-02-10
## 42	2021-02-11
## 43	2021-02-12
## 44	2021-02-13
## 45	2021-02-14
## 46	2021-02-15
## 47	2021-02-16
## 48	2021-02-17
## 49	2021-02-18
## 50	2021-02-19
## 51	2021-02-20
## 52	2021-02-21
## 53	2021-02-22

54 2021-02-23
55 2021-02-24
56 2021-02-25
57 2021-02-26
58 2021-02-27
59 2021-02-28
60 2021-03-01
61 2021-03-02
62 2021-03-03
63 2021-03-04
64 2021-03-05
65 2021-03-06
66 2021-03-07
67 2021-03-08
68 2021-03-09
69 2021-03-10
70 2021-03-11
71 2021-03-12
72 2021-03-13
73 2021-03-14
74 2021-03-15
75 2021-03-16
76 2021-03-17
77 2021-03-18
78 2021-03-19
79 2021-03-20
80 2021-03-21
81 2021-03-22
82 2021-03-23
83 2021-03-24
84 2021-03-25
85 2021-03-26
86 2021-03-27
87 2021-03-28
88 2021-03-29
89 2021-03-30
90 2021-03-31
91 2021-04-01
92 2021-04-02
93 2021-04-03
94 2021-04-04
95 2021-04-05
96 2021-04-06
97 2021-04-07
98 2021-04-08
99 2021-04-09
100 2021-04-10
101 2021-04-11
102 2021-04-12
103 2021-04-13
104 2021-04-14
105 2021-04-15
106 2021-04-16
107 2021-04-17

108 2021-04-18
109 2021-04-19
110 2021-04-20
111 2021-04-21
112 2021-04-22
113 2021-04-23
114 2021-04-24
115 2021-04-25
116 2021-04-26
117 2021-04-27
118 2021-04-28
119 2021-04-29
120 2021-04-30
121 2021-05-01
122 2021-05-02
123 2021-05-03
124 2021-05-04
125 2021-05-05
126 2021-05-06
127 2021-05-07
128 2021-05-08
129 2021-05-09
130 2021-05-10
131 2021-05-11
132 2021-05-12
133 2021-05-13
134 2021-05-14
135 2021-05-15
136 2021-05-16
137 2021-05-17
138 2021-05-18
139 2021-05-19
140 2021-05-20
141 2021-05-21
142 2021-05-22
143 2021-05-23
144 2021-05-24
145 2021-05-25
146 2021-05-26
147 2021-05-27
148 2021-05-28
149 2021-05-29
150 2021-05-30
151 2021-05-31
152 2021-06-01
153 2021-06-02
154 2021-06-03
155 2021-06-04
156 2021-06-05
157 2021-06-06
158 2021-06-07
159 2021-06-08
160 2021-06-09
161 2021-06-10

162 2021-06-11
163 2021-06-12
164 2021-06-13
165 2021-06-14
166 2021-06-15
167 2021-06-16
168 2021-06-17
169 2021-06-18
170 2021-06-19
171 2021-06-20
172 2021-06-21
173 2021-06-22
174 2021-06-23
175 2021-06-24
176 2021-06-25
177 2021-06-26
178 2021-06-27
179 2021-06-28
180 2021-06-29
181 2021-06-30
182 2021-07-01
183 2021-07-02
184 2021-07-03
185 2021-07-04
186 2021-07-05
187 2021-07-06
188 2021-07-07
189 2021-07-08
190 2021-07-09
191 2021-07-10
192 2021-07-11
193 2021-07-12
194 2021-07-13
195 2021-07-14
196 2021-07-15
197 2021-07-16
198 2021-07-17
199 2021-07-18
200 2021-07-19
201 2021-07-20
202 2021-07-21
203 2021-07-22
204 2021-07-23
205 2021-07-24
206 2021-07-25
207 2021-07-26
208 2021-07-27
209 2021-07-28
210 2021-07-29
211 2021-07-30
212 2021-07-31
213 2021-08-01
214 2021-08-02
215 2021-08-03

216 2021-08-04
217 2021-08-05
218 2021-08-06
219 2021-08-07
220 2021-08-08
221 2021-08-09
222 2021-08-10
223 2021-08-11
224 2021-08-12
225 2021-08-13
226 2021-08-14
227 2021-08-15
228 2021-08-16
229 2021-08-17
230 2021-08-18
231 2021-08-19
232 2021-08-20
233 2021-08-21
234 2021-08-22
235 2021-08-23
236 2021-08-24
237 2021-08-25
238 2021-08-26
239 2021-08-27
240 2021-08-28
241 2021-08-29
242 2021-08-30
243 2021-08-31
244 2021-09-01
245 2021-09-02
246 2021-09-03
247 2021-09-04
248 2021-09-05
249 2021-09-06
250 2021-09-07
251 2021-09-08
252 2021-09-09
253 2021-09-10
254 2021-09-11
255 2021-09-12
256 2021-09-13
257 2021-09-14
258 2021-09-15
259 2021-09-16
260 2021-09-17
261 2021-09-18
262 2021-09-19
263 2021-09-20
264 2021-09-21
265 2021-09-22
266 2021-09-23
267 2021-09-24
268 2021-09-25
269 2021-09-26

270 2021-09-27
271 2021-09-28
272 2021-09-29
273 2021-09-30
274 2021-10-01
275 2021-10-02
276 2021-10-03
277 2021-10-04
278 2021-10-05
279 2021-10-06
280 2021-10-07
281 2021-10-08
282 2021-10-09
283 2021-10-10
284 2021-10-11
285 2021-10-12
286 2021-10-13
287 2021-10-14
288 2021-10-15
289 2021-10-16
290 2021-10-17
291 2021-10-18
292 2021-10-19
293 2021-10-20
294 2021-10-21
295 2021-10-22
296 2021-10-23
297 2021-10-24
298 2021-10-25
299 2021-10-26
300 2021-10-27
301 2021-10-28
302 2021-10-29
303 2021-10-30
304 2021-10-31
305 2021-11-01
306 2021-11-02
307 2021-11-03
308 2021-11-04
309 2021-11-05
310 2021-11-06
311 2021-11-07
312 2021-11-08
313 2021-11-09
314 2021-11-10
315 2021-11-11
316 2021-11-12
317 2021-11-13
318 2021-11-14
319 2021-11-15
320 2021-11-16
321 2021-11-17
322 2021-11-18
323 2021-11-19

```
## 324 2021-11-20
## 325 2021-11-21
## 326 2021-11-22
## 327 2021-11-23
## 328 2021-11-24
## 329 2021-11-25
## 330 2021-11-26
## 331 2021-11-27
## 332 2021-11-28
## 333 2021-11-29
## 334 2021-11-30
## 335 2021-12-01
## 336 2021-12-02
## 337 2021-12-03
## 338 2021-12-04
## 339 2021-12-05
## 340 2021-12-06
## 341 2021-12-07
## 342 2021-12-08
## 343 2021-12-09
## 344 2021-12-10
## 345 2021-12-11
## 346 2021-12-12
## 347 2021-12-13
## 348 2021-12-14
## 349 2021-12-15
## 350 2021-12-16
## 351 2021-12-17
## 352 2021-12-18
## 353 2021-12-19
## 354 2021-12-20
## 355 2021-12-21
## 356 2021-12-22
## 357 2021-12-23
## 358 2021-12-24
## 359 2021-12-25
## 360 2021-12-26
## 361 2021-12-27
## 362 2021-12-28
## 363 2021-12-29
## 364 2021-12-30
## 365 2021-12-31
```

```
table.unite_ <- table.sep %>%
  # add leading zeros
  mutate_at(.tbl = .,
    .vars = c("month", "dayofmonth"),
    .funs = str_pad, 2, "left", "0") %>%
  unite(data = .,
    col = "date",
    year, month, dayofmonth,
    sep = "-") %>%
  arrange(date)
table.unite_
```

which table? - . stands for table in the pipe line
which variables are mutated?
which functions is applied? - function parameters

##	date
## 1	2021-01-01
## 2	2021-01-02
## 3	2021-01-03
## 4	2021-01-04
## 5	2021-01-05
## 6	2021-01-06
## 7	2021-01-07
## 8	2021-01-08
## 9	2021-01-09
## 10	2021-01-10
## 11	2021-01-11
## 12	2021-01-12
## 13	2021-01-13
## 14	2021-01-14
## 15	2021-01-15
## 16	2021-01-16
## 17	2021-01-17
## 18	2021-01-18
## 19	2021-01-19
## 20	2021-01-20
## 21	2021-01-21
## 22	2021-01-22
## 23	2021-01-23
## 24	2021-01-24
## 25	2021-01-25
## 26	2021-01-26
## 27	2021-01-27
## 28	2021-01-28
## 29	2021-01-29
## 30	2021-01-30
## 31	2021-01-31
## 32	2021-02-01
## 33	2021-02-02
## 34	2021-02-03
## 35	2021-02-04
## 36	2021-02-05
## 37	2021-02-06
## 38	2021-02-07
## 39	2021-02-08
## 40	2021-02-09
## 41	2021-02-10
## 42	2021-02-11
## 43	2021-02-12
## 44	2021-02-13
## 45	2021-02-14
## 46	2021-02-15
## 47	2021-02-16
## 48	2021-02-17
## 49	2021-02-18
## 50	2021-02-19
## 51	2021-02-20
## 52	2021-02-21
## 53	2021-02-22

54 2021-02-23
55 2021-02-24
56 2021-02-25
57 2021-02-26
58 2021-02-27
59 2021-02-28
60 2021-03-01
61 2021-03-02
62 2021-03-03
63 2021-03-04
64 2021-03-05
65 2021-03-06
66 2021-03-07
67 2021-03-08
68 2021-03-09
69 2021-03-10
70 2021-03-11
71 2021-03-12
72 2021-03-13
73 2021-03-14
74 2021-03-15
75 2021-03-16
76 2021-03-17
77 2021-03-18
78 2021-03-19
79 2021-03-20
80 2021-03-21
81 2021-03-22
82 2021-03-23
83 2021-03-24
84 2021-03-25
85 2021-03-26
86 2021-03-27
87 2021-03-28
88 2021-03-29
89 2021-03-30
90 2021-03-31
91 2021-04-01
92 2021-04-02
93 2021-04-03
94 2021-04-04
95 2021-04-05
96 2021-04-06
97 2021-04-07
98 2021-04-08
99 2021-04-09
100 2021-04-10
101 2021-04-11
102 2021-04-12
103 2021-04-13
104 2021-04-14
105 2021-04-15
106 2021-04-16
107 2021-04-17

108 2021-04-18
109 2021-04-19
110 2021-04-20
111 2021-04-21
112 2021-04-22
113 2021-04-23
114 2021-04-24
115 2021-04-25
116 2021-04-26
117 2021-04-27
118 2021-04-28
119 2021-04-29
120 2021-04-30
121 2021-05-01
122 2021-05-02
123 2021-05-03
124 2021-05-04
125 2021-05-05
126 2021-05-06
127 2021-05-07
128 2021-05-08
129 2021-05-09
130 2021-05-10
131 2021-05-11
132 2021-05-12
133 2021-05-13
134 2021-05-14
135 2021-05-15
136 2021-05-16
137 2021-05-17
138 2021-05-18
139 2021-05-19
140 2021-05-20
141 2021-05-21
142 2021-05-22
143 2021-05-23
144 2021-05-24
145 2021-05-25
146 2021-05-26
147 2021-05-27
148 2021-05-28
149 2021-05-29
150 2021-05-30
151 2021-05-31
152 2021-06-01
153 2021-06-02
154 2021-06-03
155 2021-06-04
156 2021-06-05
157 2021-06-06
158 2021-06-07
159 2021-06-08
160 2021-06-09
161 2021-06-10

162 2021-06-11
163 2021-06-12
164 2021-06-13
165 2021-06-14
166 2021-06-15
167 2021-06-16
168 2021-06-17
169 2021-06-18
170 2021-06-19
171 2021-06-20
172 2021-06-21
173 2021-06-22
174 2021-06-23
175 2021-06-24
176 2021-06-25
177 2021-06-26
178 2021-06-27
179 2021-06-28
180 2021-06-29
181 2021-06-30
182 2021-07-01
183 2021-07-02
184 2021-07-03
185 2021-07-04
186 2021-07-05
187 2021-07-06
188 2021-07-07
189 2021-07-08
190 2021-07-09
191 2021-07-10
192 2021-07-11
193 2021-07-12
194 2021-07-13
195 2021-07-14
196 2021-07-15
197 2021-07-16
198 2021-07-17
199 2021-07-18
200 2021-07-19
201 2021-07-20
202 2021-07-21
203 2021-07-22
204 2021-07-23
205 2021-07-24
206 2021-07-25
207 2021-07-26
208 2021-07-27
209 2021-07-28
210 2021-07-29
211 2021-07-30
212 2021-07-31
213 2021-08-01
214 2021-08-02
215 2021-08-03

216 2021-08-04
217 2021-08-05
218 2021-08-06
219 2021-08-07
220 2021-08-08
221 2021-08-09
222 2021-08-10
223 2021-08-11
224 2021-08-12
225 2021-08-13
226 2021-08-14
227 2021-08-15
228 2021-08-16
229 2021-08-17
230 2021-08-18
231 2021-08-19
232 2021-08-20
233 2021-08-21
234 2021-08-22
235 2021-08-23
236 2021-08-24
237 2021-08-25
238 2021-08-26
239 2021-08-27
240 2021-08-28
241 2021-08-29
242 2021-08-30
243 2021-08-31
244 2021-09-01
245 2021-09-02
246 2021-09-03
247 2021-09-04
248 2021-09-05
249 2021-09-06
250 2021-09-07
251 2021-09-08
252 2021-09-09
253 2021-09-10
254 2021-09-11
255 2021-09-12
256 2021-09-13
257 2021-09-14
258 2021-09-15
259 2021-09-16
260 2021-09-17
261 2021-09-18
262 2021-09-19
263 2021-09-20
264 2021-09-21
265 2021-09-22
266 2021-09-23
267 2021-09-24
268 2021-09-25
269 2021-09-26

270 2021-09-27
271 2021-09-28
272 2021-09-29
273 2021-09-30
274 2021-10-01
275 2021-10-02
276 2021-10-03
277 2021-10-04
278 2021-10-05
279 2021-10-06
280 2021-10-07
281 2021-10-08
282 2021-10-09
283 2021-10-10
284 2021-10-11
285 2021-10-12
286 2021-10-13
287 2021-10-14
288 2021-10-15
289 2021-10-16
290 2021-10-17
291 2021-10-18
292 2021-10-19
293 2021-10-20
294 2021-10-21
295 2021-10-22
296 2021-10-23
297 2021-10-24
298 2021-10-25
299 2021-10-26
300 2021-10-27
301 2021-10-28
302 2021-10-29
303 2021-10-30
304 2021-10-31
305 2021-11-01
306 2021-11-02
307 2021-11-03
308 2021-11-04
309 2021-11-05
310 2021-11-06
311 2021-11-07
312 2021-11-08
313 2021-11-09
314 2021-11-10
315 2021-11-11
316 2021-11-12
317 2021-11-13
318 2021-11-14
319 2021-11-15
320 2021-11-16
321 2021-11-17
322 2021-11-18
323 2021-11-19

324 2021-11-20
325 2021-11-21
326 2021-11-22
327 2021-11-23
328 2021-11-24
329 2021-11-25
330 2021-11-26
331 2021-11-27
332 2021-11-28
333 2021-11-29
334 2021-11-30
335 2021-12-01
336 2021-12-02
337 2021-12-03
338 2021-12-04
339 2021-12-05
340 2021-12-06
341 2021-12-07
342 2021-12-08
343 2021-12-09
344 2021-12-10
345 2021-12-11
346 2021-12-12
347 2021-12-13
348 2021-12-14
349 2021-12-15
350 2021-12-16
351 2021-12-17
352 2021-12-18
353 2021-12-19
354 2021-12-20
355 2021-12-21
356 2021-12-22
357 2021-12-23
358 2021-12-24
359 2021-12-25
360 2021-12-26
361 2021-12-27
362 2021-12-28
363 2021-12-29
364 2021-12-30
365 2021-12-31