

# Dates in R

# Date Class

- Dates can be handled with Date class in R
- Any date is represented here as a number starting from 1<sup>st</sup> Jan 1970 with 1<sup>st</sup> Jan 1970 corresponding to zero
- For days earlier than 1<sup>st</sup> Jan 1970 negative numbers are used

# as.Date()

- Function as.Date() converts dates in character to dates in Date Class

```
> dt1 <- as.Date("1970-01-01")  
> class(dt1)  
[1] "Date"  
> unclass(dt1)  
[1] 0
```

```
> dt1 <- as.Date("2015-10-23")  
> unclass(dt1)  
[1] 16731
```

# Current Date

- Current system date can be retrieved with the function `Sys.Date()`

```
> Sys.Date()  
[1] "2016-01-12"
```

```
> class(Sys.Date())  
[1] "Date"
```

# Formatting Dates

Abbreviation	Specification
%d	Day as a number (01 - 31)
%a	Abbreviated weekday (Mon, Tue)
%A	Unabbreviated weekday (Monday, Tuesday, Wednesday )
%w	Weekday (0-6) 0-Sunday, 1-Monday
%W	Week (00-53) with Monday as first day of the week
%m	Month (01 – 12)
%b	Abbreviated month (Jan, Feb)
%B	Unabbreviated month (January, February)
%y	2 digit year
%Y	4 digit year

# format()

- Formatting can be done to the date with `format()` function

*Syntax : `format(date , "format")`*

```
> dt2 <- as.Date("2016-02-12")
> format(dt2, "%d, %B %Y")
[1] "12, February 2016"
> format(dt2, "%d-%b-%Y")
[1] "12-Feb-2016"
> format(dt2, "%A, %d %B %Y")
[1] "Friday, 12 February 2016"
```

# strptime()

- For reading the date from character format to date format we use strptime() function

Syntax : strptime(character , “format”)

```
> dft <- "20, October 2015 13:30:43"  
> dfy <- strptime(dft,"%d, %B %Y %H:%M:%S")  
> dfy  
[1] "2015-10-20 13:30:43 IST"  
> class(dfy)  
[1] "POSIXlt" "POSIXt"
```

# Generating a Sequence of Dates

- The function `seq()` helps us to generate the sequence of dates

Syntax : `seq(from, length, by,...)`

where

`length` : Desired length of the sequence

`by` : character indicating "week", "day", "week",  
"month", "quarter" or "year"

```
> TransDate <- seq(from=as.Date("2017-02-25"), length=30, by="week")
```

```
> TransDate
```

```
[1] "2017-02-25" "2017-03-04" "2017-03-11" "2017-03-18" "2017-03-25"  
[6] "2017-04-01" "2017-04-08" "2017-04-15" "2017-04-22" "2017-04-29"  
[11] "2017-05-06" "2017-05-13" "2017-05-20" "2017-05-27" "2017-06-03"  
[16] "2017-06-10" "2017-06-17" "2017-06-24" "2017-07-01" "2017-07-08"  
[21] "2017-07-15" "2017-07-22" "2017-07-29" "2017-08-05" "2017-08-12"  
[26] "2017-08-19" "2017-08-26" "2017-09-02" "2017-09-09" "2017-09-16"
```



# Simple Arithmetic with Dates

- We can easily find the difference between the dates and find the next date by just subtracting and adding

```
> dt2 <- as.Date("2016-02-12")  
> dt3 <- as.Date("2016-01-03")  
>  
> dt2 - dt3  
Time difference of 40 days
```

```
> dt3 + 5  
[1] "2016-01-08"
```

# Times in R

- Time can be handled using the classes `POSIXct` and `POSIXlt`
- Class "`POSIXct`" represents the (signed) number of seconds since the beginning of 1970 (in the UTC time zone) as a numeric vector.
- Class "`POSIXlt`" is a named list of vectors representing seconds, minutes, hours, etc.

# POSIXct

- Date class object can be converted to POSIXct class object with `as.POSIXct()` function

```
> dt1 <- sys.time()
> dt_ct <- as.POSIXct(dt1)
> dt_ct
[1] "2016-01-12 15:45:21 IST"
> unclass(dt_ct)
[1] 1452593721
```

# POSIXlt

- Date class object can be converted to POSIXlt class object with as.POSIXlt() function

```
> dt1 <- sys.time()
> dt_lt <- as.POSIXlt(dt1)
> dt_lt
[1] "2016-01-12 16:00:33 IST"
> unclass(dt_lt)
$sec
[1] 33.14448

$min
[1] 0

$hour
[1] 16

$mday
[1] 12

$mon
[1] 0
```

```
> p <- names(unclass(dt_lt))
> p
[1] "sec"      "min"      "hour"     "mday"     "mon"      "year"     "yday"     "isdst"
[10] "zone"     "gmtoff"
> dt_lt$sec
[1] 10.42157
> dt_lt$hour
[1] 14
```

# Package lubridate in R

# Parsing of date-time data

---

- Fast and user friendly parsing of date-time data. Lubridate has multiple function which can transforms dates in character and numeric to dates in Date or POSIXct class.

- One can choose the function whose name models the order in which the year ('y'), month ('m') and day ('d') elements appear the string to be parsed: dmy, myd, ymd, ydm, dym, mdy, ymd\_hms)

- These functions recognize arbitrary non-digit separators as well as no separator

```
> dmy(150916)
[1] "2016-09-15"
> myd("03-2016-15")
[1] "2016-03-15"
> ymd("151210")
[1] "2015-12-10"
> ydm("16/01/05")
[1] "2016-05-01"
> dym("15-2015-01")
[1] "2015-01-15"
> mdy("04-26-2016")
[1] "2016-04-26"
> ymd_hms("2011-06-10 14:20:48")
[1] "2011-06-10 14:20:48 UTC"
```

# Parsing of date-time data cntd..

`parse_date_time()` Parse character and numeric date-time into POSIXct class

*Syntax: `parse_date_time(date, orders, tz = "UTC", truncated = 0, .....`*

```
> dt<- "20, Feb 2015 13:40:32"
> class(dt)
[1] "character"
> dt_prs <- parse_date_time(dt, "dmy_HMS")
> class(dt_prs)
[1] "POSIXct" "POSIXt"
> dt_prs
[1] "2015-02-20 13:40:32 UTC"
```

```
> dt_1<- 20022015
> class(dt)
[1] "character"
> dt_prs1 <- parse_date_time(dt_1, "dmy")
> class(dt_prs1)
[1] "POSIXct" "POSIXt"
> dt_prs1
[1] "2015-02-20 UTC"
> unclass(dt_prs1)
[1] 1424390400
attr(,"tzone")
[1] "UTC"
```

`as_date ()` converts dates in character to dates in Date Class. Also one can use function `make_datetime()`

```
> dt2 <- as_date("2015-09-26")
> dt2
[1] "2015-09-26"
> class(dt2)
[1] "Date"
```

```
> make_datetime(year=2014, month = 11, day = 18, hour = 09, min = 35, sec = 14)
[1] "2014-11-18 09:35:14 UTC"
```

# Extraction/set components of a date-time

The individual values a date-time can be extracted from an instant and set with the access or functions second, minute, hour, day, week, month, year etc

```
> dt_prs
[1] "2015-02-20 13:40:32 UTC"
> date(dt_prs)
[1] "2015-02-20"
> day(dt_prs)
[1] 20
> month(dt_prs, abbr = FALSE, label = FALSE)
[1] 2
> year(dt_prs)
[1] 2015
> hour(dt_prs)
[1] 13
> minute(dt_prs)
[1] 40
> second(dt_prs)
[1] 32
> wday(dt_prs)
[1] 6
> yday(dt_prs)
[1] 51
> mday(dt_prs)
[1] 20
> week(dt_prs)
[1] 8
> tz(dt_prs)
[1] "UTC"
> quarter(dt_prs)
```

```
> dt_st <- ymd_hms("2016-09-15 18:20:45")
> dt_st
[1] "2016-09-15 18:20:45 UTC"
> date(dt_st) <- as.Date("2016-09-14")
> dt_st
[1] "2016-09-14 18:20:45 UTC"
> day(dt_st) <- 12
> dt_st
[1] "2016-09-12 18:20:45 UTC"
> month(dt_st) <- 03
> dt_st
[1] "2016-03-12 18:20:45 UTC"
> year(dt_st) <- 2015
> dt_st
[1] "2015-03-12 18:20:45 UTC"
> hour(dt_st) <- 15
> dt_st
[1] "2015-03-12 15:20:45 UTC"
> minute(dt_st) <- 11
> dt_st
[1] "2015-03-12 15:11:45 UTC"
> second(dt_st) <- 08
> dt_st
[1] "2015-03-12 15:11:08 UTC"
```