Date and Time with base R

|  |  |  |
| --- | --- | --- |
| Clause/function | Operational purpose | Example |
| Sys.Date() | * current date, considering individual time zone |  |
| Sys.time() | * current date-time, considering individual time zone |  |
| as.Date | * parse string to date * it is beneficial to parse to date, as the latter has more features (e.g. one can do math with a date, but not with a string) | * note: this last output looks like a message, but it is actually a numeric value |
| as.POSIXct()  or  as. POSIXlt() | * parse string to date time * equivalent to as.Date, parsing to date time allows us to run calculation * unlike the date object it works at the unit of seconds, rather than days |  |
| format = | * often strings in raw data gare in a format that cannot be directly recognized by as.Date or as.POSIX\* * in these cases we have to use as set of special letters as placeholders (similar to sprintf()) to set the right format | **see ?strptime for a full list of placeholders**  **date placeholders:**   * 4 digit year: **”<yyyy>“**followed by **%Y** * 2 digit year: “<yy>” followed by %y * “<mm>” followed by %m * “<month abbreviation (e.g. Jan)>” followed by %b * “<month’s full name (e.g. January)>” followed by %B   + for both, month abbr. and full name one can change the language argument:     - see list of built-in languages with     - > data\_names\_langs()     - include: , locale = locale(“<language abbr.>”)   to code  “<dd>” followed by %d  **time placeholders**   * 0-23 hour: “<hh>” followed by %H * 1-12 hour: “<hh>” followed by %I; must be used with %p * am/pm indicator: “<am/pm>” followed by %p * “<mm>” followed by %M * integer seconds: “<ss>” followed by %S * real seconds: “<OS>” followed by %OS   **Parse to date**  > as.Date(<”String to be parsed”>, format = “<specification of the date format>”)     * note: * string and specification always have to match when it comes to separator etc. * date gets automatically parsed to R’s default formate   **Parse to datetime**  > as.POSIXlt/ct()(<”String to be parsed”>, format = “<specification of the datetime format>”) |
| as.character() | * convert date back to character vector | **format character in a specific way:**  **use same placeholders used for parsing and format()** |
|  |  |  |