Practical No. 7 Date: / / 201

Aim: Program to perform factor operations on T-shirt sizes.

Objectives:

- To study R Factors & its operations.
- Implement a program to perform operations on T-shirt sizes.

Theory:

R Factor

Factors are R objects which are used to store categorical data as levels. Categorical data contains limited number of different values representing categories. It can be nominal (no implied order) or ordinal (natural ordering).For example,
blood type: A, B, AB or O (nominal)
t-shirt size: S, M, L, XL, XXL, XXL (ordinal)

In R, factors are created from a vector using factor() function. Calling factor() function causes following things to happen:

- Given vector is scanned to get different levels & to sort these levels.
- Convert the given vector into integer values (if required) when it is displayed.

Creating Factor

Syntax

```
factor(data, ordered, levels, labels)
```

- data: input vector elements
- ordered: if TRUE, it is ordinal categorical data
- levels: if present, specify different levels other than default
- labels: if present, specify levels names other than default

Example

```
blood <- c("B", "AB", "O", "A", "O", "O", "A", "B")
blood
[1] "B", "AB", "O", "A", "O", "O", "A", "B"
blood_factor <- factor(blood)
blood_factor
[1] B, AB, O, A, O, O, A, B
Levels: A AB B O       #Levels are sorted alphabetically
str(blood_factor)
Factor w/ 4 levels "A", "AB", "B", "O": 3 2 4 1 4 4 1 3
```

Factor Levels

Changing Levels' Order

Example

```
blood <- c("B", "AB", "O", "A", "O", "O", "A", "B")
```

```
blood_factor2 <- factor(blood, levels = c("O", "A", "B", "AB"))
blood_factor2
[1] B, AB, O, A, O, O, A, B
Levels: O A B AB
str(blood_factor2)
Factor w/ 4 levels "O", "A", "B", "AB": 3 4 1 2 1 1 2 3
```

Renaming Levels: level() function

Example

```
blood <- c("B", "AB", "O", "A", "O", "O", "A", "B")
blood_factor <- factor(blood)
levels(blood_factor) = c("BT_A", "BT_AB", "BT_B", "BT_O")
blood_factor
[1] BT_B, BT_AB, BT_O, BT_A, BT_O, BT_O, BT_A, BT_B
levels: BT_A BT_AB BT_B BT_O
str(blood_factor)
Factor w/ 4 levels "BT_A", "BT_AB", "BT_B", "BT_O":  3 2 4 1 4 4 1 3
```

Renaming Levels: labels parameter

Example

```
blood <- c("B", "AB", "O", "A", "O", "O", "A", "B")
blood_factor <- factor(blood,
                       labels = c("BT_A", "BT_AB", "BT_B", "BT_O"))
blood_factor
[1] BT_B, BT_AB, BT_O, BT_A, BT_O, BT_O, BT_A, BT_B
levels: BT_A BT_AB BT_B BT_O
str(blood_factor)
Factor w/ 4 levels "BT_A", "BT_AB", "BT_B", "BT_O":  3 2 4 1 4 4 1 3
```

Reordering & Renaming Levels: levels & labels parameter

Example

```
blood <- c("B", "AB", "O", "A", "O", "O", "A", "B")
blood_factor <- factor(blood,
                       levels = c("O", "A", "B", "AB"),
                       labels = c("BT_O", "BT_A", "BT_B", "BT_AB"))
blood_factor
[1] BT_B, BT_AB, BT_O, BT_A, BT_O, BT_O, BT_A, BT_B
levels: BT_O BT_A BT_B BT_AB
str(blood_factor)
Factor w/ 4 levels "BT_O", "BT_A", "BT_B", "BT_AB": 3 4 1 2 1 1 2 3
```

Nominal v/s Ordinal Factors

Example (Nominal)

```
blood <- c("B", "AB", "O", "A", "O", "O", "A", "B")
blood_factor <- factor(blood)
blood_factor[1] < blood_factor[2]        #[] used to access elements
[1] NA                                   # < generates Warning
Warning message:
In Ops.factor(blood_factor[1], blood_factor[2]): '<' not meaningful for factors
```

Example (Ordinal)

```
tshirt <- c("M", "L", "S", "S", "L", "M", "L", "M")
tshirt_factor <- factor(blood, ordered = TRUE,
                        levels = c("S", "M", "L",))
tshirt_factor
[1] M L S S L M L M
Levels: S < M < L
tshirt_factor[1] < tshirt_factor[2]
[1] TRUE                                 # < gives result
```

Algorithm

1. Start.
2. Create an ordinal factor "Tshirt_sizes".
3. Read choice for factor operations from menu as
    a. Reorder Levels
    b. Rename Levels
    c. Compare Elements
    d. Display
4. As per choice perform factor operations as
    a. If choice is "a", change order of elements.
    b. If choice is "b", change label of elements.
    c. If choice is "c", compare two elements
    d. If choice is "d", display contents of factor.
5. Stop.