20BCE1025_Abhishek_N_N_Lab_5

20BCE1025_Abhishek_N_N

13/09/2022

Q1. Create a data frame for the sports day events as shown in Table 1. It records the points scored by three teams namely – Orange, Yellow and Blue Teams in various sports day events. (1Mark)

```
sportsScoreBoard <- data.frame(
  Sports = c(
    "Football",
    "Cricket",
    "Throw Ball",
    "Badminton",
    "Track and Field"
  ),
  Orange = c(50, 25, 19, 30, 23),
  Yellow = c(0, 20, 25, 43, 21),
  Blue = c(20, 45, 26, 21, 0)
)
sportsScoreBoard
```

```
##            Sports Orange Yellow Blue
## 1        Football     50      0   20
## 2         Cricket     25     20   45
## 3      Throw Ball     19     25   26
## 4       Badminton     30     43   21
## 5 Track and Field     23     21    0
```

Write appropriate R code to implement the following:

a) Include a new team "Green" with the score (10,10,30,30,40) (1 Mark)

```
sportsScoreBoard$Green<-c(10,10,30,30,40)
sportsScoreBoard
```

```
##            Sports Orange Yellow Blue Green
## 1        Football     50      0   20    10
## 2         Cricket     25     20   45    10
## 3      Throw Ball     19     25   26    30
## 4       Badminton     30     43   21    30
## 5 Track and Field     23     21    0    40
```

b) Display the scores of Football and Throw ball events. (1 Mark)

```
sportsScoreBoard[c(1,3),]
```

```
##       Sports Orange Yellow Blue Green
## 1   Football     50      0   20    10
## 3 Throw Ball     19     25   26    30
```

c) Find the total points scored by each team and bind the total score with the data set. (2Marks)

```
sportsScoreBoard <- rbind(
  sportsScoreBoard,
  data.frame(
    Sports = "Total",
    Orange = sum(sportsScoreBoard$Orange),
    Yellow = sum(sportsScoreBoard$Yellow),
    Blue = sum(sportsScoreBoard$Blue),
    Green = sum(sportsScoreBoard$Green)
  )
)
sportsScoreBoard
```

```
##            Sports Orange Yellow Blue Green
## 1        Football     50      0   20    10
## 2         Cricket     25     20   45    10
## 3      Throw Ball     19     25   26    30
## 4       Badminton     30     43   21    30
## 5 Track and Field     23     21    0    40
## 6           Total    147    109  112   120
```

d) Display the name of the winning team. (1 Mark)

```
names(sportsScoreBoard)[which.max(sportsScoreBoard[6,2:5])+1]
```

```
## [1] "Orange"
```

e) Find the least points scored by Orange Team and Blue Team. (1 Mark)

```
apply(sportsScoreBoard[1:5,c(2,4)],2,min)
```

```
## Orange   Blue
##     19      0
```

f) Display the favorable game of Yellow Team and Green Team. (1.5 Marks)

```
# assuming favorable game is in which they scored max
sportsScoreBoard[c(apply(sportsScoreBoard[1:5,c(3,5)],2,which.max)),1]
```

```
## [1] Badminton       Track and Field
## Levels: Badminton Cricket Football Throw Ball Track and Field Total
```

g) Display the average score of each event with the corresponding sports name. (1.5 Marks)

```
sportsScoreBoard$average<-apply(sportsScoreBoard[,2:5],1,mean)
sportsScoreBoard
```

```
##            Sports Orange Yellow Blue Green average
## 1        Football     50      0   20    10      20
## 2         Cricket     25     20   45    10      25
## 3      Throw Ball     19     25   26    30      25
## 4       Badminton     30     43   21    30      31
## 5 Track and Field     23     21    0    40      21
## 6           Total    147    109  112   120     122
```

Q2. Assume that a Fall semester registration report contains various fields such as registration no.,name of the student, course code, course name, credits, faculty name and slot.

A) Identify a suitable data structure to store the names of the courses offered in the Fall semester and justify your choice. [1 mark]. ans: data.frame its a report containing various attributes so with data.frame we can easily represent, visualize and compute required things

B) Write an R code to do the following [3 marks]:

i) Create the identified data structure with a sample set of courses.

```
report<-data.frame(
  regno=c(1,2,3,4,5,6,7,8,9,10),
  name=c('a','b','c','d','e','f','g','h','j','k'),
  code=c(10,20,30,40,50,60,70,80,90,100),
  cname=c("c1","c2","c3","c4","c5","c6","c7","c8","c9","c10"),
  credits=c(1,2,3,4,1,2,3,4,1,2),
  fname=c("f1","f2","f3","f4","f5","f6","f7","f8","f9","f10"),
  slot=c("s1","s2","s3","s4","s5","s6","s7","s8","s9","s10")
)
report
```

```
##    regno name code cname credits fname slot
## 1      1    a   10    c1       1    f1   s1
## 2      2    b   20    c2       2    f2   s2
## 3      3    c   30    c3       3    f3   s3
## 4      4    d   40    c4       4    f4   s4
## 5      5    e   50    c5       1    f5   s5
## 6      6    f   60    c6       2    f6   s6
## 7      7    g   70    c7       3    f7   s7
## 8      8    h   80    c8       4    f8   s8
## 9      9    j   90    c9       1    f9   s9
## 10    10    k  100   c10       2   f10  s10
```

ii) Extract the course names in the data structure stored at locations 2, 5 and 7.

```
report[c(2,4,7),4]
```

```
## [1] c2 c4 c7
## Levels: c1 c10 c2 c3 c4 c5 c6 c7 c8 c9
```

iii) Extract all the course names except at locations 1, 2 and 3.

```
report[c(-1,-2,-3),4]
```

```
## [1] c4  c5  c6  c7  c8  c9  c10
## Levels: c1 c10 c2 c3 c4 c5 c6 c7 c8 c9
```

C) Suggest a suitable data structure to store the values of all fields for a student and justify your choice. [2 marks]. ans: data.frame its a record containing various attributes so with data.frame we can easily represent, visualize and compute required things

D) Write an R code to do the following: [4 marks]

E) Create the identified data structure with a sample values.

```
studentRecord<-data.frame(
  regno=c(1,2,3,4,5,6,7,8,9,10),
  name=c('a','b','c','d','e','f','g','h','j','k'),
  code=c(10,20,30,40,50,60,70,80,90,100),
  cname=c("c1","c2","c3","c4","c5","c6","c7","c8","c9","c10"),
  credits=c(1,2,3,4,1,2,3,4,1,2),
  fname=c("f1","f2","f3","f4","f5","f6","f7","f8","f9","f10"),
  slot=c("s1","s2","s3","s4","s5","s6","s7","s8","s9","s10")
)
studentRecord
```

```
##    regno name code cname credits fname slot
## 1      1    a   10    c1       1    f1   s1
## 2      2    b   20    c2       2    f2   s2
## 3      3    c   30    c3       3    f3   s3
## 4      4    d   40    c4       4    f4   s4
## 5      5    e   50    c5       1    f5   s5
## 6      6    f   60    c6       2    f6   s6
## 7      7    g   70    c7       3    f7   s7
## 8      8    h   80    c8       4    f8   s8
## 9      9    j   90    c9       1    f9   s9
## 10    10    k  100   c10       2   f10  s10
```

II) Assign names to all the values stored in the data structure

```
rownames(studentRecord)<-c("record1","record2","record3","record4","record5","record6","record7","recor
studentRecord
```

```
##          regno name code cname credits fname slot
## record1      1    a   10    c1       1    f1   s1
## record2      2    b   20    c2       2    f2   s2
```

```
## record3      3    c   30    c3       3    f3   s3
## record4      4    d   40    c4       4    f4   s4
## record5      5    e   50    c5       1    f5   s5
## record6      6    f   60    c6       2    f6   s6
## record7      7    g   70    c7       3    f7   s7
## record8      8    h   80    c8       4    f8   s8
## record9      9    j   90    c9       1    f9   s9
## record10    10    k  100   c10       2   f10  s10
```

III) Extract the name of the student, course name, credits and slot in any 2 ways.

```
studentRecord[,c(2,4,5)]
```

```
##          name cname credits
## record1     a    c1       1
## record2     b    c2       2
## record3     c    c3       3
## record4     d    c4       4
## record5     e    c5       1
## record6     f    c6       2
## record7     g    c7       3
## record8     h    c8       4
## record9     j    c9       1
## record10    k   c10       2
```

Q3. As a data analyst you have been provided with the following matrix

Write a R code snippet to do the following (5x1=5 Marks)

i) Create the Matrix and display the matrix

```
peopleMatrix <-
  matrix(
    c(
      "Satchin",
      "Virat",
      "Rohit",
      "Dhoni",
      "Amitab",
      "Amir",
      "Akhya",
      "Salman",
      "Modi",
      "Amit",
      "Rahul",
      "Neharu",
      "Delhi",
      "Chennai",
      "Kolkata",
      "Mumbai"
```

```
    ),
    nrow = 4,
    ncol = 4
  )
peopleMatrix
```

```
##      [,1]      [,2]     [,3]     [,4]
## [1,] "Satchin" "Amitab" "Modi"   "Delhi"
## [2,] "Virat"   "Amir"   "Amit"   "Chennai"
## [3,] "Rohit"   "Akhya"  "Rahul"  "Kolkata"
## [4,] "Dhoni"   "Salman" "Neharu" "Mumbai"
```

ii) Define the column names as Players, Actors, Politicians, Metro city

```
colnames(peopleMatrix)<-c("Players","Actors","Politicians","Metro city")
peopleMatrix
```

```
##      Players   Actors   Politicians Metro city
## [1,] "Satchin" "Amitab" "Modi"      "Delhi"
## [2,] "Virat"   "Amir"   "Amit"      "Chennai"
## [3,] "Rohit"   "Akhya"  "Rahul"     "Kolkata"
## [4,] "Dhoni"   "Salman" "Neharu"    "Mumbai"
```

iii) Define the row names as Record1, Record2, Record3, Record4

```
rownames(peopleMatrix)<-c("Record1","Record2","Record3","Record4")
peopleMatrix
```

```
##         Players   Actors   Politicians Metro city
## Record1 "Satchin" "Amitab" "Modi"      "Delhi"
## Record2 "Virat"   "Amir"   "Amit"      "Chennai"
## Record3 "Rohit"   "Akhya"  "Rahul"     "Kolkata"
## Record4 "Dhoni"   "Salman" "Neharu"    "Mumbai"
```

iv) Display only the names of cricket players and Politicians

```
peopleMatrix[,c(1,3)]
```

```
##         Players   Politicians
## Record1 "Satchin" "Modi"
## Record2 "Virat"   "Amit"
## Record3 "Rohit"   "Rahul"
## Record4 "Dhoni"   "Neharu"
```

v) Display the records that contain Kolkata and Mumbai

```
peopleMatrix[c(3,4),]
```

```
##         Players Actors   Politicians Metro city
## Record3 "Rohit" "Akhya"  "Rahul"     "Kolkata"
## Record4 "Dhoni" "Salman" "Neharu"    "Mumbai"
```

Q4. As a data analyst, you have been asked to retrieve the following information from the given string "8/08/2022". Write an R code snippet to do the following (5x1=5 Marks)

i) Display the given string in date format

```
d1<-as.Date("8/08/2022",format="%d/%m/%Y")
d1
```

```
## [1] "2022-08-08"
```

ii) As a data analyst extract from the date object, the day number in that year, for example, 8/25/2022 is 237th day in this year.

```
format(d1,format="%j")
```

```
## [1] "220"
```

iii) As a data analyst extract from the date object, the month number in that year. For example, 08/25/2022 has the month number 08.

```
format(d1,format="%m")
```

```
## [1] "08"
```

iv) Convert the following dates which are in string format to date format "12/11/2010", "13/12/1990","30/1/2001" and "15/08/2022" and display the dates on the console.

```
ds1<-c("12/11/2010","13/12/1990","30/1/2001","15/08/2022")
d2<-as.Date(ds1,format="%d/%m/%Y")
d2
```

```
## [1] "2010-11-12" "1990-12-13" "2001-01-30" "2022-08-15"
```

```
class(d2)
```

```
## [1] "Date"
```

v) Extract two dates "30/01/2001" and "12/11/2010" from the above created object and calculate the difference between the given dates in terms of days and months and display the results.

```r
difftime(d2[3],d2[1], units = "days")
```

```
## Time difference of -3573 days
```

```r
# direct months argument in units is not accepted so
print(paste("Time difference of",as.double(difftime(d2[3],d2[1], units = "days"))*12/365,"months",sep="
```

```
## [1] "Time difference of -117.468493150685 months"
```