

R version 4.4.2 (2024-10-31 ucrt) -- "Pile of Leaves"
Copyright (C) 2024 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64

```
> x1 <- c(10, 15, 20, 21, 24, 14, 20, 19)
> is.vector(x1)
[1] TRUE
> is.data.frame(x1)
[1] FALSE
> set.seed(123)
> x2 <- sample(1:100, 10)
> x2
[1] 31 79 51 14 67 42 50 43 97 25
> x2
[1] 31 79 51 14 67 42 50 43 97 25
> x2 <- sample(1:100, 10)
> x2
[1] 90 91 69 99 57 92  9 93 72 26
> x3 <- runif(10)
> x3
[1] 0.28915974 0.14711365 0.96302423 0.90229905
[5] 0.69070528 0.79546742 0.02461368 0.47779597
[9] 0.75845954 0.21640794
> set.seed(123)
> x2 <- sample(1:100, 10)
> x2
[1] 31 79 51 14 67 42 50 43 97 25
> set.seed(1234)
> x2 <- sample(1:100, 10)
> x2
[1] 28 80 22  9  5 38 16  4 86 90
> set.seed(1234)
> x9 <- sample(1:100, 10)
> x9
[1] 28 80 22  9  5 38 16  4 86 90
> summary(x2)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   4.00   10.75   25.00   37.80   69.50   90.00
> sample(1:10, 8)
[1]  6  4 10  8  7  9  5  1
> sample(1:10, 8)
[1]  8  4 10  3  9  5  2  1
> sample(1:10, 8)
[1] 2 8 3 4 7 9 5 1
```

```

> sample(1:10, 8, replace=TRUE)
[1] 3 6 4 8 10 2 5 6
> table(sample(1:10, 8, replace=TRUE))

1 3 6 8 9
3 1 2 1 1
> sample(1:10, 8, replace=TRUE)
[1] 8 10 1 8 10 6 3 9
> table(sample(1:10, 8, replace=TRUE))

3 6 7 8 9 10
1 2 1 1 2 1
> sample(1:10, 8, replace=TRUE)
[1] 3 3 2 5 8 10 7 6
> table(sample(1:10, 8, replace=TRUE))

1 3 4 5 6 7 8 9
1 1 1 1 1 1 1 1
> sample(c("Apple", "Banana", "Mnago"), 2)
[1] "Mnago" "Banana"
> sample(c("Apple", "Banana", "Manago"), 2)
[1] "Manago" "Apple"
> sample(c("Apple", "Banana", "Mango"), 2)
[1] "Banana" "Mango"
> x3 <- runif(10)
> x3
[1] 0.9423607 0.4861354 0.2834595 0.2515457
[5] 0.5032552 0.4969662 0.3184458 0.9622228
[9] 0.6340994 0.1274334
> x3 <- runif(10)
> x3
[1] 0.4230470 0.9143169 0.4677923 0.9081691
[5] 0.5977433 0.6317428 0.8691583 0.5027498
[9] 0.9836351 0.3243860
> x3 <- runif(10)
> x3
[1] 0.48137495 0.35698708 0.62747768 0.74160019
[5] 0.56596682 0.98078651 0.57681274 0.43904205
[9] 0.22859970 0.08215807
> set.seed(0)
> x8 <-runif(10)
> x8
[1] 0.8966972 0.2655087 0.3721239 0.5728534
[5] 0.9082078 0.2016819 0.8983897 0.9446753
[9] 0.6607978 0.6291140
> set.seed(0)
> x8 <-runif(10)

```

```

> x8
[1] 0.8966972 0.2655087 0.3721239 0.5728534
[5] 0.9082078 0.2016819 0.8983897 0.9446753
[9] 0.6607978 0.6291140
> set.seed(0)
> x8 <- runif(10)
> x8
[1] 0.8966972 0.2655087 0.3721239 0.5728534
[5] 0.9082078 0.2016819 0.8983897 0.9446753
[9] 0.6607978 0.6291140
> help(sample)
> sample(x=c("Apple", "Banana", "Mango"), size= 2, prob=c(0.5,0.5, 0.0, 0.0))
Error in sample.int(length(x), size, replace, prob) :
  incorrect number of probabilities
> sample(x=c("Apple", "Banana", "Mango"), size= 2, prob=c(0.5,0.5, 0.4, 0.2))
Error in sample.int(length(x), size, replace, prob) :
  incorrect number of probabilities
> sample(x=c("Apple", "Banana", "Mango"), size= 2, prob=c(0.5,0.4, 0.1, 0.0))
Error in sample.int(length(x), size, replace, prob) :
  incorrect number of probabilities
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1))
[1] "Apple" "Banana"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1))
[1] "Apple" "Banana"
> x3 <- runif(10)
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1))
[1] "Apple" "Banana"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1))
[1] "Apple" "Banana"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1))
[1] "Apple" "Mango"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1))
[1] "Apple" "Banana"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1))
[1] "Banana" "Apple"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1),
replace=TRUE)
[1] "Apple" "Banana"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1),
replace=TRUE)
[1] "Banana" "Banana"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1),
replace=TRUE)
[1] "Apple" "Banana"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1),
replace=TRUE)
[1] "Apple" "Banana"

```

```

> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1),
replace=TRUE)
[1] "Banana" "Banana"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1),
replace=TRUE)
[1] "Banana" "Banana"
> sample(1:10, 8, replace=TRUE)
[1] 4 10 9 7 6 9 8 9
> sample(1:10, 8, replace=TRUE)
[1] 7 8 6 10 7 3 10 6
> sample(1:10, 8, replace=TRUE)
[1] 8 2 2 6 6 1 3 3
> sample(1:10, 8, replace=TRUE)
[1] 8 6 7 6 8 7 1 4
> sample(1:10, 8, replace=TRUE)
[1] 8 9 9 7 4 7 6 1
> sample(1:10, 8, replace=TRUE)
[1] 5 6 1 9 7 7 3 6
> sample(1:10, 8, replace=TRUE)
[1] 2 10 10 7 3 2 10 1
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1),
replace=FALSE)
[1] "Banana" "Apple"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1),
replace=FALSE)
[1] "Apple" "Banana"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1),
replace=FALSE)
[1] "Mango" "Banana"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1),
replace=FALSE)
[1] "Banana" "Apple"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1),
replace=FALSE)
[1] "Mango" "Apple"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1),
replace=FALSE)
[1] "Banana" "Banana"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1),
replace=FALSE)
[1] "Apple" "Banana"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.5, 0.4, 0.1),
replace=FALSE)
[1] "Banana" "Apple"
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.25, 0.25,
0.25,0.25), replace=FALSE)
Error in sample.int(length(x), size, replace, prob) :

```

```

incorrect number of probabilities
> sample(x = c("Apple", "Banana", "Mango"), size = 2, prob = c(0.25, 0.25,
0.25,0.25), replace=FALSE)
Error in sample.int(length(x), size, replace, prob) :
  incorrect number of probabilities
> sample(x = c("Apple", "Banana", "Mango", "Lichi"), size = 2, prob = c(0.25,
0.25, 0.25,0.25), replace=FALSE)
[1] "Banana" "Apple"
> sample(x = c("Apple", "Banana", "Mango", "Lichi"), size = 2, prob = c(0.25,
0.25, 0.25,0.25), replace=FALSE)
[1] "Banana" "Apple"
> sample(x = c("Apple", "Banana", "Mango", "Lichi"), size = 2, prob = c(0.25,
0.25, 0.25,0.25), replace=FALSE)
[1] "Lichi" "Mango"
> sample(x = c("Apple", "Banana", "Mango", "Lichi"), size = 2, prob = c(0.25,
0.25, 0.25,0.25), replace=FALSE)
[1] "Mango" "Lichi"
> sample(x = c("Apple", "Banana", "Mango", "Lichi"), size = 2, prob = c(0.25,
0.25, 0.25,0.25), replace=FALSE)
[1] "Lichi" "Banana"
> sample(x = c("Apple", "Banana", "Mango", "Lichi"), size = 2, prob = c(0.25,
0.25, 0.25,0.25), replace=FALSE)
[1] "Lichi" "Banana"
> income=c(10000, 25000, 40000, 25000)
> household=c(10,5,2,4)
> sample(income, size=2, prob=household)
[1] 10000 25000
> income=c(10000, 25000, 40000, 25000)
> household=c(10,5,2,4)
> sample(income, size=2, prob=household)
[1] 10000 40000
> income=c(10000, 25000, 40000, 25000)
> household=c(10,5,2,4)
> sample(income, size=2, prob=household)
[1] 10000 25000
> income=c(10000, 25000, 40000, 25000)
> household=c(10,5,2,4)
> sample(income, size=2, prob=household)
[1] 25000 10000
> income=c(10000, 25000, 40000, 25000)
> household=c(10,5,2,4)
> sample(income, size=2, prob=household)
[1] 10000 25000
> income=c(10000, 25000, 40000, 25000)
> household=c(10,5,2,4)
> sample(income, size=2, prob=household)
[1] 25000 10000

```

```

> income=c(10000, 25000, 40000, 25000)
> household=c(10,5,2,4)
> sample(income, size=2, prob=household)
[1] 25000 25000
> income=c(10000, 25000, 40000, 25000)
> household=c(10,5,2,4)
> sample(income, size=2, prob=household)
[1] 25000 10000
> income=c(10000, 25000, 40000, 25000)
> household=c(10,5,2,4)
> sample(income, size=2, prob=household)
[1] 10000 40000
> income=c(10000, 25000, 40000, 25000)
> household=c(10,5,2,4)
> sample(income, size=2, prob=household)
[1] 25000 25000
> income=c(10000, 25000, 40000, 25000)
> household=c(10,5,2,4)
> sample(income, size=2, prob=household)
[1] 25000 40000
> income=c(10000, 25000, 40000, 25000)
> household=c(10,5,2,4)
> sample(income, size=2, prob=household)
[1] 10000 25000
> source("D:/RProgramming/Class3/Class3(inClass).R")
> household/sum(household)
[1] 0.4761905 0.2380952 0.0952381 0.1904762
> x6 <- 10:1
> x6
[1] 10 9 8 7 6 5 4 3 2 1
> x4 <- seq(from=1, to=11, by=2)
> x4
[1] 1 3 5 7 9 11
> x5 <- seq(from =1, to=11, length.out=5)
> x5
[1] 1.0 3.5 6.0 8.5 11.0
> ?? "random"
> seq_len(10)
[1] 1 2 3 4 5 6 7 8 9 10
> ## Selection
> x1>19
[1] FALSE FALSE TRUE TRUE TRUE FALSE TRUE
[8] FALSE
> x1[c(FALSE, FALSE, TRUE, TRUE,TRUE,FALSE,TRUE,FALSE)]
[1] 20 21 24 20
> x1[c(3,4,5,7)]
[1] 20 21 24 20

```

```

> x1[x1>19]
[1] 20 21 24 20
> x1[x1 > 14 & x1 < 20]
[1] 15 19
> which(x1[x1 > 14 & x1 < 20])
Error in which(x1[x1 > 14 & x1 < 20]) :
  argument to 'which' is not logical
> which(x1 > 14 & x1 < 20)
[1] 2 8
> ## replace
> replace(x1, x1>19, 15)
[1] 10 15 15 15 15 14 15 19
> x1[x1>19] <- 15
> x1
[1] 10 15 15 15 15 14 15 19
> replace(x1, which(x1>19), 15)
[1] 10 15 15 15 15 14 15 19
> x1[x1>19] <- 15
> x1
[1] 10 15 15 15 15 14 15 19
> source("D:/RProgramming/Class3/Class3(inClass).R")
> x1[x1==0] <-15
> x1
[1] 10 15 15 15 15 14 15 19
> x1 <- c(10, 15, 20, 21, 24, 14, 20, 19)
> replace(x1, x1>19, NA)
[1] 10 15 NA NA NA 14 NA 19
> x1 <- replace(x1, which(x1>19), 15)
> mean(x1, n.rm=TRUE)
[1] 14.75
> mean
function (x, ...)
UseMethod("mean")
<bytecode: 0x000001d1159c6e28>
<environment: namespace:base>
> # replace(x1, x1>19, NA)
> # x1 <- replace(x1, which(x1>19), 15)
> # mean(x1, n.rm=TRUE)
> # mean
>
> # Replace elements greater than 19 with NA
> x1 <- replace(x1, x1 > 19, NA)
>
> # Replace elements greater than 19 with 15
> x1 <- replace(x1, which(x1 > 19), 15)
>
> # Calculate the mean of x1, ignoring NA values

```

```
> mean_x1 <- mean(x1, na.rm = TRUE)
>
> # Print the mean value
> print(mean_x1)
[1] 14.75
> x1 <- replace(x1, x1 > 19, NA)
>
> x1 <- replace(x1, which(x1 > 19), 15)
>
> mean_x1 <- mean(x1, na.rm = TRUE)
>
> print(mean_x1)
[1] 14.75
>
>
>
>
>
>
>
>
>
>
>
>
>
> ?matrix
> # Matrix -----
> mat1 <- matrix(c(1,2,3,4), ncol=2, nrow=2, byrow=TRUE)
> mat1
      [,1] [,2]
[1,]    1    2
[2,]    3    4
> is.matrix(mat1)
[1] TRUE
> is.matrix(matrix(c(0,10, 2), ncol=3))
[1] TRUE
> is.matrix(c(0,10,2))
[1] FALSE
> mat1[, ]
      [,1] [,2]
[1,]    1    2
[2,]    3    4
> mat[1, ]
Error: object 'mat' not found
> # Creating the first matrix
> matrix1 <- matrix(c(1, 2, 3, 4, 5, 6), nrow = 2, ncol = 3)
```



```

> print(matrix1)
      [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
>
> # Creating the second matrix
> matrix2 <- matrix(c(7, 8, 9, 10, 11, 12), nrow = 2, ncol = 3)
> print(matrix2)
      [,1] [,2] [,3]
[1,]    7    9   11
[2,]    8   10   12
> mat1 %*% mat2
Error: object 'mat2' not found
> matrix1 %*% matrix2
Error in matrix1 %*% matrix2 : non-conformable arguments
> ## Matrix multiplication
>
> # Creating the first matrix
> matrix1 <- matrix(c(1, 2, 3, 4, 5, 6), nrow = 2, ncol = 3)
> print(matrix1)
      [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
>
> # Creating the second matrix
> matrix2 <- matrix(c(7, 8, 9, 10, 11, 12), nrow = 3, ncol = 2)
> print(matrix2)
      [,1] [,2]
[1,]    7   10
[2,]    8   11
[3,]    9   12
>
> # Multiplying the two matrices
> result <- matrix1 %*% matrix2
> print(result)
      [,1] [,2]
[1,]   76  103
[2,]  100  136
> # Checking the dimensions of the first matrix
> dim_matrix1 <- dim(matrix1)
> print(dim_matrix1)
[1] 2 3
>
> # Checking the dimensions of the second matrix
> dim_matrix2 <- dim(matrix2)
> print(dim_matrix2)
[1] 3 2

```

```

> # Computing the Kronecker product
> kronecker_product <- kronecker(matrix1, matrix2)
> print(kronecker_product)
      [,1] [,2] [,3] [,4] [,5] [,6]
[1,]     7    10    21    30    35    50
[2,]     8    11    24    33    40    55
[3,]     9    12    27    36    45    60
[4,]    14    20    28    40    42    60
[5,]    16    22    32    44    48    66
[6,]    18    24    36    48    54    72
> source("D:/RProgramming/Class3/Class3(inClass).R")
Error in eval(ei, envir) : object 'mat' not found
> # Creating vectors
> names <- c("Alice", "Bob", "Charlie")
> ages <- c(25, 30, 35)
> genders <- c("Female", "Male", "Male")
>
> # Creating a DataFrame
> df <- data.frame(Name = names, Age = ages, Gender = genders)
>
> # Printing the DataFrame
> print(df)
      Name Age Gender
1  Alice  25 Female
2   Bob  30   Male
3 Charlie  35   Male
> df1 <- data.frame(id = c(1,2,3),
+                   age= c(24, 23, 19),
+                   gender = c("F", "M", "F"), stringAsFactors=TRUE)
> df1
  id age gender stringAsFactors
1  1  24      F              TRUE
2  2  23      M              TRUE
3  3  19      F              TRUE
> is.data.frame(df1)
[1] TRUE
> summary(df1)
      id      age
Min.   :1.0   Min.   :19.0
1st Qu.:1.5   1st Qu.:21.0
Median :2.0   Median :23.0
Mean    :2.0   Mean    :22.0
3rd Qu.:2.5   3rd Qu.:23.5
Max.    :3.0   Max.    :24.0
  gender      stringAsFactors
Length:3          Mode:logical
Class :character   TRUE:3

```

Mode :character

```
> df1 <- data.frame(id = c(1,2,3,4),
+                   age = c(24, 23, 19, 10),
+                   gender = c("F", "M", "F", "F"),
+                   location = c("rural","urban","rural","urban"),
+                   stringsAsFactors = TRUE)
> df1
  id age gender location
1  1  24      F   rural
2  2  23      M   urban
3  3  19      F   rural
4  4  10      F   urban
> is.data.frame(df1)
[1] TRUE
> summary(df1)
      id          age          gender
Min.   :1.00   Min.   :10.00   F:3
1st Qu.:1.75   1st Qu.:16.75   M:1
Median :2.50   Median :21.00
Mean   :2.50   Mean    :19.00
3rd Qu.:3.25   3rd Qu.:23.25
Max.    :4.00   Max.    :24.00
 location
rural:2
urban:2

> is.character(c("F","M","F","F"))
[1] TRUE
> as.factor(c("F","M","F","F"))
[1] F M F F
Levels: F M
> is.factor(c("F","M","F","F"))
[1] FALSE
> is.numeric(as.factor(c("F","M","F","F")))
[1] FALSE
> df1$id
[1] 1 2 3 4
> df1$age
[1] 24 23 19 10
> df1$gender
[1] F M F F
Levels: F M
> mean(df1$age)
[1] 19
```

```

> df1$id
[1] 1 2 3 4
> df1$age
[1] 24 23 19 10
> mean(df1$age)
[1] 19
> df1$gender
[1] F M F F
Levels: F M
> rownames(df1)
[1] "1" "2" "3" "4"
> colnames(df1)
[1] "id"      "age"      "gender"   "location"
> rownames(df1)
[1] "1" "2" "3" "4"
> colnames(df1) <-c('id','age','gender','location' )
> colnames(df1)=="age"
[1] FALSE  TRUE  FALSE  FALSE
> colnames(df1)[colnames(df1)=="age"] <- "Age_of_respondent"
> rownames(df1)
[1] "1" "2" "3" "4"
> rownames(df1) <- c("person 1", "galib", "kashem", "roni")
> rownames(df1)
[1] "person 1" "galib"      "kashem"      "roni"
> df1[2,]
      id Age_of_respondent gender location
galib  2                 23      M    urban
> df[c(1,4),]
Error in df[c(1, 4), ] : object of type 'closure' is not subsettable
> df[c(1,4),]
Error in df[c(1, 4), ] : object of type 'closure' is not subsettable
> df1[c(1,4),]
      id Age_of_respondent gender location
person 1  1                 24      F    rural
roni      4                 10      F    urban
> df1$Age_of_respondent <= 23
[1] FALSE  TRUE  TRUE  TRUE
> df1[df1$Age_of_respondent <= 23,]
      id Age_of_respondent gender location
galib  2                 23      M    urban
kashem 3                 19      F    rural
roni   4                 10      F    urban
> !is.na(df1$Age_of_respondent)
[1] TRUE TRUE TRUE TRUE
> is.na(df1$Age_of_respondent)
[1] FALSE FALSE FALSE FALSE
> df1[df1$Age_of_respondent <= 23 & !is.na(df1$Age_of_respondent),]
      id Age_of_respondent gender location
galib  2                 23      M    urban

```

```

kashem 3          19      F      rural
roni   4          10      F      urban
> df1$height <- c(5,6,5,5)
> df1
      id Age_of_respondent gender location
person 1 1              24      F      rural
galib   2              23      M      urban
kashem  3              19      F      rural
roni    4              10      F      urban
      height
person 1 5
galib    6
kashem   5
roni     5
> df1$height <- c(5,6,5,5,6,1)
Error in `$<-data.frame`(`*tmp*`, height, value = c(5, 6, 5, 5, 6, 1)) :
  replacement has 6 rows, data has 4
> df1$height <- c(5,6,5.5,6.1)
> df1
      id Age_of_respondent gender location
person 1 1              24      F      rural
galib   2              23      M      urban
kashem  3              19      F      rural
roni    4              10      F      urban
      height
person 1 5.0
galib    6.0
kashem   5.5
roni     6.1
> df1$age_in_months <- df1$Age_of_respondent * 12
> ls1 <- list(c(1,2,3), c(24,23,19), c("F","M","F"))
> ls1
[[1]]
[1] 1 2 3

[[2]]
[1] 24 23 19

[[3]]
[1] "F" "M" "F"

> source("D:/RProgramming/Class3/Self/Class3(inClass).R", echo=TRUE)

> # Data structure
> # Author: Md mahfujul karim Sheikh
> # Date: 03-01-25
>
>
> # Vector -----

```

```

>
> x1 <- c(10, 15, 20, 21, 24, 14, 20, 19)

> is.vector(x1)
[1] TRUE

> is.data.frame(x1)
[1] FALSE

> ## Random number generator
>
> set.seed(123)

> x2 <- sample(1:100, 10)

> x2
[1] 31 79 51 14 67 42 50 43 97 25

> # set.seed(1234)
> # x9 <- sample(1:100, 10)
> # x9
> # summary(x2)
>
> sample(1:10, 8)
[1] 10 5 3 8 1 4 6 9

> sample(1:10, 8, replace=TRUE)
[1] 3 8 10 7 10 9 3 4

> table(sample(1:10, 8, replace=TRUE))

 1  5  7  9 10
1  1  2  2  2

> sample(c("Apple", "Banana", "Mango"), 2)
[1] "Mango" "Apple"

> sample(x = c("Apple", "Banana", "Mango", "Lichi"), size = 2, prob = c(0.25,
0.25, 0.25, 0.25), replace=FALSE)
[1] "Banana" "Lichi"

> income=c(10000, 25000, 40000, 25000)

> household=c(10,5,2,4)

> sample(income, size=2, prob=household)
[1] 25000 10000

> household/sum(household)

```

```

[1] 0.4761905 0.2380952 0.0952381 0.1904762

> # set.seed(0)
> # x8 <- runif(10)
> # x8
>
>
> x3 <- runif(10)

> x3
[1] 0.56094798 0.20653139 0.12753165 0.75330786
[5] 0.89504536 0.37446278 0.66511519 0.09484066
[9] 0.38396964 0.27438364

> help(sample)

> ## Sequence Generator
>
> x4 <- seq(from=1, to=11, by=2)

> x4
[1] 1 3 5 7 9 11

> x5 <- seq(from =1, to = 11, length.out=5)

> x5
[1] 1.0 3.5 6.0 8.5 11.0

> # Talk Later
> seq(from =1, to = 11, length.out=5)
[1] 1.0 3.5 6.0 8.5 11.0

> seq(from =1, to = 11, along.with=5)
[1] 1

> x6 <- 10:1

> x6
[1] 10 9 8 7 6 5 4 3 2 1

> seq_len(10)
[1] 1 2 3 4 5 6 7 8 9 10

> ## Selection
> x1>19
[1] FALSE FALSE TRUE TRUE TRUE FALSE TRUE
[8] FALSE

> x1[c(FALSE, FALSE, TRUE, TRUE,TRUE,FALSE,TRUE,FALSE)]

```

```

[1] 20 21 24 20

> x1[c(3,4,5,7)]
[1] 20 21 24 20

> x1[x1>19]
[1] 20 21 24 20

> x1[x1 > 14 & x1 < 20]
[1] 15 19

> which(x1 > 14 & x1 < 20)
[1] 2 8

> ## replace
> replace(x1, x1>19, 15)
[1] 10 15 15 15 15 14 15 19

> x1[x1>19] <- 15

> x1
[1] 10 15 15 15 15 14 15 19

> # replace(x1, which(x1>19), 15)
> # x1[x1>19] <- 15
> # x1
>
> x1[x1==0] <-15

> x1
[1] 10 15 15 15 15 14 15 19

> # replace(x1, x1>19, NA)
> # x1 <- replace(x1, which(x1>19), 15)
> # mean(x1, n.rm=TRUE)
> # mean
>
>
> # x1 <- replace(x1, x1 > 19, NA)
> #
> # .... [TRUNCATED]

> mat1
      [,1] [,2]
[1,]    1    2
[2,]    3    4

> is.matrix(mat1)
[1] TRUE

```



```

> is.matrix(matrix(c(0,10, 2), ncol=3))
[1] TRUE

> is.matrix(c(0,10,2))
[1] FALSE

> mat1[,]
      [,1] [,2]
[1,]     1     2
[2,]     3     4

> mat[1,]
Error in eval(ei, envir) : object 'mat' not found
> ls1 <- list(c(1,2,3), c(24,23,19), c("F","M","F","M"))
> ls1
[[1]]
[1] 1 2 3

[[2]]
[1] 24 23 19

[[3]]
[1] "F" "M" "F" "M"

> df1
      id Age_of_respondent gender location
person 1    1             24      F    rural
galib    2             23      M    urban
kashem   3             19      F    rural
roni     4             10      F    urban
      height age_in_months
person 1    5.0          288
galib     6.0          276
kashem    5.5          228
roni      6.1          120
> ls1[[1]]
[1] 1 2 3
> ls1[[2]]
[1] 24 23 19
> ls1[[3]]
[1] "F" "M" "F" "M"
> ls1 <- list(c(1,2,3), c(24,23,19), c("F","M","F"))
> ls2 <- list(id = c(1,2,3), age = c(24, 23, 19), gender = c("F","M","F"))
> ls3 <- list(id = c(1,2,3), age = c(24, 23, 19), gender = c("F","M","F"),
array1 = arr1)
Error: object 'arr1' not found

```