

1 Use various utility functions discussed to create vectors containing these values. You need to create both [first containing 10 values and another containing 26] vectors separately.

Solution:

```
2^(1:10)
```

```
## [1] 2 4 8 16 32 64 128 256 512 1024
```

```
paste0(letters,26:1)
```

```
## [1] "a26" "b25" "c24" "d23" "e22" "f21" "g20" "h19" "i18" "j17" "k16"
## [12] "l15" "m14" "n13" "o12" "p11" "q10" "r9" "s8" "t7" "u6" "v5"
## [23] "w4" "x3" "y2" "z1"
```

2 Below given is a vector containing first lines of many addresses. Extract city names from all the addresses using string functions. Hint: you can use a for loop to iterate over results of strsplit for further processing.

```
address_list=c("802/hiranandani/Mumbai",
               "2A/kalka-Delhi",
               "345#near adyar#Chennai",
               "10-shyaam bazzar-Kolkata")
```

Solution:

```
temp=gsub("-", "/", address_list)
temp=gsub("#", "/", temp)
l=strsplit(temp, "/")

for(i in 1:4){
  print(l[[i]][3])
}
```

```
## [1] "Mumbai"
## [1] "Delhi"
## [1] "Chennai"
## [1] "Kolkata"
```

3 Use following bit to create a vector with prime numbers in 1:47. [Prime numbers are numbers which are divisible only by themselves.]

```
primes=c(2,3,5,7,11,13,17,19,23,29,31,37,41,43,47)
```

Any number from 48 to 100 which is not divisible by any of the above listed primes is also a prime. Print those prime numbers from 48 to 100.

Solution:

```
for(i in 48:100){
  temp=i%%primes
  condition=temp==0
  if(sum(condition)==0){print(i)}
}
```

```
## [1] 53
## [1] 59
## [1] 61
## [1] 67
## [1] 71
## [1] 73
## [1] 79
## [1] 83
## [1] 89
## [1] 97
```

4 Find out , how many cars are there are in the dataset mtcars which have automatic transmission, number of forward gears higher than 3 and below average mileage. List their names. [calculate average mileage from the data itself]. To find out which variable in the data represent mentioned above information do ?mtcars

Solution:

```
?mtcars
avg=mean(mtcars$mpg)
d=mtcars[mtcars$mpg<avg & mtcars$gear>3 & mtcars$am==0,]
rownames(d)
```

```
## [1] "Merc 280" "Merc 280C"
```

5 There is no native function in R to calculate mode for a variable. The function “mode” returns storage mode of an object, not the statistical mode that we discussed in the class.

write a function which returns modes of a character vector. Test that on the following vectors

```
set.seed(2)
x=sample(letters[1:5],50,replace=T)
y=sample(letters[1:3],50,replace=T)
```

Solution:

```
mymode=function(x){
  t=table(x)
  result=names(t)[which(t==max(t))]
  return(result)
}
mymode(x)
```

```
## [1] "e"
```

```
mymode(y)
```

```
## [1] "a"
```

Prepared By : Lalit Sachan

Contact : lalit.sachan@edvancer.in