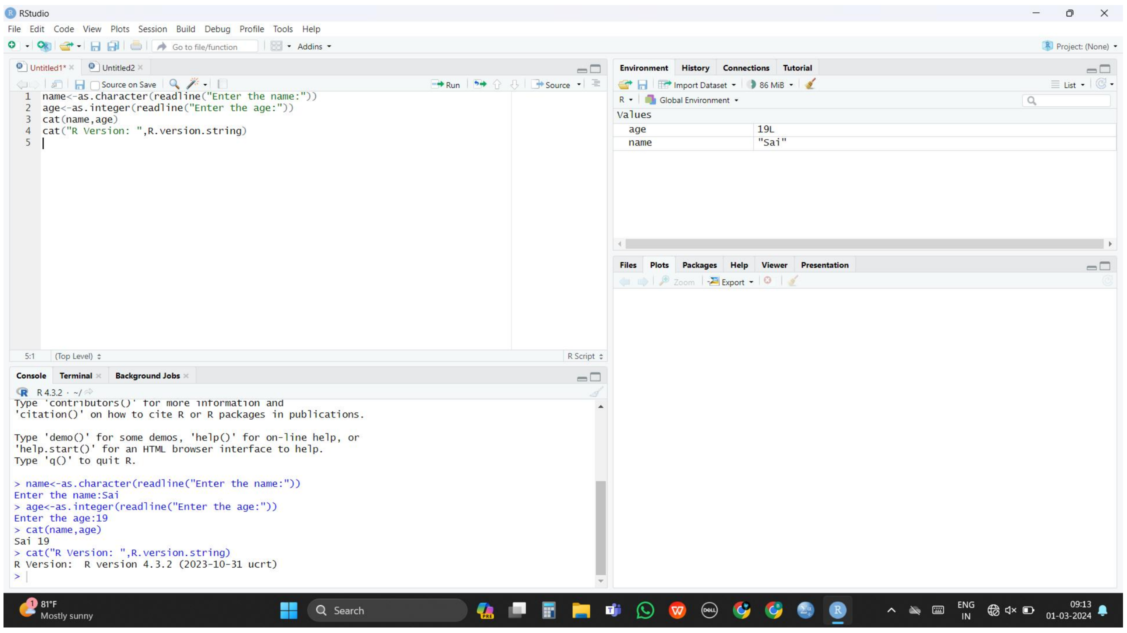
1. Write a R program to take input from the user (name and age) and display the values. Also print the version of R installation.

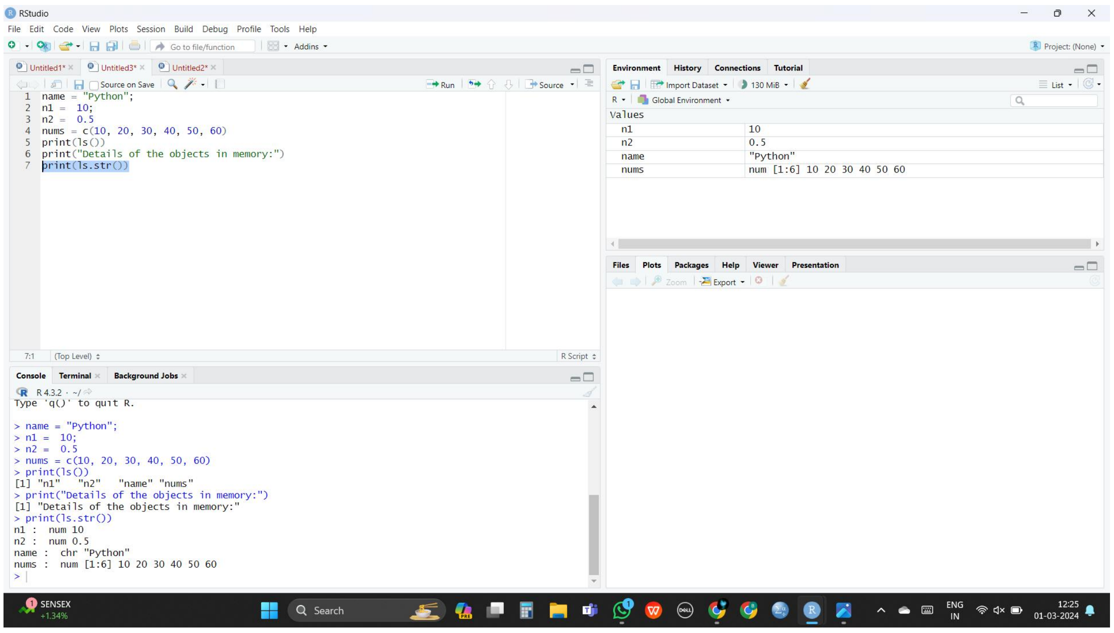


```r
name<-as.character(readline("Enter the name:"))
age<-as.integer(readline("Enter the age:"))
cat(name,age)
cat("R Version: ",R.version.string)
```

```
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> name<-as.character(readline("Enter the name:"))
Enter the name:Sai
> age<-as.integer(readline("Enter the age:"))
Enter the age:19
> cat(name,age)
Sai 19
> cat("R Version: ",R.version.string)
R Version:  R version 4.3.2 (2023-10-31 ucrt)
>
```

```
Values
  age        19L
  name       "Sai"
```

2. Write a R program to get the details of the objects in memory

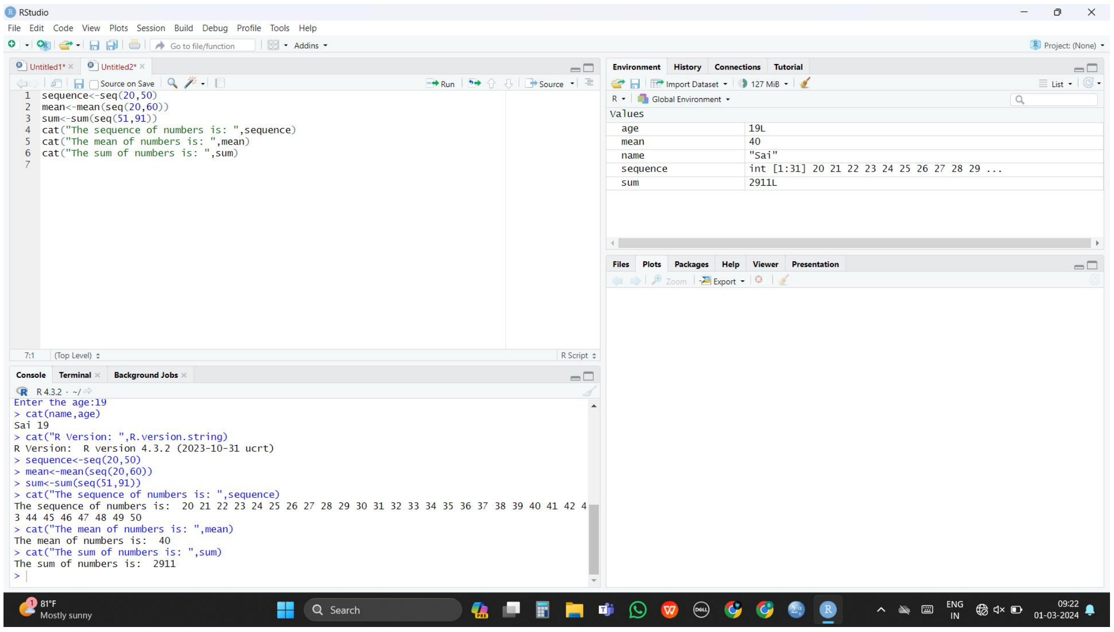


```r
name = "Python";
n1 =  10;
n2 =  0.5
nums = c(10, 20, 30, 40, 50, 60)
print(ls())
print("Details of the objects in memory:")
print(ls.str())
```

```
Type 'q()' to quit R.

> name = "Python";
> n1 =  10;
> n2 =  0.5
> nums = c(10, 20, 30, 40, 50, 60)
> print(ls())
[1] "n1"   "n2"   "name" "nums"
> print("Details of the objects in memory:")
[1] "Details of the objects in memory:"
> print(ls.str())
n1 :  num 10
n2 :  num 0.5
name :  chr "Python"
nums :  num [1:6] 10 20 30 40 50 60
>
```

```
Values
  n1         10
  n2         0.5
  name       "Python"
  nums       num [1:6] 10 20 30 40 50 60
```

3. Write a R program to create a sequence of numbers from 20 to 50 and find the mean of numbers from 20 to 60 and sum of numbers from 51 to 91

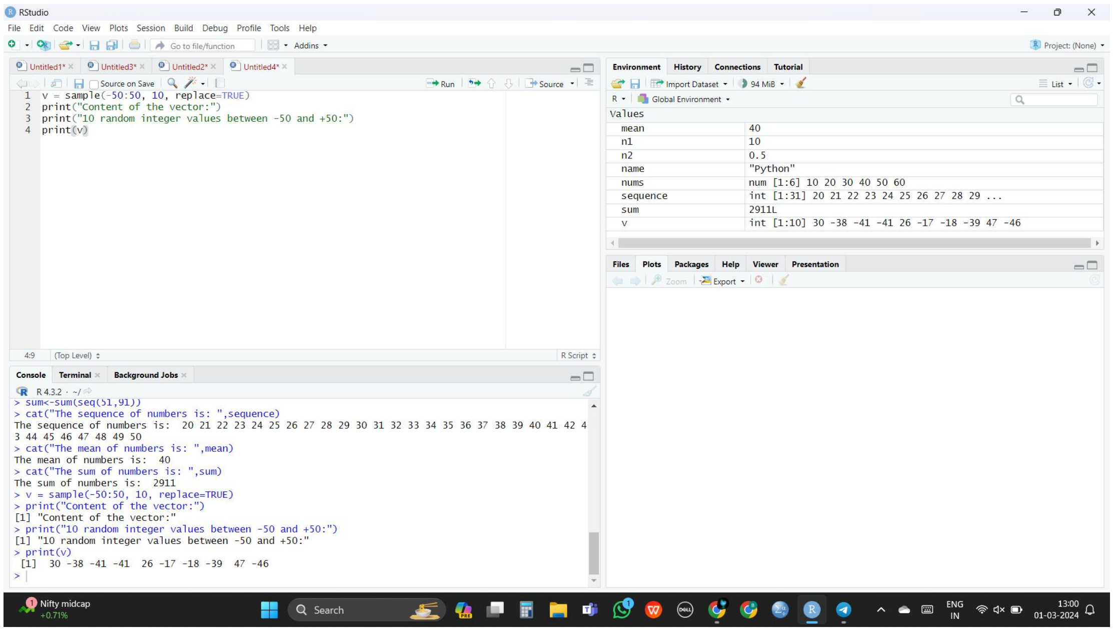


```r
sequence<-seq(20,50)
mean<-mean(seq(20,60))
sum<-sum(seq(51,91))
cat("The sequence of numbers is: ",sequence)
cat("The mean of numbers is: ",mean)
cat("The sum of numbers is: ",sum)
```

```
Values
  age          19L
  mean         40
  name         "Sai"
  sequence     int [1:31] 20 21 22 23 24 25 26 27 28 29 ...
  sum          2911L
```

```
Enter the age:19
> cat(name,age)
Sai 19
> cat("R Version: ",R.version.string)
R Version:  R version 4.3.2 (2023-10-31 ucrt)
> sequence<-seq(20,50)
> mean<-mean(seq(20,60))
> sum<-sum(seq(51,91))
> cat("The sequence of numbers is: ",sequence)
The sequence of numbers is:  20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 4
3 44 45 46 47 48 49 50
> cat("The mean of numbers is: ",mean)
The mean of numbers is:  40
> cat("The sum of numbers is: ",sum)
The sum of numbers is:  2911
> 
```

4. Write a R program to create a vector which contains 10 random integer values between -50 and +50.

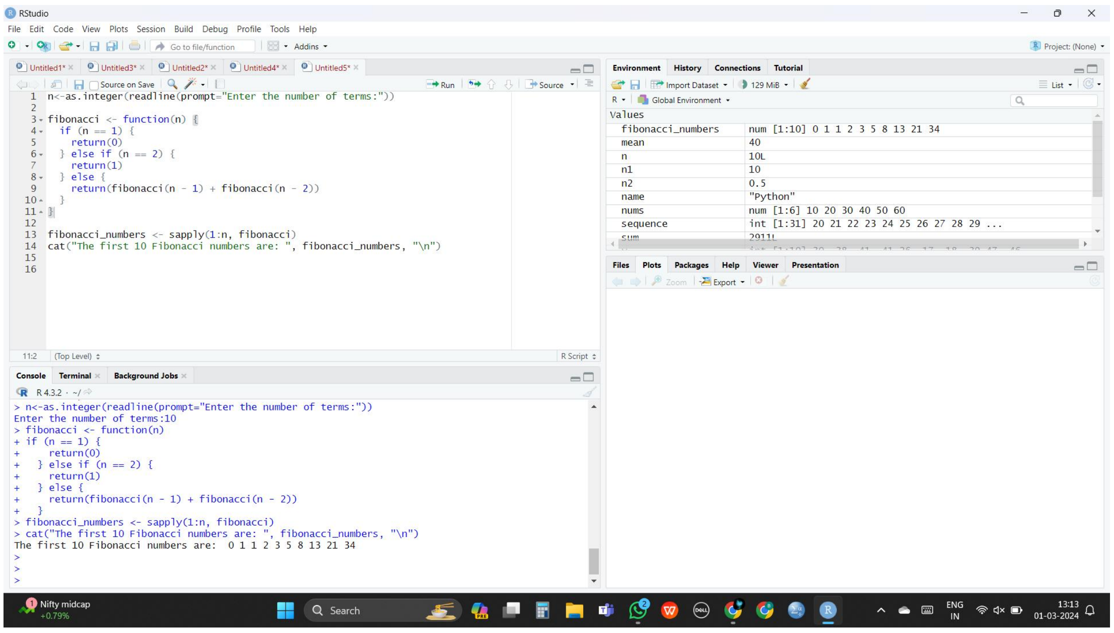


```r
v = sample(-50:50, 10, replace=TRUE)
print("Content of the vector:")
print("10 random integer values between -50 and +50:")
print(v)
```

```
Values
  mean         40
  n1           10
  n2           0.5
  name         "Python"
  nums         num [1:6] 10 20 30 40 50 60
  sequence     int [1:31] 20 21 22 23 24 25 26 27 28 29 ...
  sum          2911L
  v            int [1:10] 30 -38 -41 -41 26 -17 -18 -39 47 -46
```

```
> sum<-sum(seq(51,91))
> cat("The sequence of numbers is: ",sequence)
The sequence of numbers is:  20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 4
3 44 45 46 47 48 49 50
> cat("The mean of numbers is: ",mean)
The mean of numbers is:  40
> cat("The sum of numbers is: ",sum)
The sum of numbers is:  2911
> v = sample(-50:50, 10, replace=TRUE)
> print("Content of the vector:")
[1] "Content of the vector:"
> print("10 random integer values between -50 and +50:")
[1] "10 random integer values between -50 and +50:"
> print(v)
 [1]  30 -38 -41 -41  26 -17 -18 -39  47 -46
> 
```

5. Write a R program to get the first 10 Fibonacci numbers.

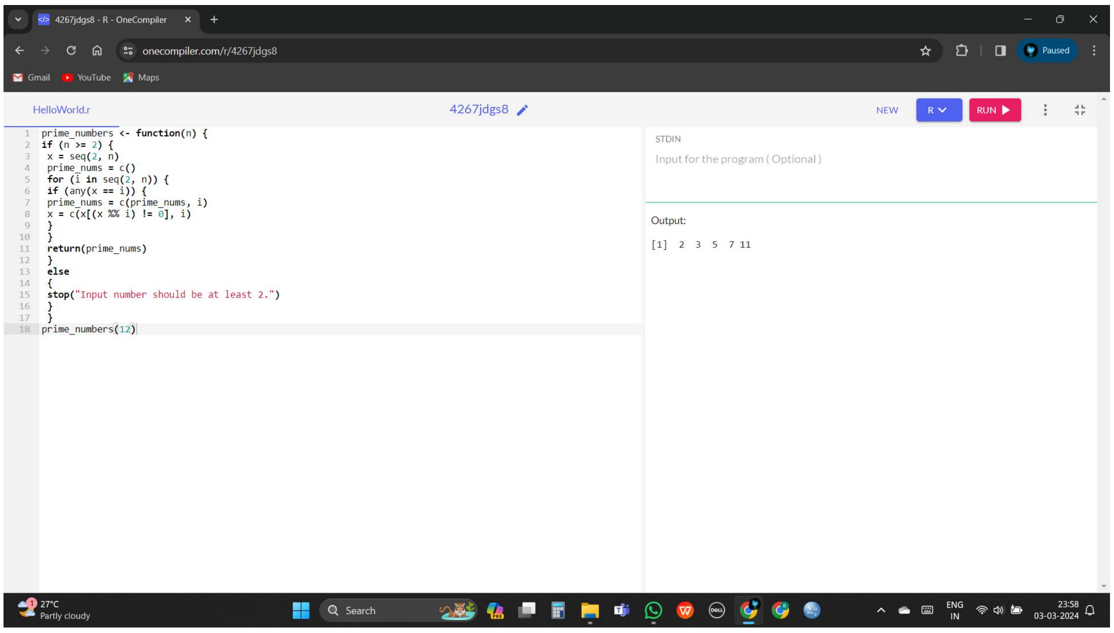


```r
n<-as.integer(readline(prompt="Enter the number of terms:"))

fibonacci <- function(n) {
  if (n == 1) {
    return(0)
  } else if (n == 2) {
    return(1)
  } else {
    return(fibonacci(n - 1) + fibonacci(n - 2))
  }
}

fibonacci_numbers <- sapply(1:n, fibonacci)
cat("The first 10 Fibonacci numbers are: ", fibonacci_numbers, "\n")
```

```
> n<-as.integer(readline(prompt="Enter the number of terms:"))
Enter the number of terms:10
> fibonacci <- function(n)
+ if (n == 1) {
+     return(0)
+   } else if (n == 2) {
+     return(1)
+   } else {
+     return(fibonacci(n - 1) + fibonacci(n - 2))
+   }
> fibonacci_numbers <- sapply(1:n, fibonacci)
> cat("The first 10 Fibonacci numbers are: ", fibonacci_numbers, "\n")
The first 10 Fibonacci numbers are:  0 1 1 2 3 5 8 13 21 34
>
>
>
```

```
Values
  fibonacci_numbers     num [1:10] 0 1 1 2 3 5 8 13 21 34
  mean                  40
  n                     10L
  n1                    10
  n2                    0.5
  name                  "Python"
  nums                  num [1:6] 10 20 30 40 50 60
  sequence              int [1:31] 20 21 22 23 24 25 26 27 28 29 ...
  sum                   2911L
```

6. Write a R program to get all prime numbers up to a given number (based on the sieve of Eratosthenes)

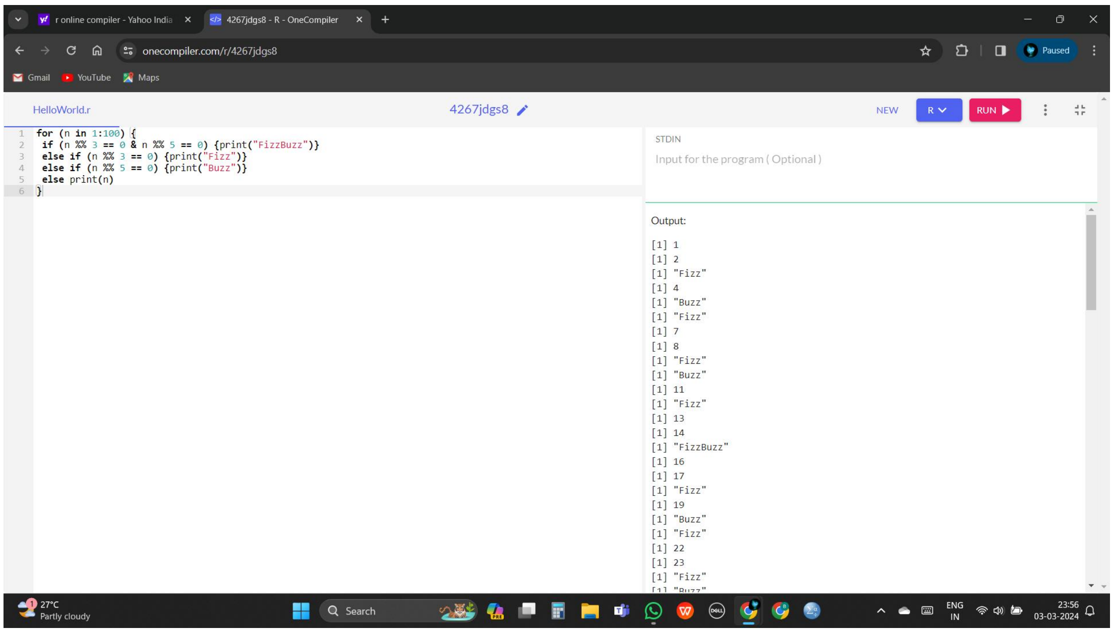


```r
prime_numbers <- function(n) {
if (n >= 2) {
 x = seq(2, n)
 prime_nums = c()
 for (i in seq(2, n)) {
 if (any(x == i)) {
 prime_nums = c(prime_nums, i)
 x = c(x[(x %% i) != 0], i)
 }
 }
 return(prime_nums)
 }
 else
 {
 stop("Input number should be at least 2.")
 }
 }
prime_numbers(12)
```

Output:

```
[1]  2  3  5  7 11
```

7. Write a R program to print the numbers from 1 to 100 and print "Fizz" for multiples of 3, print "Buzz" for multiples of 5, and print "FizzBuzz" for multiples of both.

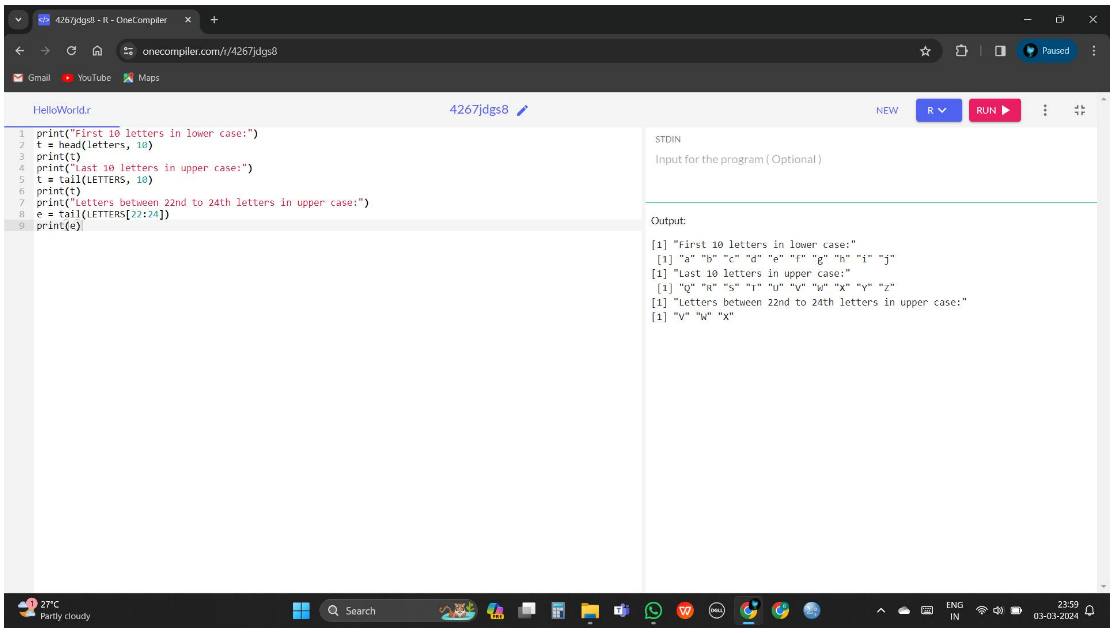


```r
for (n in 1:100) {
 if (n %% 3 == 0 & n %% 5 == 0) {print("FizzBuzz")}
 else if (n %% 3 == 0) {print("Fizz")}
 else if (n %% 5 == 0) {print("Buzz")}
 else print(n)
}
```

Output:

```
[1] 1
[1] 2
[1] "Fizz"
[1] 4
[1] "Buzz"
[1] "Fizz"
[1] 7
[1] 8
[1] "Fizz"
[1] "Buzz"
[1] 11
[1] "Fizz"
[1] 13
[1] 14
[1] "FizzBuzz"
[1] 16
[1] 17
[1] "Fizz"
[1] 19
[1] "Buzz"
[1] "Fizz"
[1] 22
[1] 23
[1] "Fizz"
```

8. Write a R program to extract first 10 English letters in lower case and last 10 letters in upper case and extract letters between 22nd to 24th letters in upper case.

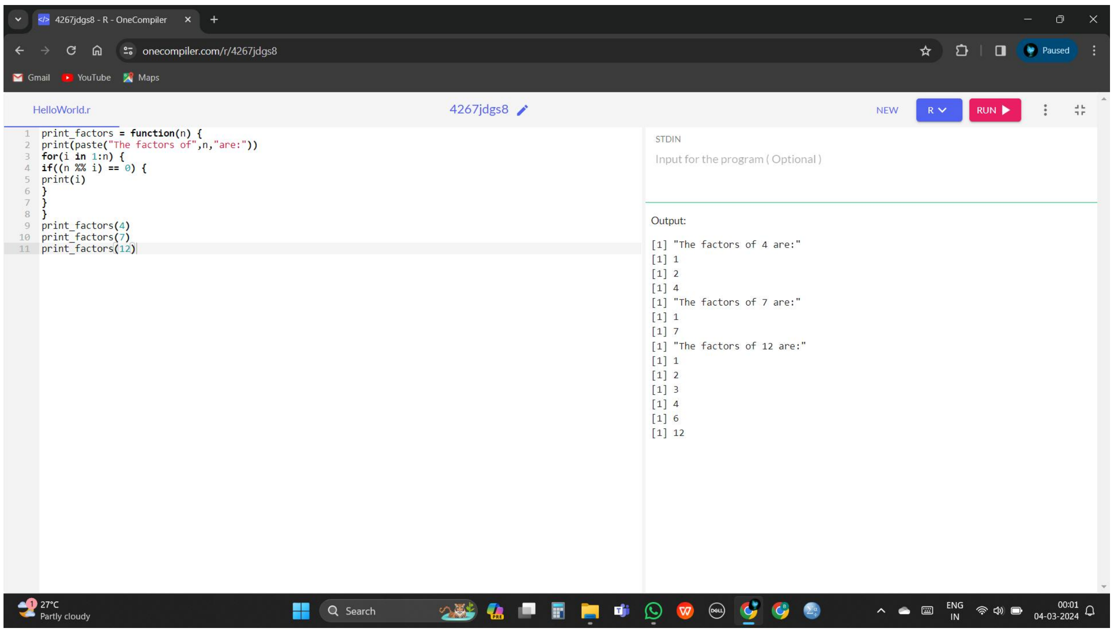


```r
print("First 10 letters in lower case:")
t = head(letters, 10)
print(t)
print("Last 10 letters in upper case:")
t = tail(LETTERS, 10)
print(t)
print("Letters between 22nd to 24th letters in upper case:")
e = tail(LETTERS[22:24])
print(e)
```

Output:

```
[1] "First 10 letters in lower case:"
 [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j"
[1] "Last 10 letters in upper case:"
 [1] "Q" "R" "S" "T" "U" "V" "W" "X" "Y" "Z"
[1] "Letters between 22nd to 24th letters in upper case:"
[1] "V" "W" "X"
```

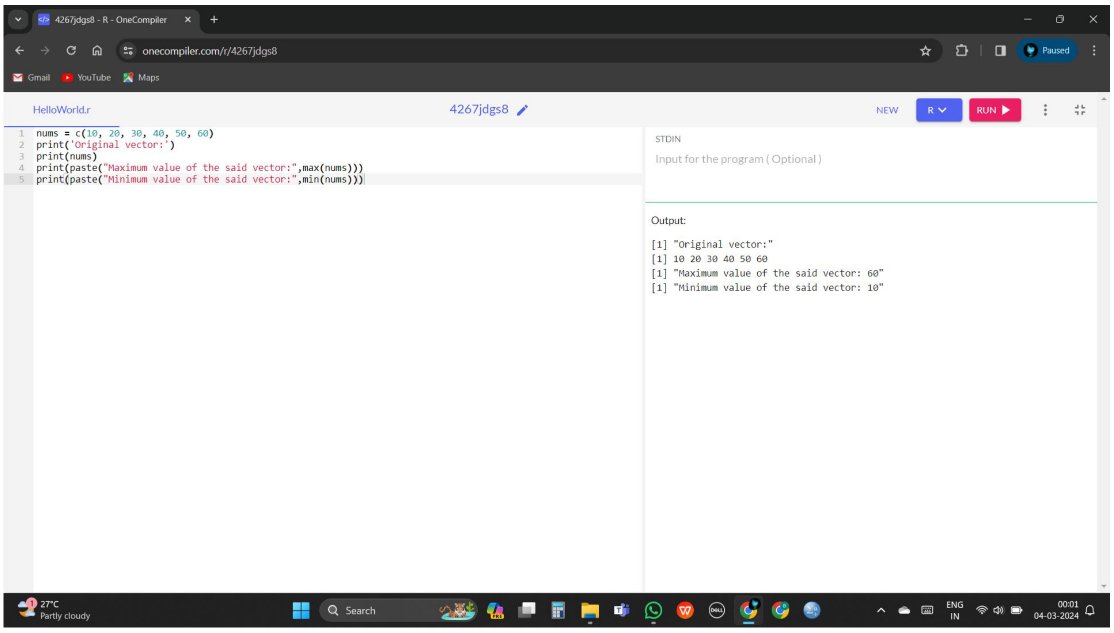
9. Write a R program to find the factors of a given number


4267jdgs8 - R - OneCompiler
onecompiler.com/r/4267jdgs8
HelloWorld.r
4267jdgs8
NEW
R
RUN
STDIN
Input for the program (Optional)
Output:

```r
print_factors = function(n) {
print(paste("The factors of",n,"are:"))
for(i in 1:n) {
if((n %% i) == 0) {
print(i)
}
}
}
print_factors(4)
print_factors(7)
print_factors(12)
```

```
[1] "The factors of 4 are:"
[1] 1
[1] 2
[1] 4
[1] "The factors of 7 are:"
[1] 1
[1] 7
[1] "The factors of 12 are:"
[1] 1
[1] 2
[1] 3
[1] 4
[1] 6
[1] 12
```

10. Write a R program to find the maximum and the minimum value of a given vector


4267jdgs8 - R - OneCompiler
onecompiler.com/r/4267jdgs8
HelloWorld.r
4267jdgs8
NEW
R
RUN
STDIN
Input for the program (Optional)
Output:

```r
nums = c(10, 20, 30, 40, 50, 60)
print('Original vector:')
print(nums)
print(paste("Maximum value of the said vector:",max(nums)))
print(paste("Minimum value of the said vector:",min(nums)))
```

```
[1] "Original vector:"
[1] 10 20 30 40 50 60
[1] "Maximum value of the said vector: 60"
[1] "Minimum value of the said vector: 10"
```