Семинар 1. Практикум

В качестве отчета по каждому пункту должен быть приведен листинг кода на R и, если требуется, скриншоты полученных результатов или ответы на вопросы.
Целью задания является освоить базовые команды языка R, которые понадобятся на протяжении всего курса. Ниже приведены примеры их использования и задания для закрепления. Для получения подробной справки по конкретной функции можно пользоваться командой **help(<имя функции>)** или любимой поисковой системой.

1. Базовые операции: присваивание

```
> n = 42
> x = 273.15
> name = "School of Data Analysis"
> flag = TRUE
```

2. Базовые операции: вывод всех текущих переменных

```
> ls()
[1] "flag", "n", "name", "x"
```

3. Базовые операции: удаление переменной

```
> rm(n)
> ls()
[1] "flag", "name", "x"
```

4. Базовые операции: простые вычисления

```
> a = 2
> b = 3
> b + a
[1] 5
> b - a
[1] 1
> b * a
[1] 6
> b / a
[1] 1.5
> b ^ a
[1] 9
```

5. Векторы и операции с ними

```
> a = c(1, 4, 5, 8)
> a
[1] 1 4 5 8
> b = 1:10
> b
 [1]  1  2  3  4  5  6  7  8  9 10
> d = seq(1, 5, by=0.5)
```

```
> d
[1] 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0
> c(1, 2, 3) + c(4, 5, 6)
[1] 5 7 9
> sqrt(c(100, 225, 400))
[1] 10 15 20
> d
[1] 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0
> d[3]
[1] 2
> d[5:7]
[1] 3.0 3.5 4.0
> d > 2.8
[1] FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
> d[d > 2.8]
[1] 3.0 3.5 4.0 4.5 5.0
> length(d)
[1] 9
```

Задание 1:
Создайте в R следующие векторы

```
a = (5, 10, 15, 20, ..., 160)
b = (87, 86, 85, ..., 56)
```

Перемножьте поэлементно эти векторы и сохраните результат в переменной d. Ответьте на вопросы:
(a) Чему равны 19-ый, 20-ый, 21-ый элементы d?
(b) Чему равны элементы d, меньшие 2000?
(c) Какое количество элементов d превосходит 6000?

6. Простейшие статистические процедуры

```
> 1:4
[1] 1 2 3 4
> sum(1:4)            # сумма
[1] 10
> prod(1:4) # произведение
[1] 24
> max(1:10)          # максимум
[1] 10
> min(1:10)           # минимум
[1] 1
> range(1:10)        # границы диапазона
[1]  1 10
> X <- rnorm(10)   # десять значений нормально распределенной случайной величины
> X
 [1]  0.2993040 -1.1337012 -0.9095197 -0.7406619 -1.1783715  0.7052832
 [7]  0.4288495 -0.8321391  1.1202479 -0.9507774
> mean(X)             # среднее
[1] -0.3191486
> sort(X)                # отсортированный вектор
 [1] -1.1783715 -1.1337012 -0.9507774 -0.9095197 -0.8321391 -0.7406619
 [7]  0.2993040  0.4288495  0.7052832  1.1202479
> median(X)           # медиана
[1] -0.7864005
> var(X)                  # дисперсия
[1] 0.739266
```

```
> sd(X)                    # среднеквадратичное отклонение
[1] 0.8598058
```

7. Матрицы и операции с ними

```
> A <- matrix(1:12, nr=3, nc=4)
> A
     [,1] [,2] [,3] [,4]
[1,]    1    4    7   10
[2,]    2    5    8   11
[3,]    3    6    9   12
> A[1:2,]
     [,1] [,2] [,3] [,4]
[1,]    1    4    7   10
[2,]    2    5    8   11
> A[,c(1,3)]
     [,1] [,2]
[1,]    1    7
[2,]    2    8
[3,]    3    9
> a = c(1,2,3)
> a
[1] 1 2 3
> b = c(10, 20, 30)
> b
[1] 10 20 30
> c = c(100, 200, 300)
> c
[1] 100 200 300
> d = c(1000, 2000, 3000)
> d
[1] 1000 2000 3000
> C <- cbind(a, b, c, d)
> C
     a  b   c    d
[1,] 1 10 100 1000
[2,] 2 20 200 2000
[3,] 3 30 300 3000
```

В векторах и матрицах все элементы должны иметь один и тот же тип, а реальные данные могут описываться признаками разных типов (строковые, логические, числовые). Поэтому для описания наборов данных используют датафреймы — таблицы, в которых различные столбцы могут иметь различные типы.

8. Датафреймы

```
> Name = c("Ivan", "Petr", "Fedor")
> Test1 = c(80, 95, 92)
> Test2 = c(40, 87, 90)
> grades = data.frame(Name, Test1, Test2)
> grades
   Name Test1 Test2
1  Ivan     80    40
2  Petr     95    87
3  Fedor  92    90
> grades$Test1
[1] 80 95 92
> mean(grades)
```

```
    Name    Test1    Test2
      NA 89.00000 72.33333
Warning message:
argument is not numeric or logical: returning NA in: mean.default(X[[1]], ...)
> mean(grades[,2:3])
   Test1    Test2
89.00000 72.33333
```

9. Чтение из файла

```
> iris = read.csv("Iris.csv",header=TRUE)    # возвращает датафрейм
```

Задание 2:
Загрузите в R набор данных из файла Iris.csv, ответьте на следующие вопросы:

1. Сколько объектов в этом наборе данных?
2. Сколькими признаками описан каждый объект?
3. Каковы названия признаков, описывающих данные? (см. функцию **colnames()**)
4. Каковы средние значения и дисперсии у числовых признаков?
5. Каковы средние значения и дисперсии у числовых признаков для объектов, принадлежащих виду Iris-setosa?

Для построения гистограмм в R используется функция **hist()**, для добавления еще одной гистограммы к уже существующим можно использовать параметр add=TRUE.

6. Постройте примеры гистограмм значений числовых признаков. Повыводите совместно гистограммы значений признаков объектов вида Iris-setosa и Iris-virginica. На основании значений какого признака проще всего разделить объекты этих двух видов?

Для построения графиков можно использовать функцию **plot()** в виде **plot(x,y)**.

7. Постройте график зависимости значений признака *petal_w* от значений признака *petal_l*. Каковы параметры линейной зависимости, наилучшим образом приближающей реальную зависимость? (воспользуйтесь функцией **lm()**)