**Задание 2.** **Загрузка данных из Интернет (Web-scraping) средствами специализированных программ и визуализация данных в R.**

Выполнил студент 2 курса

группы 09-715(1)

Санамян Артак.

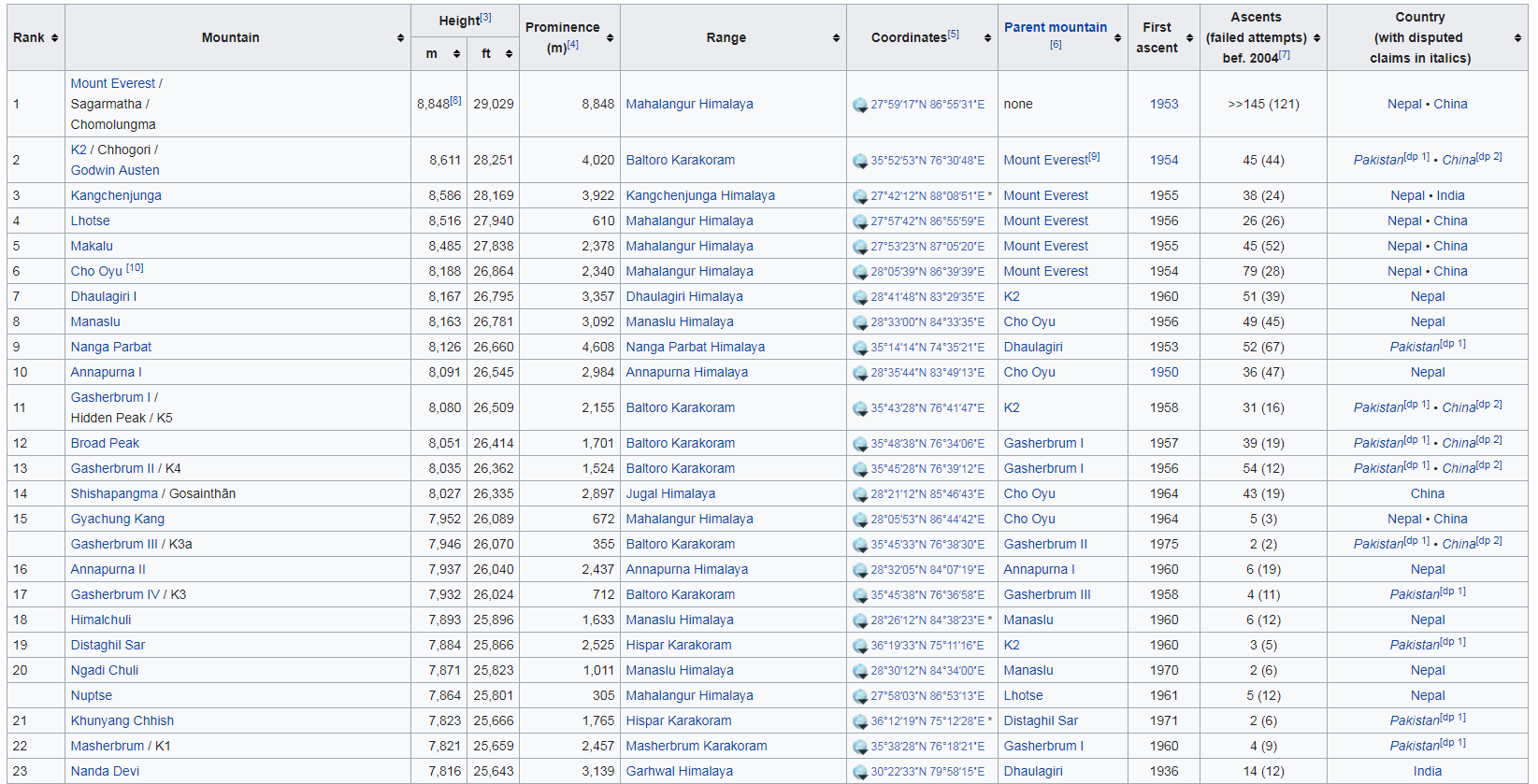
**Текст задания:**

В этом задании будем извлекать информацию из Интернет-страниц с помощью специализированных программ для web-скрэпинга. Несколько таких программ приведено по ссылке [Software for Web Scraping](http://scraping.pro/software-for-web-scraping/). Выбирайте, разумеется, бесплатные версии программ. Изучите интерфейс и функционал выбранной программы и загружайте данные в текстовый файл в формате csv или json или xml или каком-либо еще. Затем примените к загруженным данным какие-либо средства визуализации из R.

**Описание выполненной работы:**

Для анализа выбрана веб-страница <https://en.wikipedia.org/wiki/List_of_highest_mountains_on_Earth>, которая содержит информацию о высочайших горах мира, а именно

* Высота (метры, футы)
* Относительная высота
* Место расположения
* Координаты (широта, долгота)
* Родительская гора
* Год первого восхождения
* Количество восхождений (неудачных)
* Страна



В качестве системы для web-скрепинга будем использовать **Import.io**. Для начала работы необходимо зарегистрироваться на сайте <https://www.import.io/signup/>. После регистрации вы получите бесплатное пользование системой на 7 дней. Вводим url для анализа и переходим на вкладку редактирования. **Плюсы системы**, замеченные в процессе работы:

+)Щелкнув по ячейке таблицы, система спрашивает, нужно ли извлечь всю таблицу;

+)Возможность переименовать столбцы, удалить лишние или добавить новые;

+) Возможность применить ко всему столбцу регулярное выражение;

+)Возможность выгрузить извлеченные данные в трех форматах: Excel, CSV, NDJSON;

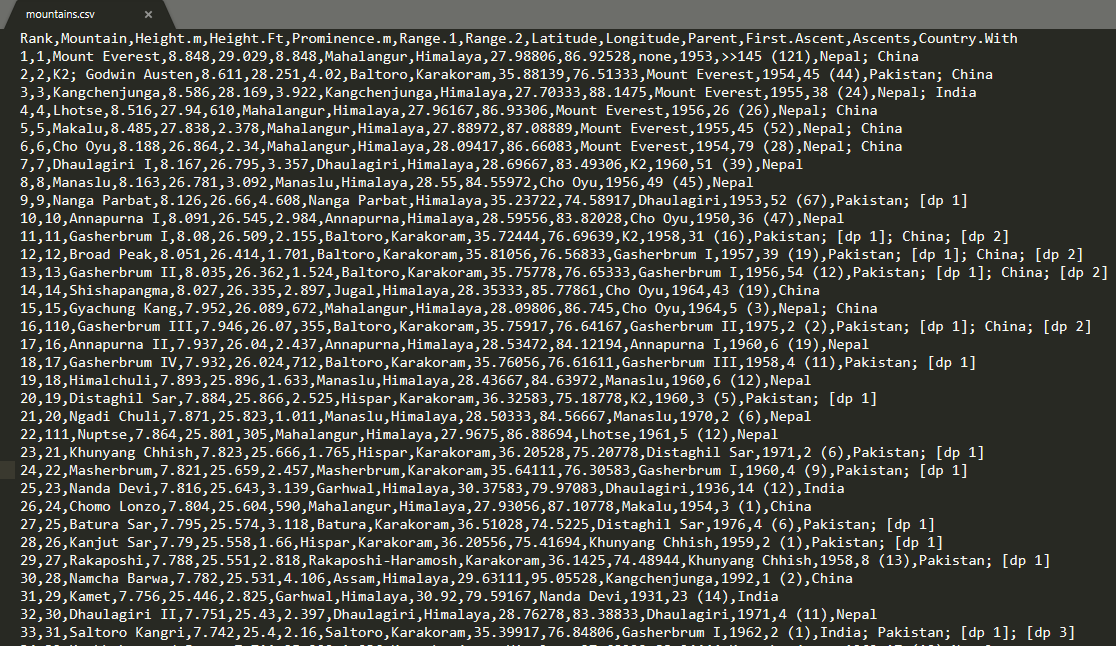
+)Возможность предварительного просмотра.

**Минусы системы**:

-)Невозможность анализировать страницу, в адресе которой содержатся русские буквы.

С помощью регулярных выражений из координаты были на два столбца: “Latitude” (Широта), “Longitude” (Долгота).

Загрузим полученные данные в файл [**mountains.csv**](https://e.mail.ru/attachment/15384004990000000343/0;1). Данные содержат информацию о высочайших горах.



В файле 119 записей, 13 столбцов:

* Rank – номер записи
* Mountain – наименование горы
* Height m – высота в метрах
* Height ft – высота в футах
* Prominence – относительная высота
* Range 1 – место расположения (хребет)
* Range 2 – место расположения (горная система)
* Latitude – широта
* Longitude – долгота
* Parent – родительская гора
* First Ascent – первое восхождение
* Ascents – количество восхождений, в скобках указано количество неудачных восхождений
* Country with – принадлежность странам

Используя данные о географических координатах гор, изобразим их места на карте мира с помощью точек, размер точек будет меняться в зависимости от роста горы, а также цвет точек будет меняться от желтого к красному.

Прочитаем данные из файла в переменную dataFrame:

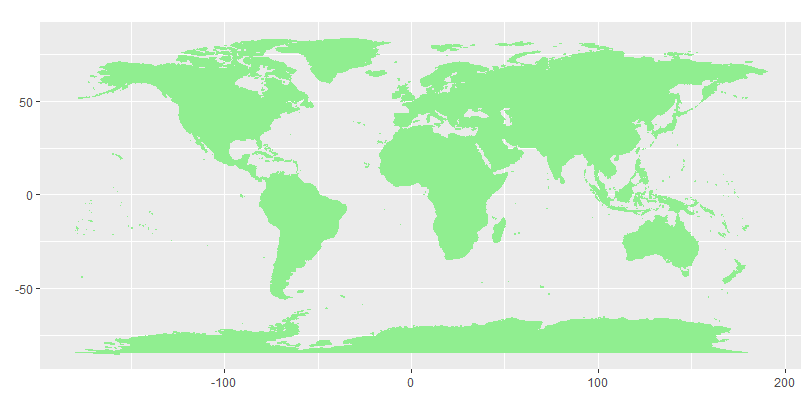
|  |
| --- |
| dataFrame = read.table("mountains.csv", header=TRUE, sep = ',') |

Для визуализации данных будем использовать библиотеку **ggplot2.**

|  |
| --- |
| library(ggplot2) |

Построим карту мира:

|  |
| --- |
| world\_map <- map\_data("world")  p <- ggplot() + coord\_fixed() +  xlab("") + ylab("")  world <- p + geom\_polygon(data=world\_map, aes(x=long, y=lat, group=group),  colour="light green", fill="light green") |



С помощью функции geom\_point нанесем места горных вершин на карту в виде точек. В параметре aes в качестве координаты x указываем dataFrame$Longitude (долгота), в качестве координаты y - dataFrame$Latitude (широта). Укажем параметр size=dataFrame$Height.m, чтобы размер точки менялся в зависимости от высоты горы, параметр fill=dataFrame$Height.m, чтобы цвет точки менялся в зависимости от высоты горы.

С помощью функции scale\_fill\_gradient указываем, что цвета точек меняются от желтого к красному, в зависимости от роста магнитуды:

* scale\_fill\_gradient(low = "yellow", high = "red")

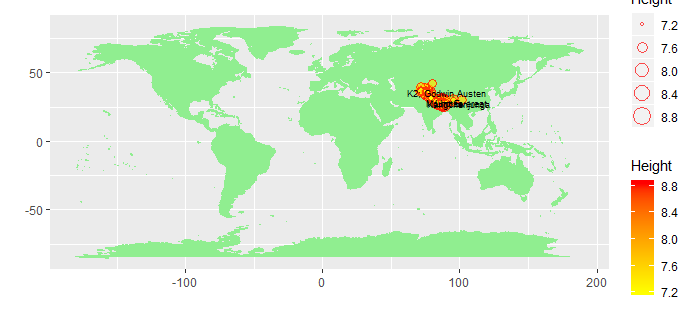
С помощью функции labs добавляем название легенды:

* labs(fill = "Height")
* labs(size = "Height")

Создадим массив text\_data, в котором будем хранить информацию только о тех горных вершинах, у которых высота >= 8.5km, и нанесем на карту названия мест с помощью функции geom\_text.

|  |
| --- |
| map\_data <-  world +  geom\_point(data=dataFrame,  aes(x=dataFrame$Longitude, y=dataFrame$Latitude, size=dataFrame$Height.m, fill=dataFrame$ Height.m), colour="red",  pch=21, alpha=I(0.7)) +  scale\_fill\_gradient(low = "yellow", high = "red") +  labs(fill = " Height") +  labs(size = " Height") +  geom\_text(aes(x=text\_data$V1, y=text\_data$V2, label=text\_data$V3),  hjust = 0.25,  nudge\_x = 0.05,  size=2.3  ) |

Результат визуализации:



**Код программы:**

# назначим рабочий директорий

setwd("C:/Users/admin.WIN-OSD1NEIENB0/Documents/Rprojects")

# прочитаем данные из файла в переменную dataFrame

# library(readxl)

# dataFrame = read\_excel("mountains.xlsx")

# dataFrame = read.table("mountains.txt", sep = "\t", header = TRUE)

# dataFrame = dataFrame[,-4]

# write.table(dataFrame, "mountains.csv", quote=FALSE, row.names=FALSE, col.names=TRUE, sep=',')

dataFrame = read.table("mountains.csv", header=TRUE, sep = ',')

print(dataFrame)

# число записей

nrow(dataFrame)

library(ggplot2)

# построим карту мира

world\_map <- map\_data("world")

p <- ggplot() + coord\_fixed() +

xlab("") + ylab("")

world <- p + geom\_polygon(data=world\_map, aes(x=long, y=lat, group=group),

colour="light green", fill="light green")

world

# data.frame для хранения информации только о тех записях, у которых

# высота >= 8.5

text\_data = data.frame();

count = 1;

for(i in 1:nrow(dataFrame))

{

if(dataFrame[i,3] >= 8.5)

{

text\_data[count, 1] = dataFrame[i, 9];

text\_data[count, 2] = dataFrame[i, 8];

text\_data[count, 3] = dataFrame[i, 2];

count = count+1;

}

}

text\_data

# наносим на карту горы

map\_data <-

world +

geom\_point(data=dataFrame,

aes(x=dataFrame$Longitude, y=dataFrame$Latitude, size=dataFrame$Height.m, fill=dataFrame$Height.m), colour="red",

pch=21, alpha=I(0.7)) +

scale\_fill\_gradient(low = "yellow", high = "red") +

labs(fill = "Height") +

labs(size = "Height") +

geom\_text(aes(x=text\_data$V1, y=text\_data$V2, label=text\_data$V3),

hjust = 0.25,

nudge\_x = 0.05,

size=2.3

)

map\_data