ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ

Ордена Трудового Красного Знамени

федеральное государственное бюджетное

образовательное учреждение высшего образования

«Московский Технический Университет Связи и Информатики» (МТУСИ)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Лабораторная работа №2

Выполнил:

Студент группы БСТ1701

Тешаев Парвиз

# Задание

Ниже можно найти данные об иммиграции для десяти ведущих стран. Инициализируйте 2D-массив в Java с этими данными. Затем, используя циклы, где это удобно, чтобы избежать ненужного набора текста, попросите вашу программу ответить на следующие вопросы или выполнить следующие задачи.

* Отобразите данные таблицы (не заголовки) на консоли, используя вкладки и / или System.out.printf () для выравнивания столбцов.
* Каково общее количество иммигрантов в этих странах?
* Какой общий процент иммигрантов в мире приходится на эти страны?
* В каких странах самый высокий и самый низкий процент иммигрантов?
* Основываясь на этих цифрах, оцените общую численность населения всех десяти стран.

Дополнительно:

* Прочтите все данные из текстового файла вместо жесткого задания параметров.
* Выведите эти данные, отсортированные по количеству иммигрантов в процентах от населения, сначала самые высокие.

| **Country** | **Number of Immigrants** | **Percentage of Total Immigrants in the World** | **Immigrants as Percentage of National Population** |
| --- | --- | --- | --- |
| United States | 45,785,090 | 19.8 | 14.3 |
| Russia | 11,048,064 | 4.8 | 7.7 |
| Germany | 9,845,244 | 4.3 | 11.9 |
| Saudi Arabia | 9,060,433 | 3.9 | 31.4 |
| United Arab Emirates | 7,826,981 | 3.4 | 83.7 |
| United Kingdom | 7,824,131 | 3.4 | 12.4 |
| France | 7,439,086 | 3.2 | 11.6 |
| Canada | 7,284,069 | 3.1 | 20.7 |
| Australia | 6,468,640 | 2.8 | 27.7 |
| Spain | 6,466,605 | 2.8 | 13.8 |

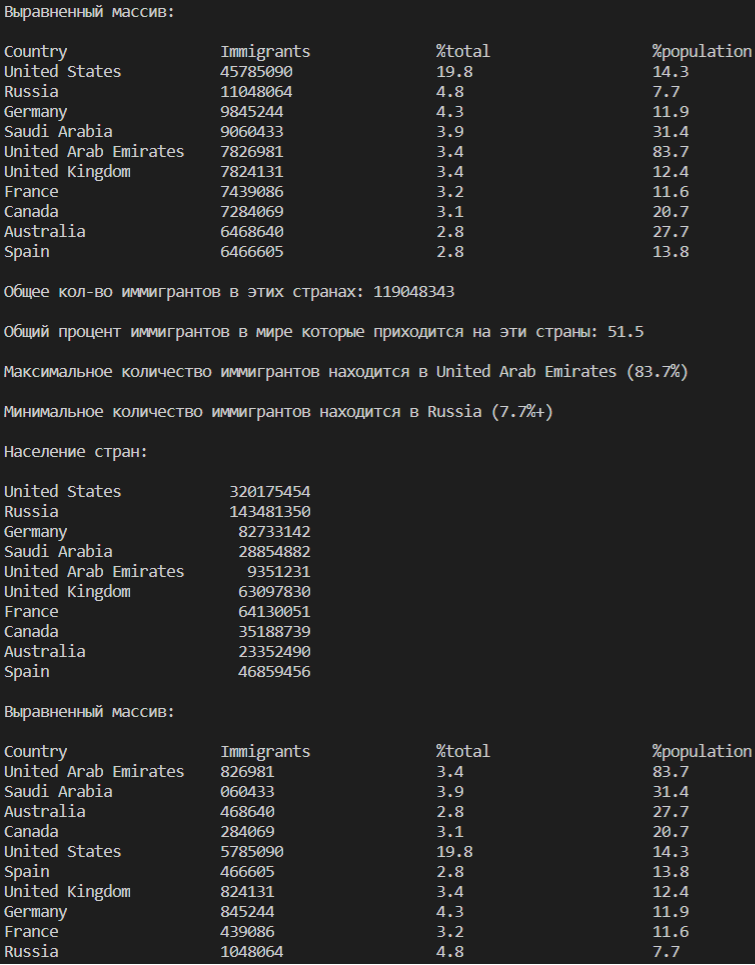
# Выполнение

На листинге 1 предоставлен код программы.

Листинг 1 – Код программы.

|  |
| --- |
| import java.io.FileReader;  import java.io.IOException;  public class Lab2 {      // функция для выравнивания и вывода столбцов      public static void print2dArr(String[][] arr) {          System.out.println("\nВыравненный массив:\n");          for (int i = 0; i < arr.length; i++) {              for (int j = 0; j < arr[i].length; j++)                  System.out.printf("%20s\t", arr[i][j]);              System.out.printf("\n");          }      }      // Функция для посчета общего кол-ва эмигрантов      public static void allImmQuant(String[][] arr) {          int sum = 0;          for (int i = 1; i < arr.length; i++)              sum += Integer.parseInt(arr[i][1]);          System.out.println("Общее кол-во иммигрантов в этих странах: " + sum);      }      // Функция для подсчета общего процента иммигрантов в мире которые приходится на      // эти страны      public static void percentImm(String[][] arr) {          double sum = 0;          for (int i = 1; i < arr.length; i++)              sum += Double.parseDouble(arr[i][2]);          System.out.println("Общий процент иммигрантов в мире которые приходится на эти страны: " + sum);      }      // Функция для нахождения в странах самого высокого и самого низккого процент      // иммигрантов      public static void percentMaxMin(String[][] arr) {          double max = Double.parseDouble(arr[1][3]);          double min = Double.parseDouble(arr[1][3]);          String maxStr = "", minStr = "";          for (int i = 2; i < arr.length; i++) {              if (max < Double.parseDouble(arr[i][3])) {                  max = Double.parseDouble(arr[i][3]);                  maxStr = arr[i][0];              } else if (min > Double.parseDouble(arr[i][3])) {                  min = Double.parseDouble(arr[i][3]);                  minStr = arr[i][0];              }          }          System.out.println("Максимальное количество иммигрантов в находится в " + maxStr + " (" + Double.toString(max)                  + "%" + ")\nМинимальное количество иммигрантов находится в " + minStr + " (" + Double.toString(min)                  + "%)");      }      // Фунция подсчета и вывода общей численности населения всех десяти стран.      public static void population(String[][] arr) {          System.out.println("Население стран:");          for (int i = 1; i < arr.length; i++) {              // Формула, для подсчета общей числености насиления              int populat = (int) (Integer.parseInt(arr[i][1]) / Double.parseDouble(arr[i][3]) \* 100);              // Вывод              System.out.printf("%20s\t%10d\n", arr[i][0], populat);          }      }      // Функция для чтения всех данных из текстового файла      public static String[][] read2dArr() {          // Новый 2D массив          String[][] arr = new String[10][4];          // Открытие фалйа          try (FileReader reader = new FileReader("Immigr.txt")) {              int ch, i = 0, j = 0;              while ((ch = reader.read()) != -1) {                  // Если встречается пробел в строке                  if (ch == 32 && j > 0) {                      j++;                      continue;                  } // Eсли конец строки                  else if (ch == 10) {                      arr[i][j] = arr[i][j].replaceAll("\\s+", "");                      i++;                      j = 0;                      continue;                  } // Если встречается цифра и это не первый столбец                  else if (ch >= 48 && ch <= 57 && j == 0)                      j++;                  // Если новая ячейка                  else if (arr[i][j] == null)                      arr[i][j] = "" + (char) ch;                  else // Если старая ячейка                      arr[i][j] += (char) ch;              }          } catch (IOException e) {              System.out.println(e.getMessage());          }          return arr;      }      //      public static String[][] sort2dArr(String arr[][]) {          String[][] arr2 = new String[arr.length + 1][arr[0].length];          arr2[0][0] = "Country";          arr2[0][1] = "Immigrants";          arr2[0][2] = "%total";          arr2[0][3] = "%population";          double max;          int cool = 0;          for (int i = 0; i < arr.length; i++) {              max = -1.0;              for (int j = 0; j < arr.length; j++) {                  if (max < Double.parseDouble(arr[j][3])) {                      max = Double.parseDouble(arr[j][3]);                      cool = j;                  }              }              //Заполнение нового массива              for (int k = 0; k < arr2[0].length; k++) {                  arr2[i + 1][k] = arr[cool][k];              }              arr[cool][3] = "-1";          }            return arr2;      }      public static void main(String[] args) {          //2D-массив входных данных          String[][] arr = { { "Country", "Immigrants", "%total", "%population" },                  { "United States", "45785090", "19.8", "14.3" }, { "Russia", "11048064", "4.8", "7.7" },                  { "Germany", "9845244", "4.3", "11.9" }, { "Saudi Arabia", "9060433", "3.9", "31.4" },                  { "United Arab Emirates", "7826981", "3.4", "83.7" }, { "United Kingdom", "7824131", "3.4", "12.4" },                  { "France", "7439086", "3.2", "11.6" }, { "Canada", "7284069", "3.1", "20.7" },                  { "Australia", "6468640", "2.8", "27.7" }, { "Spain", "6466605", "2.8", "13.8" } };          // Вызов функции для выравнивания и вывода столбцов          print2dArr(arr);          // Вызов функции для посчета общего кол-ва эмигрантов          allImmQuant(arr);          // Вызов функции для подсчета общего процента иммигрантов в мире которые приходится на эти страны          percentImm(arr);          // Вызов функции для нахождения в странах самого высокого и самого низккого процент иммигрантов          percentMaxMin(arr);          // Вывод фунции подсчета и вывода общей численности населения всех десяти стран.          population(arr);          // Запись в новый массив функции которая читатет данные из файла          String[][] arr2 = read2dArr();          //отсортировка нового массива          arr2 = sort2dArr(arr2);          //Вывод нового массива          print2dArr(arr2);      }  } |

На рисунке 1 предоставлен результат выполнения программы.



**Рисунок 1 – Результат работы программы**