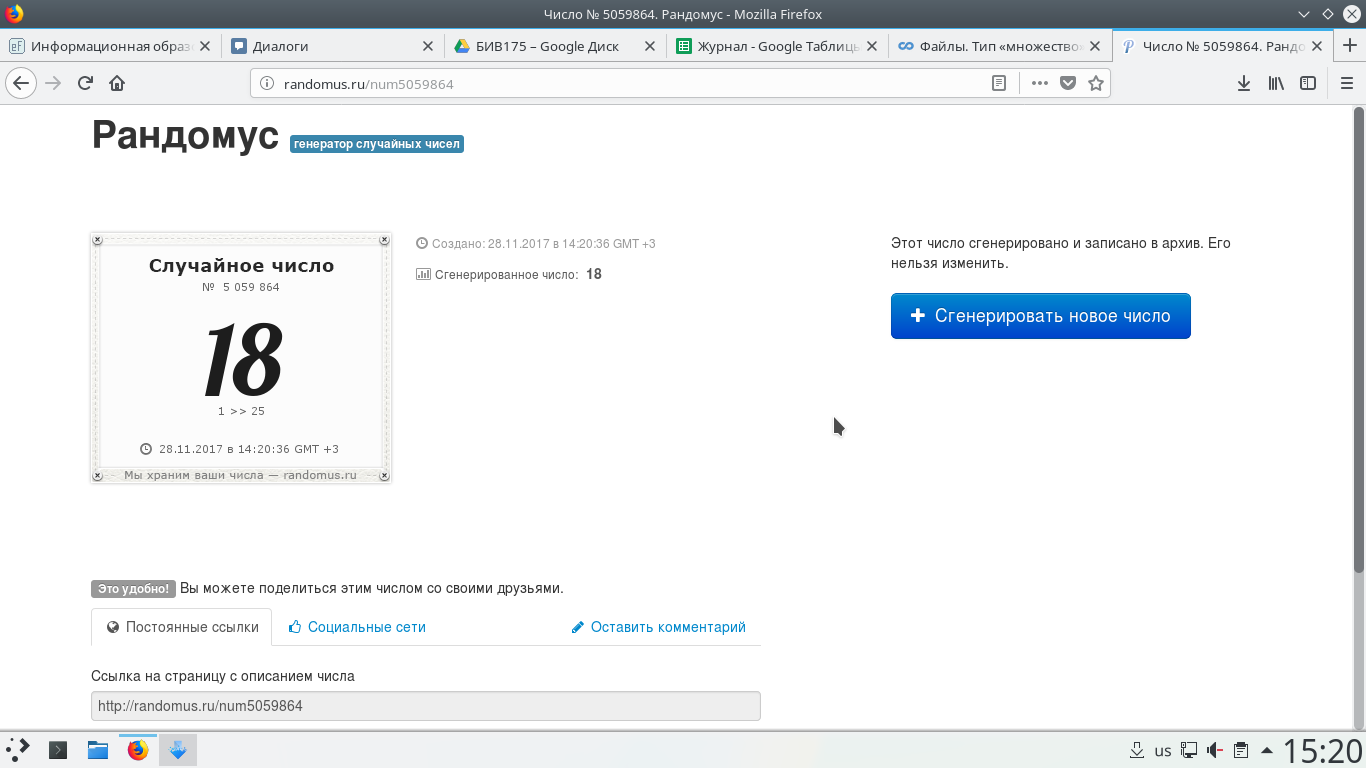
Coursera, peer-graded assignment for Week 10

# Task



Написать программу, которая в режиме запроса вводит записи и помещает их в файл. Сведения о телевизорах и книгах вводятся с новой строки для каждого поля.

После ввода записей программа осуществляет поиск следующей информации.

Названия самых дешевых книг автора N.

# Source code

|  |
| --- |
| *{$MODE DELPHI}*    **Program** Week10;  **Uses** sysutils;    **Const**  s\_size = 30;    **Type**  element = **Record**  book: **Record**  **name**, author: **string**[s\_size];  price, pages: **integer**;  **End**;  printed: **Record**  by, city: **string**[s\_size];  year: **Integer**;  **End**;  **End**;    Database = **file** **Of** element;    **Var**  FileName: **String**;  DB: Database;  rewrite\_flag: **Boolean**;    **Procedure** readbool(**Var** flag: **Boolean**);  *{Возвращает TRUE, если пользователь вводит Y, y или оставляет строку пустой}*  **Var** str: **String**;  **Begin**  readln(str);  flag := (str='Y') **Or** (str='y') **Or** (Length(str)=0);  **End**;    **Procedure** readSafeInt(**Var** int: **Integer**; param: **String**);  **Var**  err: **Integer**;  str: **String**;  **Begin**  readln(str);  val(str, int, err);  **While** (err<>0) **Do**  **Begin**  write('Дубина, введи '+ param +'. Это ЧИСЛО: ');  readln(str);  val(str, int, err)  **End**;  **End**;    **Procedure** readelement(**Var** cur: element);  *{Запрос информации об одной книге}*  **Var**  str: **String**;  **Begin**  **With** cur **Do**  **Begin**  **Repeat**  write('Название:');  readln(str);  **Until** length(str) <= s\_size;  book.**name** := str;  *{Ввод цены}*  write('Цена:');  readSafeInt(book.price, 'ЦЕНУ');  *{Ввод количества страниц}*  write('Число страниц:');  readSafeInt(book.pages, 'КОЛИЧЕСТВО СТРАНИЦ');  *{Ввод аатора}*  **Repeat**  write('Автор:');  readln(str);  **Until** length(str) <= s\_size;  book.author := str;  *{Информация об издании}*  writeln('Издание');  **Repeat**  write('Название:');  readln(str);  **Until** length(str) <= s\_size;  printed.by := str;  **Repeat**  write('Город:');  readln(str);  **Until** length(str) <= s\_size;  printed.city := str;  *{Год издания}*  write('Год:');  readSafeInt(printed.year, 'ГОД ИЗДАНИЯ');  **End**;  **End**;    *{Фцнкция ввода данных в базу}*  **Procedure** input\_data(**Var** DB: Database);  **Var**  continue : **Boolean**;  cur\_element : element;  **Begin**  **Repeat**  readelement(cur\_element);  write(DB, cur\_element);  writeln('Continue? Y/n');  readbool(continue);  **Until** continue=**false**;  **End**;    **Procedure** process(**Var** DB: Database);  **Var**  cur\_element : element;  **index**: **Integer**;  MinPrice: **Integer** = MaxInt;  **Begin**  **For** **index**:=1 **To** FileSize(DB) **Do**  **Begin**  read(DB, cur\_element);  **If** MinPrice > cur\_element.book.price **Then**  MinPrice := cur\_element.book.price;  **End**;  Reset(DB);  **For** **index**:=1 **To** FileSize(DB) **Do**  **Begin**  read(DB, cur\_element);  **If** MinPrice = cur\_element.book.price **Then**  Writeln(cur\_element.book.**name**);  **End**;  **End**;    *{Главная функция}*  **Begin**  write('Введите имя файла ');*{предложение ввести имя файла}*  readln(FileName);*{ввод имени файла}*  assign (DB, FileName);*{связь имени файла и переменной}*  **If** FileExists(FileName) **Then**  **Begin**  writeln('File exists. Do you want to rewrite it? Y/n');  readbool(rewrite\_flag);  **End**  **Else** rewrite\_flag := **true**;  **If** rewrite\_flag **Then**  **Begin**  rewrite(DB);*{открытие файла для записи}*  input\_data(DB); *{ввод данных в базу}*  close(DB); *{закрытие файла}*  **End**;  Reset(DB);  process(DB);  close(DB);  **End**. |

# Tests

# 

# 

# 