

ДЕНЬ 2В. Работа с файлами

2В.1. Прочитать данные из консоли и сформировать CSV-файл

Пользователь вводит записи, состоящие из нескольких полей. Как только он вводит пустую строку в первое из полей, ввод завершается. Необходимо сохранить записи в виде CSV-файла (табличные данные, в которых каждая строка — это строка таблицы, а колонки разделены запятыми или точкой с запятой)

Пример

Запись: фамилия (строка), возраст (целое число), результат (вещественное число).

Программа:

Введите данные:

Фамилия: Петров
Возраст: 20
Результат: 5.3

Фамилия: Иванов
Возраст: 22
Результат: 7.6

Фамилия: Сидоров
Возраст: 20
Результат: 5

Фамилия:

Введите путь для сохранения файла:
> data_new.csv

Выходной файл:

Петров, 20, 5.3
Иванов, 22, 7.6
Сидоров, 20, 5

Варианты

№	Запись
1	Производитель (строка), объем выпуска (вещественное число), средняя цена (вещественное число)
2	Название товара (строка), количество на складе (целое число), количество зарезервированных (целое число)
3	Фамилия (строка), число ролей (целое число), гонорар в млн. руб. (вещественное число)
4	Город (строка), улица (строка), номер дома (число), номер этажа (целое число)
5	Фамилия (строка), год поступления (целое число), средний балл (вещественное число)
6	Компания (строка), сумма поступлений в млн. руб. (вещественное число), сумма списаний в млн. руб. (вещественное число)
7	Адрес отправления (строка), адрес доставки (строка), вес (вещественное число)

8	Тема письма (строка), адресат (строка), есть ли вложения (логический тип), число слов (целое число)
9	Название (строка), число сезонов (целое число), год выпуска первого сезона (целое число)
10	Фамилия (строка), оценка за теорию (целое число), оценка за практику (целое число)
11	Автомобильный номер (строка), год выпуска (целое число), пробег в км (целое число)
12	Дисциплина (строка), номер курса (целое число), количество часов (целое число)
13	Название материала (строка), объем (вещественное число), вес (вещественное число)
14	Фамилия (строка), должность (строка), оклад в руб (целое число)
15	Номер заказа (строка), описание (строка), выполнен или нет (логический тип), сумма заказа (целое число)
16	Название цеха (строка), план выпуска деталей (целое число), фактический выпуск деталей (целое число)
17	Фамилия (строка), номер группы (строка), номер в группе (целое число), число выполненных заданий (целое число)
18	Номер телефона (строка), имя оператора (строка), баланс в копейках (целое число)
19	Адрес сайта (строка), число посетителей (целое число), число уникальных посетителей (целое число)
20	Фамилия (строка), рост (вещественное число), вес (вещественное число)

2В.2. Выполнить указанную операцию над набором чисел, записанных в файле

Программа запрашивает путь к файлу, в котором записаны числа, каждый на новой строке. Необходимо прочитать числа из файла и выполнить заданную операцию

Пример

Просуммировать числа из файла

Входной файл:

```
10
55.4
12.1
40
```

Программа:

```
Введите путь к файлу:
> digits.txt
```

Результат: 117.5

Варианты

№	Операция	№	Операция
1	Вычислить среднее значение	11	Разделить все числа на первое из чисел и вычислить сумму результатов

2	Просуммировать все числа, которые меньше, чем предыдущее число	12	Просуммировать все четные числа
3	Просуммировать дробные части всех чисел	13	Умножить числа на их порядковые номера и вычислить среднее
4	Перемножить все отрицательные числа	14	Вычислить среднее из квадратов чисел
5	Просуммировать квадраты всех чисел	15	У каждого второго числа поменять знак и просуммировать все числа
6	Умножить числа на их порядковые номера и сложить результаты	16	У каждого второго числа поменять знак и посчитать среднее
7	Просуммировать все нечетные числа	17	Перемножить все положительные числа
8	Перемножить дробные части всех чисел	18	Просуммировать все числа, которые больше первого из чисел
9	Просуммировать модули всех чисел	19	Извлечь квадратный корень из суммы квадратов чисел
10	Просуммировать все числа, которые больше, чем предыдущее число	20	Округлить числа и сложить результаты

2В.3. Прочитать все файлы из указанной директории и сформировать объединенный CSV-файл с учетом фильтра

Пользователь указывает путь до директории с исходными CSV файлами и путь до файла с результатом. Необходимо прочитать все исходные CSV-файлы в директории и записать в результирующий CSV-файл только строки, попадающие под заданное условие. В выходном файле помимо колонок исходного файла должна быть добавлена колонка с названием файла-источника.

Для получения списка файлов, см. *System.IO.Directory.GetFiles*

Пример

Запись: фамилия (строка), возраст (целое число), результат (вещественное число).
Вывести записи, у которых возраст меньше 21

Входной файл jan.csv:

```
Петров, 20, 5.3
Иванов, 22, 7.6
Сидоров, 20, 5
```

Входной файл feb.csv:

```
Потемкин, 19, 2.3
Пушкин, 32, 9.6
```

Входной файл mar.csv:

```
Журавлев, 18, 7.3
Никитин, 33, 8.6
Жаров, 22, 3.5
Семечкин, 55, 4
```

Программа:

Введите путь к директории:
> data/dir/

Введите путь для сохранения файла:
> report.csv

Выходной файл:

jan.csv, Петров, 20, 5.3
jan.csv, Сидоров, 20, 5
feb.csv, Потемкин, 19, 2.3
mar.csv, Журавлев, 18, 7.3

Варианты

№	Запись	Условие
1	Дисциплина (строка), номер курса (целое число), количество часов (целое число)	Количество часов < 40
2	Номер заказа (строка), описание (строка), выполнен или нет (логический тип), сумма заказа (целое число)	Сумма заказа > 800
3	Название материала (строка), объем (вещественное число), вес (вещественное число)	Объем < 10
4	Фамилия (строка), год поступления (целое число), средний балл (вещественное число)	Средний балл >= 4
5	Фамилия (строка), оценка за теорию (целое число), оценка за практику (целое число)	Среднее между оценками за теорию и практику >= 3.5
6	Фамилия (строка), должность (строка), оклад в руб (целое число)	Оклад > 25000
7	Адрес сайта (строка), число посетителей (целое число), число уникальных посетителей (целое число)	Число уникальных посетителей > 1000
8	Компания (строка), сумма поступлений в млн. руб. (вещественное число), сумма списаний в млн. руб. (вещественное число)	Разница между суммами поступления и списания > 0
9	Адрес отправления (строка), адрес доставки (строка), вес (вещественное число)	Вес > 500
10	Тема письма (строка), адресат (строка), есть ли вложения (логический тип), число слов (целое число)	Число слов > 200
11	Город (строка), улица (строка), номер дома (число), номер этажа (целое число)	Номер этажа > 2
12	Фамилия (строка), номер группы (строка), номер в группе (целое число), число выполненных заданий (целое число)	Число выполненных заданий >= 7
13	Фамилия (строка), число ролей (целое число), гонорар в млн. руб. (вещественное число)	Гонорар >= 1
14	Название (строка), число сезонов (целое число), год выпуска первого сезона (целое число)	Число сезонов > 2

15	Производитель (строка), объем выпуска (вещественное число), средняя цена (вещественное число)	Средняя цена > 100
16	Номер телефона (строка), имя оператора (строка), баланс в копейках (целое число)	Баланс < 0
17	Название товара (строка), количество на складе (целое число), количество зарезервированных (целое число)	Количество на складе < количества зарезервированных
18	Автомобильный номер (строка), год выпуска (целое число), пробег в км (целое число)	Пробег < 10000
19	Фамилия (строка), рост (вещественное число), вес (вещественное число)	Рост > 200
20	Название цеха (строка), план выпуска деталей (целое число), фактический выпуск деталей (целое число)	Значение плана выпуска деталей больше значения фактического выпуска деталей

2В.4. Разработка теста с вводом нескольких вариантов ответов и сохранением результатов в файл

Вам необходимо разработать программу для тестирования знаний на заданную тему, посвященную прикладному программированию и языку C#. Тест должен состоять из 3-ех вопросов, формулировку которых вам нужно придумать самостоятельно.

Перед началом теста программа запрашивает имя пользователя.

Далее программа должна выводить вопросы **в случайном порядке** (для этого используйте класс *Random*). После вывода вопроса программа предлагает на выбор 4 варианта ответа. При этом **правильными являются несколько из них** (от двух до трех).

Для того, чтобы ответить, пользователь вводит номера вариантов ответа через пробел. Пользователь может вводить числа в любом порядке. Если пользователь ввел одно и то же число несколько раз, то оно должно учитываться только однажды. Числа вне диапазона 1-4 должны игнорироваться.

После ввода вариантов пользователь нажимает клавишу *Enter*, и программа должна вывести либо следующий вопрос (если они не кончились), либо результат тестирования. Результат тестирования считается по следующей схеме:

Каждый ответ на вопрос оценивается от 0 до 1 балла:

- если пользователь ничего не вписал или выбрал только неверные ответы, он зарабатывает 0 баллов
- если пользователь выбрал хотя бы один верный ответ, то:
 - за каждый верный ответ он получает $1/R$ баллов, где R – количество верных ответов на этот вопрос
 - за каждый неверно вписанный ответ он получает -0.5 баллов штрафа, при этом оценка за вопрос не может стать меньше 0

После ответа на все вопросы число заработанных баллов делится на количество вопросов и выводится на экран в виде процентов.

Кроме вывода на экран, результат должен записывать в файл *“results.txt”* в формате: <Имя>: <Процент>. Каждая попытка прохождения теста записывается на новой строке.

Пример

Тема: переменные

Введите ваше имя:
> Misha

Вопрос 1. Какими из способов можно задать переменную в C#?

- 1) `int x;`
 - 2) `var y;`
 - 3) `var z = 5;`
 - 4) `x: int;`
- > 3 1

Вопрос 2. Что можно указать переменной ее при объявлении?

- 1) тип
 - 2) начальное значение
 - 3) имя
 - 4) дорогу
- > 1 2 4

Вопрос 3. Какие символы допустимы в названии переменной?

- 1) Тире
 - 2) Латинские буквы
 - 3) Цифры, если они стоят не в начале
 - 4) Знак доллара
- > 2

Ваш результат: 55.6%

Результат был рассчитан следующим образом:

- 1) полностью верно = 1 балл
- 2) 1 и 2 – верно, 3 - пропущено, 4 – нет = $1/3 + 1/3 - 0.5 \sim 0.167$ баллов
- 3) 2 – верно, 3 – пропущено = $1/2 = 0.5$ баллов

Итого: $(1 + 0.167 + 0.5) / 3 \sim 0.556$ или 55.6%

В файле *results.txt* добавится строка:

Misha: 55.6%

Варианты

№	Преобразование	№	Преобразование
1	Функции и методы в C#	11	Тестирование
2	Типы данных C#	12	Ключевые слова C#
3	Механизм событий	13	Объектно-ориентированное программирование
4	Обобщения (generics) в C#	14	Обработка файлов
5	Этапы разработки ПО	15	Операторы языка C#
6	Стандартная библиотека C#	16	Структуры в C#
7	UML-диаграммы	17	Отличия C# от C

8	Массивы в C#	18	Преобразование типов в C#
9	Классы в C#	19	Механизм исключений
10	Коллекции в C#	20	Форматирование вывода в C#

2В.5. Интерактивный выбор обрабатываемой строки текстового файла

Программа должна запросить путь к входному файлу, а затем очистить экран и вывести на него содержимое указанного файла.

Если строки файла не помещаются по ширине, то при выводе они должны обрезаться. Если файл не помещается по высоте, то он выводится постранично. Для того, чтобы узнать размер окна используйте *Console.WindowWidth* и *Console.WindowHeight*

При постраничном выводе предпоследняя строка экрана должна быть пустой, а последняя – содержать номер текущей страницы и общее число страниц в формате:
Страница 2 из 4

Одна из строк (при запуске – первая) является активной. Активная строка подсвечивается желтым цветом (используйте *Console.BackgroundColor*). Смена активной строки осуществляется с помощью стрелок клавиатуры.

Нажатие стрелки «Вниз» делает активной следующую строку. Если текущей строкой была последняя строка страницы, то должна отобразиться следующая страница (если есть) и активной стать первая строка.

Нажатие стрелки «Вверх» делает активной предыдущую строку. Если текущей строкой была первая строка страницы, то должна отобразиться предыдущая страница (если есть) и активной стать последняя строка.

Нажатие кнопки «Вправо» переключает файл на следующую страницу (если есть), активной становится первая строка.

Нажатие кнопки «Влево» переключает файл на предыдущую страницу (если есть), активной становится последняя строка

Нажатие кнопки «Пробел» преобразует текущую активную строку согласно варианту. Важно отметить, что преобразовываться должна вся строка целиком, а не только ее обрезанная при выводе часть. Если пользователь нажимает «Пробел» вновь, то строка должна быть преобразована вновь. Сохранять выполненное преобразование обратно в файл не нужно.

Нажатие кнопки «Esc» завершает работу программы

Варианты

№	Преобразование	№	Преобразование
1	Перевести в верхний регистр все согласные	11	Удалить все согласные
2	Удалить все слова без гласных букв	12	Удаление каждого второго слова

3	Перевести в верхний регистр все гласные	13	Перевернуть каждое слово в строке (абг деж -> гба жед)
4	Удалить все слова, в которых меньше 5 согласных	14	Перевести первую букву каждого слова в верхний регистр
5	Перевернуть строки (абг деж -> жед гба)	15	Удалить все слова размером меньше, чем из 5 букв
6	Удалить первое и последнее слова в строках	16	Удалить все слова без согласных букв
7	Перемешать в случайном порядке все слова в строке	17	Удалить знаки препинания
8	Перевести все буквы первого слова каждой строке в верхний регистр	18	Поменять местами слова в каждой паре слов: аб вг де жз -> вг аб жз де
9	Удалить все гласные	19	Удалить все слова размером больше, чем из 5 букв
10	Перевести в верхний регистр каждую вторую букву слов: абв гдеж -> аБв гДеЖ	20	Удалить все слова, в которых больше 5 согласных