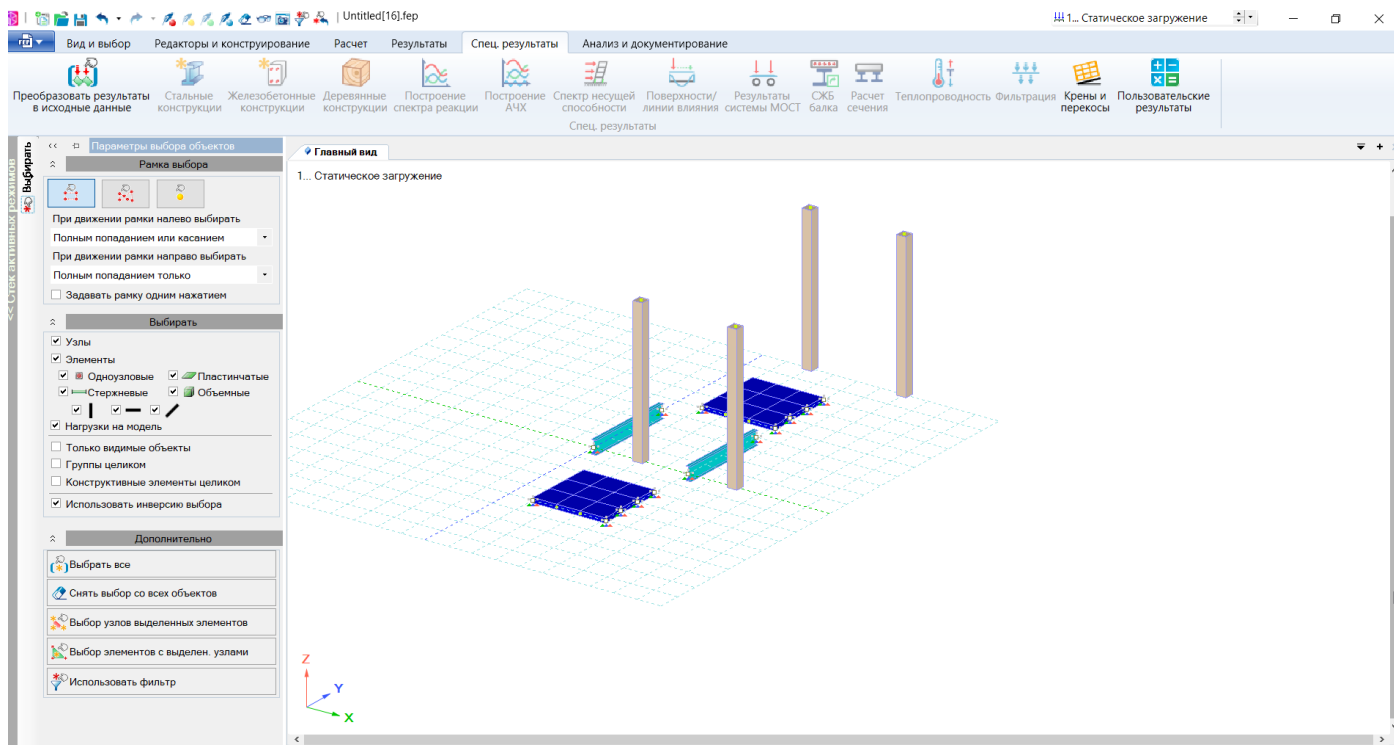


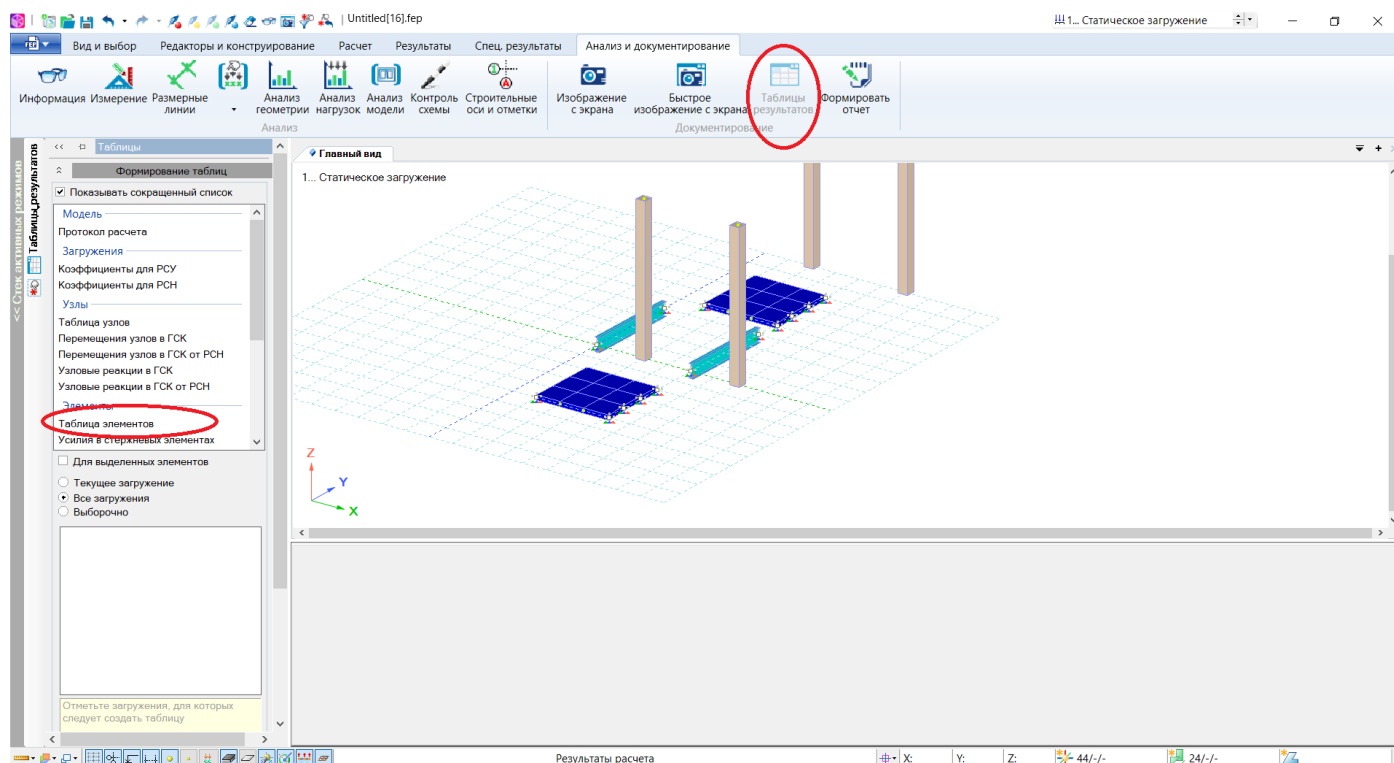
Пример использования системы «Пользовательские результаты» Лира 10.

Скрипт для вывода объемов материалов

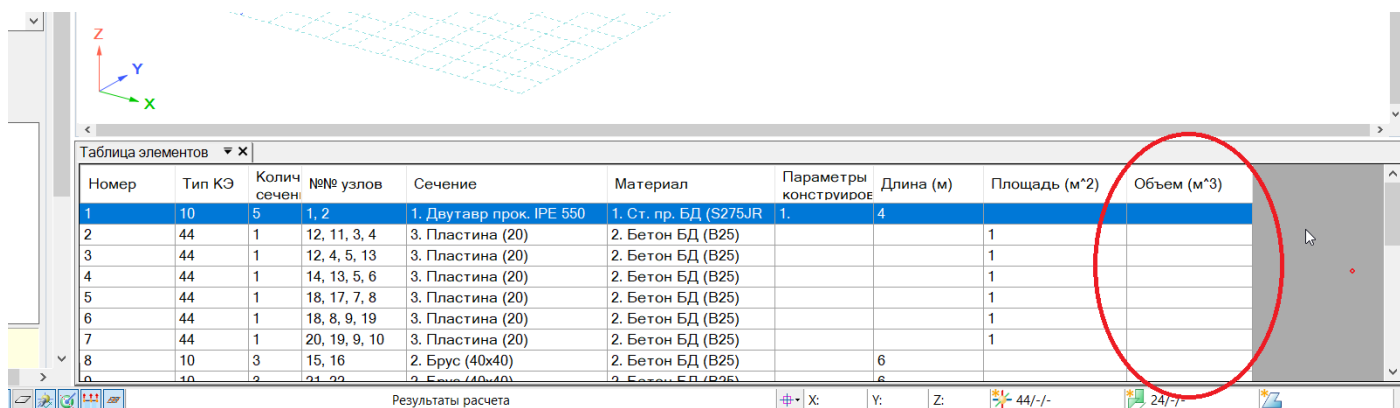
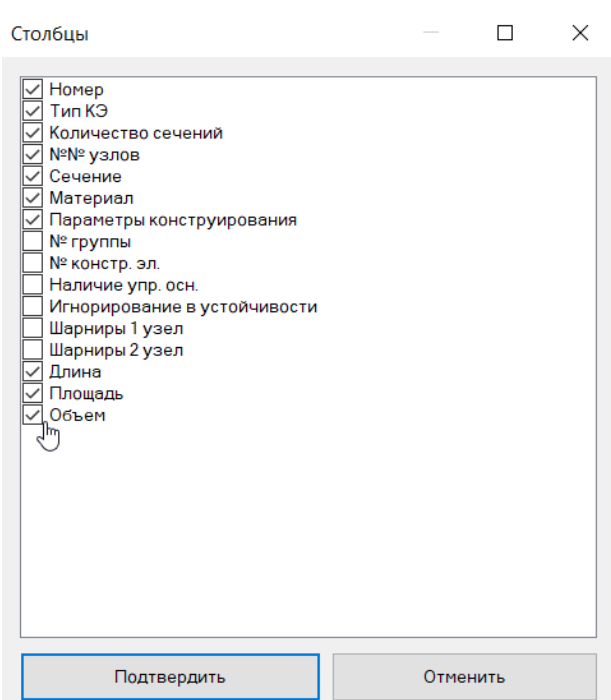
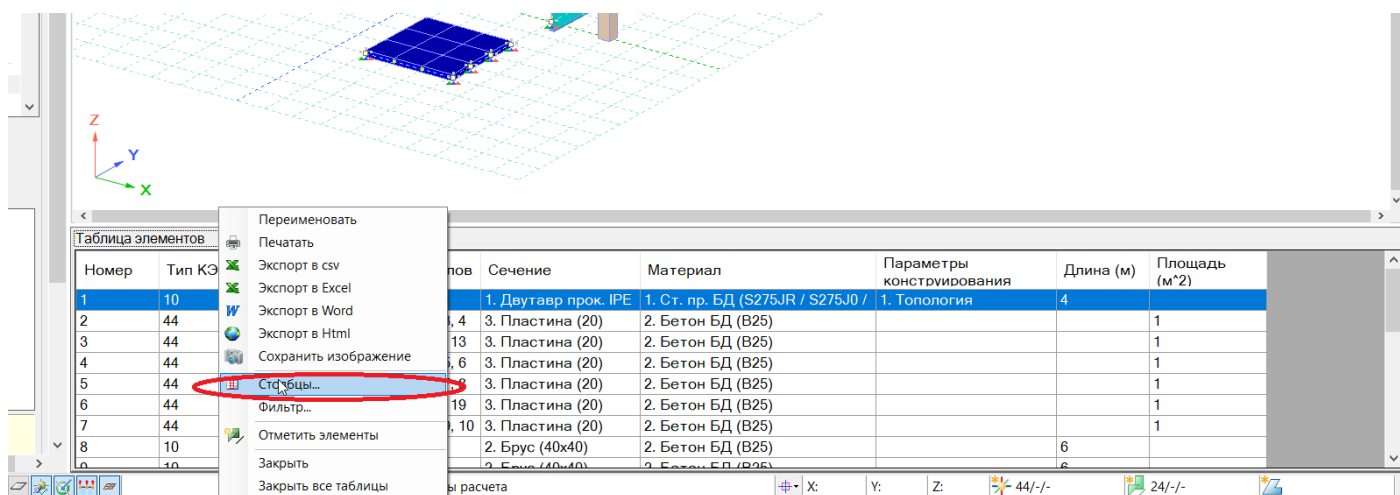
1. Задача. Вывод объемов материалов для пластинчатых и стержневых элементов



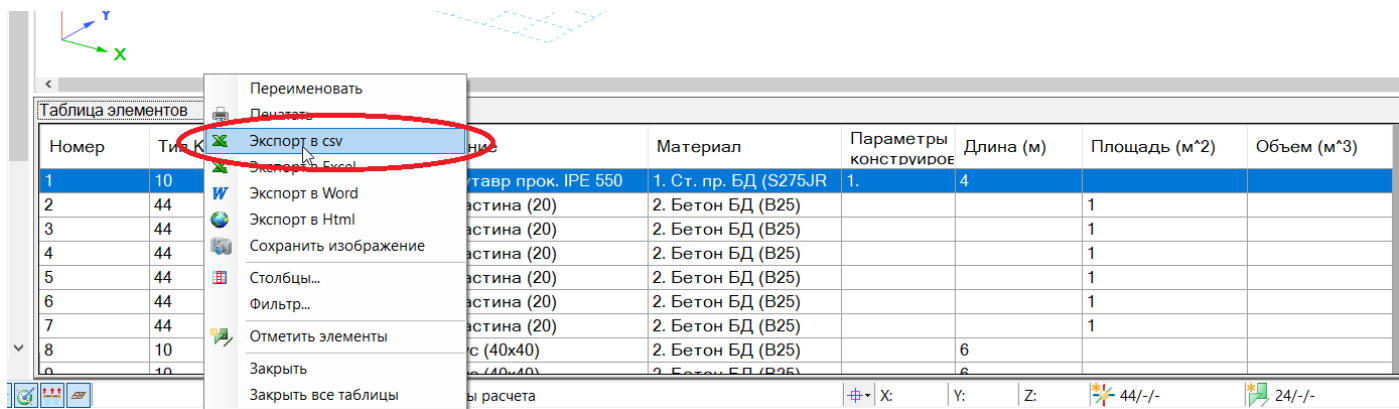
2. Выполните расчет, перейдите на вкладку результаты. Создайте таблицу элементов для всех или только для выделенных.



3. Добавьте столбец «Объем».



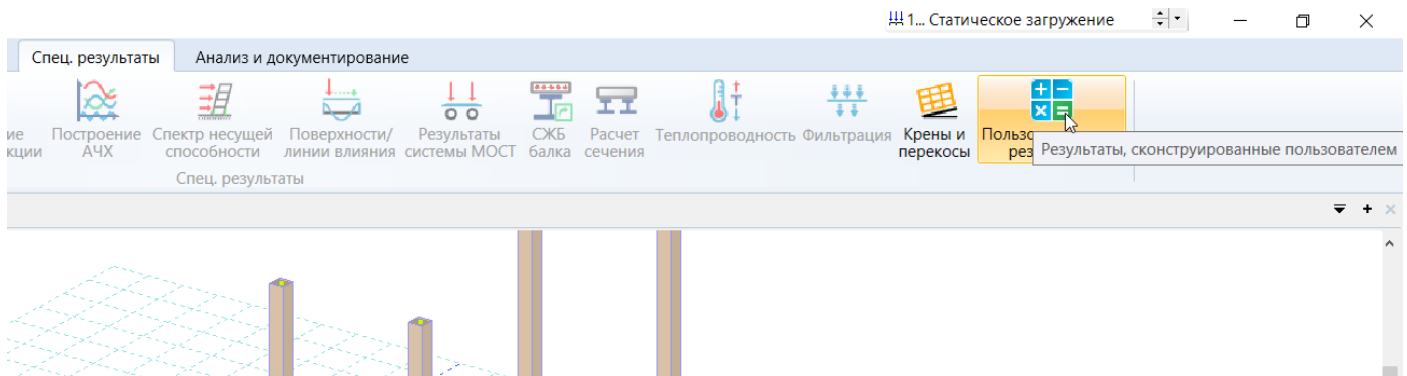
- На текущий момент автоматический вывод объемов элементов не реализован в Лира 10, но благодаря новой системе «Пользовательские результаты» можно дополнить таблицу элементов значениями объемов.
- Предварительно необходимо сделать экспорт таблицы в формат CSV.



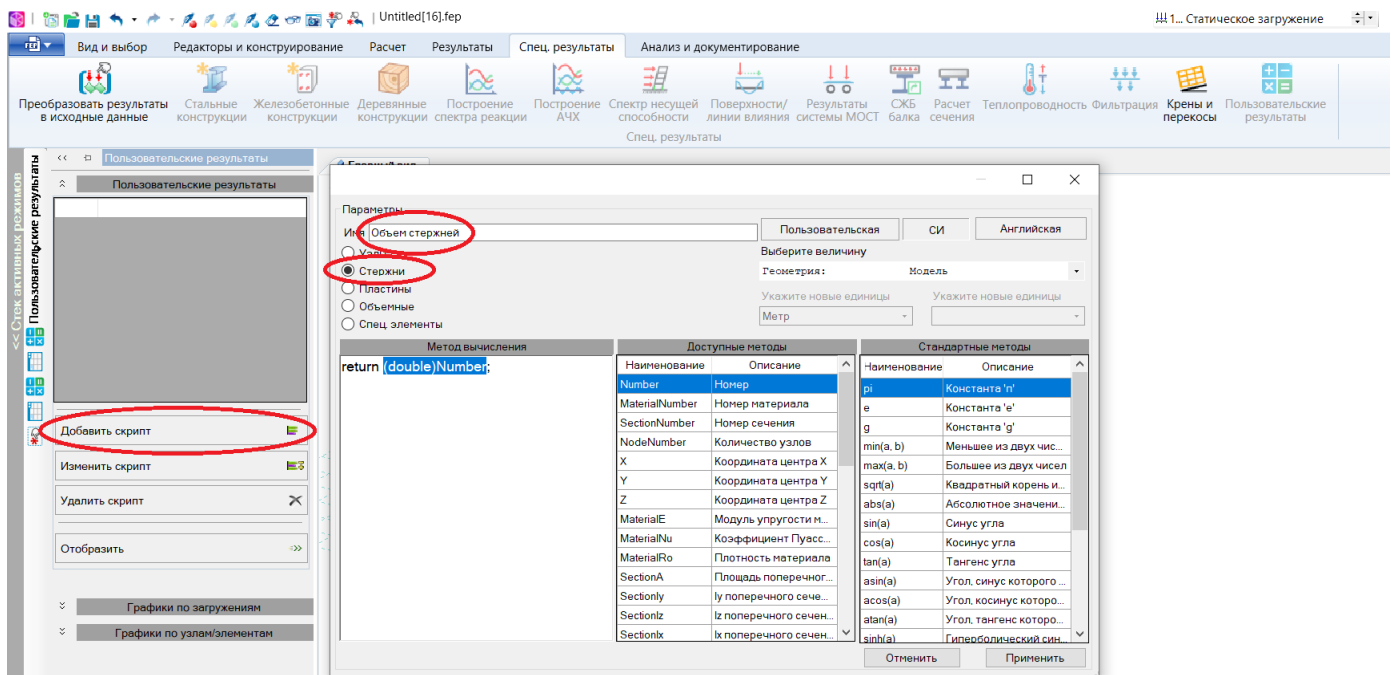
Формат CSV можно открыть в Excel. Закрываем Excel.

Автосохранение <input type="checkbox"/> <div>Таблица элементов.csv</div> <div>Поиск</div>											
Файл Главная Вставка Разметка страницы Формулы Данные Рецензирование Вид											
<div><div><div>Вставить</div><div>Буфер обмена</div></div><div><div>Calibri</div><div>11</div><div>A⁺ A⁻</div><div>Ж К Ч</div><div><div></div><div></div></div><div><div></div><div></div></div><div>Шрифт</div></div><div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div><div>Выравнивание</div></div><div><div>Текстовый</div><div><div>Число</div></div></div></div>											
<div><div>P20</div><div><div></div><div></div><div></div></div><div></div></div>											
	A	B	C	D	E	F	G	H	I	J	K
1	Номер	Тип КЭ	Количество	№ узло	Сечение	Материал	Параметр	Длина (м)	Площадь	Объем (м^3)	
2	1	10	5	1, 2	1. Двутавр	1. Ст. пр. БД (S275JR)	1. Топологический	4			
3	2	44	1	12, 11, 3, 4	3. Пластин	2. Бетон БД (B25)		1			
4	3	44	1	12, 4, 5, 13	3. Пластин	2. Бетон БД (B25)		1			
5	4	44	1	14, 13, 5, 6	3. Пластин	2. Бетон БД (B25)		1			
6	5	44	1	18, 17, 7, 8	3. Пластин	2. Бетон БД (B25)		1			
7	6	44	1	18, 8, 9, 19	3. Пластин	2. Бетон БД (B25)		1			
8	7	44	1	20, 19, 9, 13	3. Пластин	2. Бетон БД (B25)		1			
9	8	10	3	15, 16	2. Брус (40x40)	2. Бетон БД (B25)	6				
10	9	10	3	21, 22	2. Брус (40x40)	2. Бетон БД (B25)	6				
11	10	44	1	24, 23, 11, 3	3. Пластин	2. Бетон БД (B25)		1			
12	11	44	1	25, 24, 12, 3	3. Пластин	2. Бетон БД (B25)		1			
13	12	44	1	26, 25, 13, 3	3. Пластин	2. Бетон БД (B25)		1			
14	13	44	1	28, 27, 17, 3	3. Пластин	2. Бетон БД (B25)		1			
15	14	44	1	29, 28, 18, 3	3. Пластин	2. Бетон БД (B25)		1			
16	15	44	1	30, 29, 19, 3	3. Пластин	2. Бетон БД (B25)		1			
17	16	44	1	31, 24, 25, 3	3. Пластин	2. Бетон БД (B25)		1			
18	17	44	1	33, 32, 25, 3	3. Пластин	2. Бетон БД (B25)		1			
19	18	44	1	36, 28, 29, 3	3. Пластин	2. Бетон БД (B25)		1			
20	19	44	1	38, 37, 29, 3	3. Пластин	2. Бетон БД (B25)		1			
21	20	44	1	31, 39, 23, 3	3. Пластин	2. Бетон БД (B25)		1.05			
22	21	44	1	36, 40, 27, 3	3. Пластин	2. Бетон БД (B25)		1.05			
23	22	10	5	34, 35	1. Двутавр	1. Ст. пр. БД (S275JR)	1. Топологический	4			
24	23	10	3	41, 42	2. Брус (40x40)	2. Бетон БД (B25)	6				
25	24	10	3	43, 44	2. Брус (40x40)	2. Бетон БД (B25)	6				

6. Переход к пользовательским результатам



7. Создадим скрипт для вычисления объемов стержневых элементов



8. В окно метод вычисления вставляем следующий скрипт:

```
{
//
////
////////
string pathLog = @"B:\Таблица элементов.csv"; //путь к файлу (задает пользователь)
////////
////
//
string[] arrLine = System.IO.File.ReadAllLines(pathLog, System.Text.Encoding.GetEncoding("windows-1251")); //чтение файла
int[] ColumnName = ColumnVolumeNumber (arrLine[0]); //определяем подписи столбцов
//основной блок
for (int i = 1; i < arrLine.Length; i++)
{
    string[] str; //массив для хранения строки
    str = arrLine[i].Split(new char[] { ';' }); //запись строки в массив
    if (Convert.ToInt32(str[0]) == (Int32)Number && str[ColumnName[0]] != string.Empty && ColumnName[0] != 0) //если текущий элемент стержень
    {
        double SectionArea = SectionA; //площадь поперечного сечения
    }
}
```

```

        double LengthElement = Convert.ToDouble(strs[ColumnName[0]]); //длина элемента
        double Volume = System.Math.Round(LengthElement*SectionArea, 3); //объем элемента
        strs[ColumnName[2]]=Volume.ToString(); //запись объема в строку
        arrLine[i]=combineLine (strs); //обновление строки в общем массиве
    }
}
System.IO.File.WriteAllLines(pathLog, arrLine, System.Text.Encoding.GetEncoding("windows-1251"));
return (Int32)Number; //вывод номеров элементов
}
//поиск номера столбца с подписями
int[] ColumnVolumeNumber (string line)
{
    string[] firstLine = line.Split(new char[] { ';' }); //запись строки в массив
    int[] Columns = { 0, 0, 0 }; //массив для хранения столбцов - длина,площадь,объем
    for (int i = 0; i<firstLine.Length; i++)
    {
        if (firstLine[i]=="Длина (м)") //если найден столбец длина
        {
            Columns[0]=i; //номер столбца с длиной
        }
        if (firstLine[i]=="Площадь (м^2)") //если найден столбец площадь
        {
            Columns[1]=i; //номер столбца с площадью
        }
        if (firstLine[i]=="Объем (м^3)") //если найден столбец объем
        {
            Columns[2]=i; //номер столбца с объемом
        }
    }
    return Columns;
}
//сборка массива в строку
string combineLine (string[] baseMassive)
{
    string combiLine = string.Empty;
    for (int i = 0; i<baseMassive.Length; i++)
    {
        combiLine = combiLine+baseMassive[i]+";";
    }
    return combiLine;
}

```

Параметры

Имя

☐ Узлы
☒ Стержни
☐ Пластины
☐ Объемные
☐ Спец. элементы

Метод вычисления

```

{
//
////
/////
string pathLog = @"B:\Таблица элементов.csv"; //путь к файлу (задает пользователь)
/////
////
//
string[] arrLine = System.IO.File.ReadAllLines(pathLog, System.Text.Encoding.GetEncoding
("windows-1251")); //чтение файла
int[] ColumnName = ColumnVolumeNumber (arrLine[0]); //определяем подписи столбцов
//основной блок
for (int i = 1; i<arrLine.Length; i++)
{
string[] str; //массив для хранения строки
str = arrLine[i].Split(new char[] { ',' }); //запись строки в массив
if (Convert.ToInt32(strs[0])==(Int32)Number && strs[ColumnName[0]]!=string.Empty &&
ColumnName[0]!=0) //если текущий элемент стержень
  
```

Наименование

Number
MaterialNumber
SectionNumber
NodeNumber
X
Y
Z
MaterialE
MaterialNu
MaterialRo
SectionA
Sectionly
SectionIz
SectionIx
SectionP
getN (...)

В отмеченной строке указываем путь к файлу, который мы ранее экспортировали.

Отменить Применить

Нажимаем применить

9. Создадим скрипт для вычисления объемов пластинчатых элементов

Пользовательские результаты

1. Объем стержней

Добавить скрипт

Изменить скрипт

Удалить скрипт

Отобразить

Графики по загрузкам

Графики по узлам/элементам

Параметры

Имя

☐ Узлы
☒ Пластины
☐ Объемные
☐ Спец. элементы

Выберите величину

Геометрия:

Укажите новые единицы

Метр

Метод вычисления

```
return (double)Number;
```

Доступные методы

Наименование	Описание
Number	Номер
MaterialNumber	Номер материала
SectionNumber	Номер сечения
NodeNumber	Количество узлов
X	Координата центра X
Y	Координата центра Y
Z	Координата центра Z
MaterialE	Модуль упругости м...
MaterialNu	Коэффициент Пуасс...
MaterialRo	Плотность материала
SectionH	Толщина сечения
getNx(...)	Усилие Nx
getNy(...)	Усилие Ny
getNz(...)	Усилие Nz

Стандартные методы

Наименование	Описание
pi	Константа 'п'
e	Константа 'е'
g	Константа 'g'
min(a, b)	Меньшее из двух чис...
max(a, b)	Большее из двух чисел
sqrt(a)	Квадратный корень и...
abs(a)	Абсолютное значени...
sin(a)	Синус угла
cos(a)	Косинус угла
tan(a)	Тангенс угла
asin(a)	Угол, синус которого ...
acos(a)	Угол, косинус которо...
atan(a)	Угол, тангенс которо...
sinh(a)	Гиперболический син...

Отменить Применить

10. В окно метод вычисления вставляем следующий скрипт:

```

{
//
  
```

```

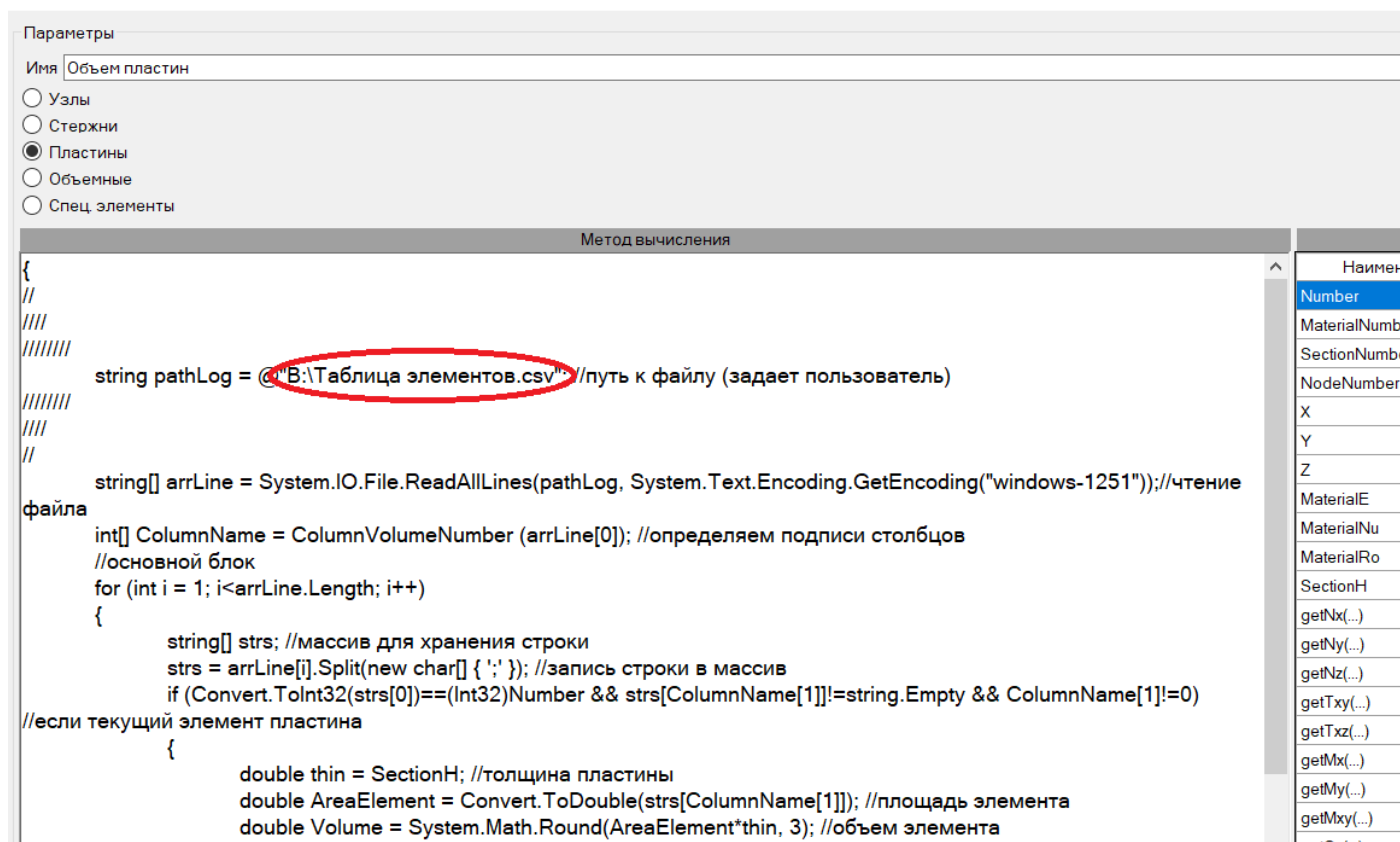
////
////////
string pathLog = @"B:\Таблица элементов.csv"; //путь к файлу (задает пользователь)
////////
////
//
string[] arrLine = System.IO.File.ReadAllLines(pathLog, System.Text.Encoding.GetEncoding("windows-
1251")); //чтение файла
int[] ColumnName = ColumnVolumeNumber (arrLine[0]); //определяем подписи столбцов
//основной блок
for (int i = 1; i<arrLine.Length; i++)
{
    string[] str; //массив для хранения строки
    str = arrLine[i].Split(new char[] { ';' }); //запись строки в массив
    if (Convert.ToInt32(strs[0])==(Int32)Number && str[ColumnName[1]]!=string.Empty &&
ColumnName[1]!=0) //если текущий элемент пластина
    {
        double thin = SectionH; //толщина пластины
        double AreaElement = Convert.ToDouble(strs[ColumnName[1]]); //площадь элемента
        double Volume = System.Math.Round(AreaElement*thin, 3); //объем элемента
        str[ColumnName[2]]=Volume.ToString(); //запись объема в строку
        arrLine[i]=combineLine (str); //обновление строки в общем массиве
    }
}
System.IO.File.WriteAllLines(pathLog, arrLine, System.Text.Encoding.GetEncoding("windows-1251"));
return (Int32)Number; //вывод номеров элементов
}
//поиск номера столбца с подписями
int[] ColumnVolumeNumber (string line)
{
    string[] firstLine = line.Split(new char[] { ';' }); //запись строки в массив
    int[] Columns = { 0, 0, 0 }; //массив для хранения столбцов - длина,площадь,объем
    for (int i = 0; i<firstLine.Length; i++)
    {
        if (firstLine[i]=="Длина (м)") //если найден столбец длина
        {
            Columns[0]=i; //номер столбца с длиной
        }
        if (firstLine[i]=="Площадь (м^2)") //если найден столбец площадь
        {
            Columns[1]=i; //номер столбца с площадью
        }
        if (firstLine[i]=="Объем (м^3)") //если найден столбец объем
        {
            Columns[2]=i; //номер столбца с объемом
        }
    }
    return Columns;
}
//сборка массива в строку
string combineLine (string[] baseMassive)
{
    string combiLine = string.Empty;

```

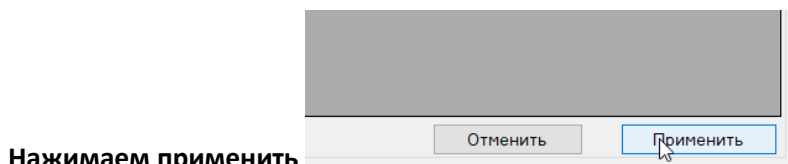
```

for (int i = 0; i < baseMassive.Length; i++)
{
    combiLine = combiLine + baseMassive[i] + ";";
}
return combiLine;
}

```

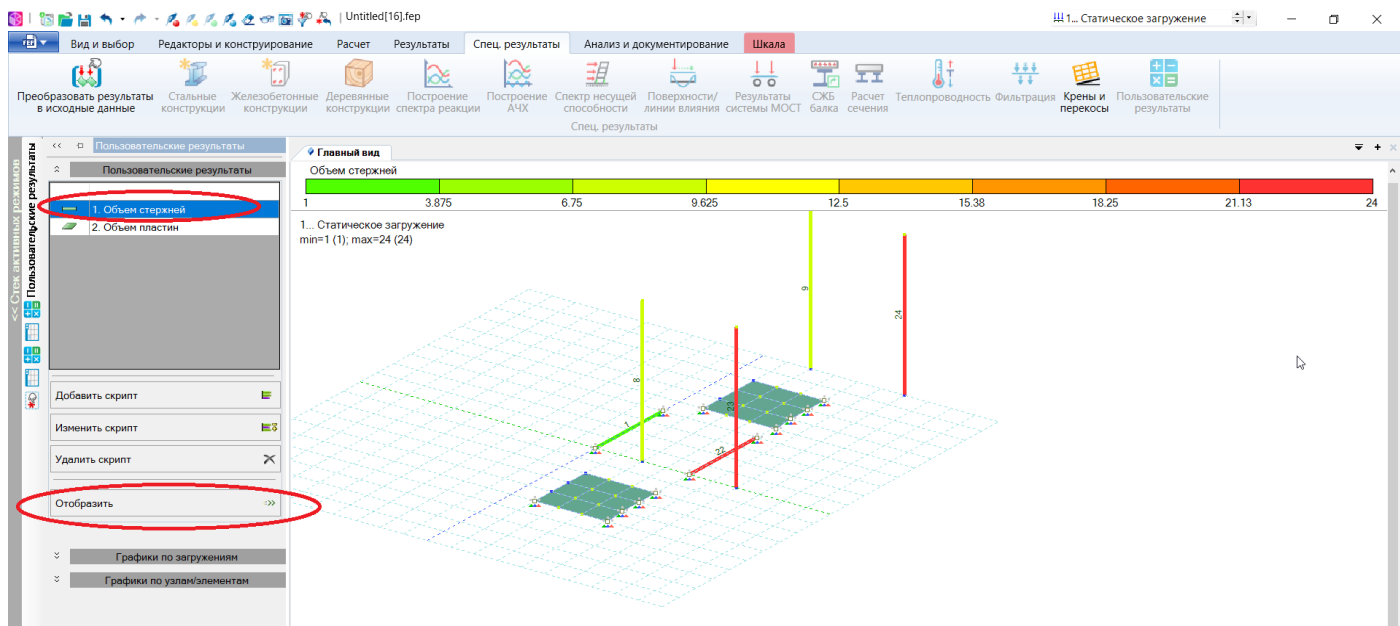


В отмеченной строке указываем путь к файлу, который мы ранее экспортировали.

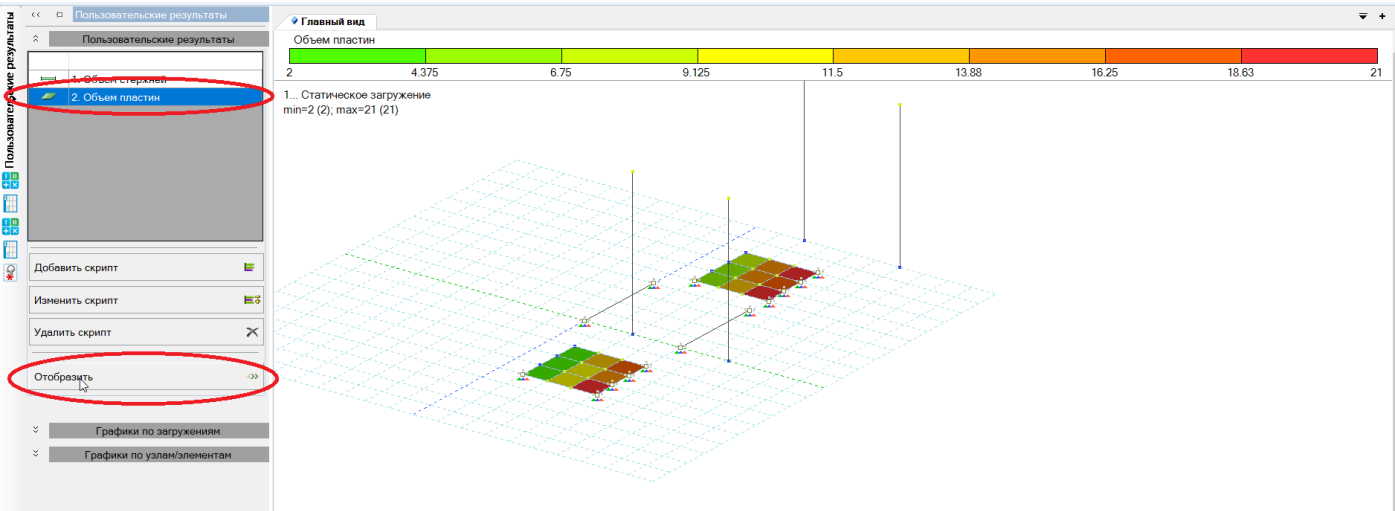


Нажимаем применить

11. Запустим скрипты



Если скрип сработает корректно, то на элементах возникнет подпись с номером.



12. Откроем файл Excel.

Автосохранение ☐ Таблица элементов.csv Поиск

Файл Главная Вставка Разметка страницы Формулы Данные Рецензирование Вид

Вставить Шрифт Выравнивание Число

Буфер обмена

N21

	A	B	C	D	E	F	G	H	I	J	K
1	Номер	Тип КЭ	Количество	№ узло	Сечение	Материал	Параметр	Длина (м)	Площадь	Объем (м³)	
2	1	10	5	1, 2	1. Двутавр	1. Ст. пр. I	1. Тополо	4		0.054	
3	2	44	1	12, 11, 3, 4	3. Пластин	2. Бетон БД (B25)				1 0.2	
4	3	44	1	12, 4, 5, 13	3. Пластин	2. Бетон БД (B25)				1 0.2	
5	4	44	1	14, 13, 5, 6	3. Пластин	2. Бетон БД (B25)				1 0.2	
6	5	44	1	18, 17, 7, 8	3. Пластин	2. Бетон БД (B25)				1 0.2	
7	6	44	1	18, 8, 9, 15	3. Пластин	2. Бетон БД (B25)				1 0.2	
8	7	44	1	20, 19, 9, 13	3. Пластин	2. Бетон БД (B25)				1 0.2	
9	8	10	3	15, 16	2. Брус (40	2. Бетон БД (B25)		6		0.96	
10	9	10	3	21, 22	2. Брус (40	2. Бетон БД (B25)		6		0.96	
11	10	44	1	24, 23, 11, 3	3. Пластин	2. Бетон БД (B25)				1 0.2	
12	11	44	1	25, 24, 12, 3	3. Пластин	2. Бетон БД (B25)				1 0.2	
13	12	44	1	26, 25, 13, 3	3. Пластин	2. Бетон БД (B25)				1 0.2	
14	13	44	1	28, 27, 17, 3	3. Пластин	2. Бетон БД (B25)				1 0.2	
15	14	44	1	29, 28, 18, 3	3. Пластин	2. Бетон БД (B25)				1 0.2	
16	15	44	1	30, 29, 19, 3	3. Пластин	2. Бетон БД (B25)				1 0.2	
17	16	44	1	31, 24, 25, 3	3. Пластин	2. Бетон БД (B25)				1 0.2	
18	17	44	1	33, 32, 25, 3	3. Пластин	2. Бетон БД (B25)				1 0.2	
19	18	44	1	36, 28, 29, 3	3. Пластин	2. Бетон БД (B25)				1 0.2	
20	19	44	1	38, 37, 29, 3	3. Пластин	2. Бетон БД (B25)				1 0.2	
21	20	44	1	31, 39, 23, 3	3. Пластин	2. Бетон БД (B25)			1.05	0.21	
22	21	44	1	36, 40, 27, 3	3. Пластин	2. Бетон БД (B25)			1.05	0.21	
23	22	10	5	34, 35	1. Двутавр	1. Ст. пр. I	1. Тополо	4		0.054	
24	23	10	3	41, 42	2. Брус (40	2. Бетон БД (B25)		6		0.96	
25	24	10	3	43, 44	2. Брус (40	2. Бетон БД (B25)		6		0.96	
26											

В результате работы скриптов выполнено заполнение столбца с объемами.