

Функциональное и логическое программирование
Лекция №8

Фреймы

+ csv

—

Беляков А.Ю.

Фрейм

phpMyAdmin



(Недавние таблицы) ...

gb_mytest

- groups
- initial
- players
- results
- students
- tests

Создать таблицу

mysql94.1gb.ru » gb_mytest » students

Обзор

Структура

SQL

Поиск

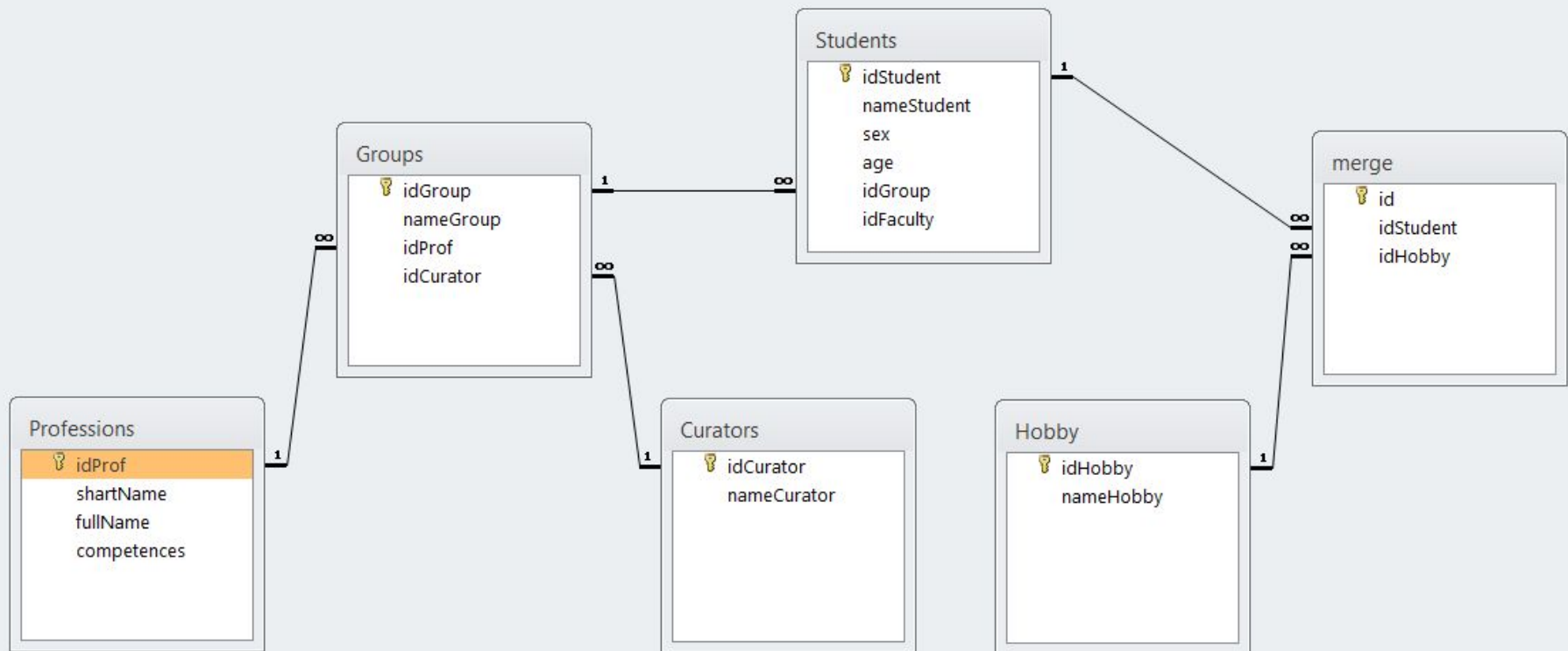
Вставить

		idstud	name1	name2	namegr	password
<input type="checkbox"/>		81	Батуев	Андрей	8	FT3h
<input type="checkbox"/>		82	Бахтиаров	Матвей	8	a1ro
<input type="checkbox"/>		83	Березин	Илья	8	4MWm
<input type="checkbox"/>		84	Каверзин	Александр	8	SnJZ
<input type="checkbox"/>		85	Калинин	Вадим	8	YAMb
<input type="checkbox"/>		86	Колчанов	Данила	8	YGJo
<input type="checkbox"/>		87	Коновалова	Элина	8	GHob
<input type="checkbox"/>		88	Коньков	Дмитрий	8	8uuE
<input type="checkbox"/>		89	Копытов	Вадим	8	iT5z
<input type="checkbox"/>		90	Куликов	Денис	8	IG7G

Как
хранить
фреймы?

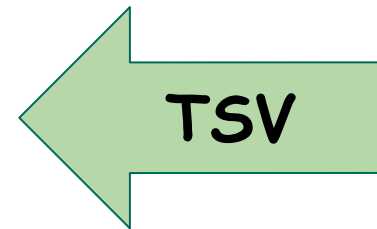
txt
array json
csv
excel

Схема данных

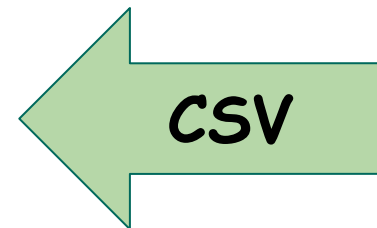


txt -> CSV

```
1 ● id nameSt sex age idGroup
2 ● 1 Иванов 1 22 1
3 ● 2 Петров 1 17 1
4 ● 3 Сидов 1 17 2
5 ● 4 Ушкина 0 20 2
6 ● 5 Берг 0 21 1
7 ● 6 Немов 1 23 2
8 ● 7 Мамина 0 17 2
```



```
9
10
11 ● id,nameSt,sex,age,idGroup
12 ● 1,Иванов,true,22,1
13 ● 2,Петров,true,17,1
14 ● 3,Сидов,true,17,2
15 ● 4,Ушкина,true,20,3
16 ● 5,Берг,true,21,3
17 ● 6,Немов,true,23,2
18 ● 7,Мамина,true,17,2
```



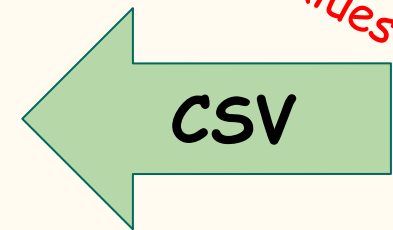
txt -> CSV

```
1997,Ford,E350,"ac, abs, moon",3000.00
1999,Chevy,"Venture «Extended Edition»","",4900.00
1996,Jeep,Grand Cherokee,"MUST SELL! air, moon roof, loaded",4799.00
```

Результирующая таблица:

1997	Ford	E350	ac, abs, moon	3000
1999	Chevy	Venture «Extended Edition»		4900
1996	Jeep	Grand Cherokee	MUST SELL! air, moon roof, loaded	4799

Comma-Separated Values



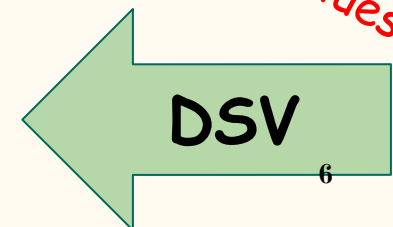
```
1965;Пиксель;E240 – формальдегид (опасный консервант)!;"красный, зелёный, битый";"3000,00"
1965;Мышка;"А правильной использовать ""Ёлочки""";"4900,00"
"Н/д";Кнопка;Сочетания клавиш;"MUST USE! Ctrl, Alt, Shift";"4799,00"
```

Результирующая таблица:

; <- Excel

1965	Пиксель	E240 — формальдегид (опасный консервант)!	красный, зелёный, битый	3000
1965	Мышка	А правильной использовать "Ёлочки"		4900
Н/д	Кнопка	Сочетания клавиш	MUST USE! Ctrl, Alt, Shift	4799

Delimiter-Separated Values



Читать #1

```
id,nameSt,sex,age,idGroup
1,Иванов,true,22,1
2,Петров,true,17,1
3,Сидов,true,17,2
4,Ушкина,true,20,3
5,Берг,true,21,3
6,Немов,true,23,2
7,Мамина,true,17,2
```

```
1
2
3 const fs = require('fs');
4 const fastcsv = require('fast-csv');
5
6 fs
7   .createReadStream('data.csv')
8   .pipe(fastcsv.parse({ headers: true }))
9   .on('data', row => console.log(row));
```

```
10
11 { id: '1', nameSt: 'Иванов', sex: 'true', age: '22', idGroup: '1' }
12 { id: '2', nameSt: 'Петров', sex: 'true', age: '17', idGroup: '1' }
13 { id: '3', nameSt: 'Сидов', sex: 'true', age: '17', idGroup: '2' }
14 { id: '4', nameSt: 'Ушкина', sex: 'true', age: '20', idGroup: '3' }
15 { id: '5', nameSt: 'Берг', sex: 'true', age: '21', idGroup: '3' }
   { id: '6', nameSt: 'Немов', sex: 'true', age: '23', idGroup: '2' }
   { id: '7', nameSt: 'Мамина', sex: 'true', age: '17', idGroup: '2' }
```

Читатель #1

```
id,nameSt,sex,age,idGroup
1,Иванов,true,22,1
2,Петров,true,17,1
3,Сидов,true,17,2
4,Ушкина,true,20,3
5,Берг,true,21,3
6,Немов,true,23,2
7,Мамина,true,17,2
```

```
1
2
3 const fs = require('fs');
4 const fastcsv = require('fast-csv');
5
6 fs.createReadStream('data.csv')
7   .pipe(fastcsv.parse({ headers: true }))
8   .on('data', row => console.log(row));
9
10 fs.createReadStream('data.csv')
11   .pipe(fastcsv.parse({ headers: true }))
12   .on('data', row =>
13     console.log(row.id + '\t' + row.nameSt)
14   );
15
```


Читать #1

```
id,nameSt,sex,age,idGroup
1,Иванов,true,22,1
2,Петров,true,17,1
3,Сидов,true,17,2
4,Ушкина,true,20,3
5,Берг,true,21,3
6,Немов,true,23,2
7,Мамина,true,17,2
```

Непредсказуемый результат

```
1 { id: '1', nameSt: 'Иванов', sex: 'true', age: '22', idGroup: '1' }
2 { id: '2', nameSt: 'Петров', sex: 'true', age: '17', idGroup: '1' }
3 { id: '3', nameSt: 'Сидов', sex: 'true', age: '17', idGroup: '2' }
4 { id: '4', nameSt: 'Ушкина', sex: 'true', age: '20', idGroup: '3' }
5 { id: '5', nameSt: 'Берг', sex: 'true', age: '21', idGroup: '3' }
6 { id: '6', nameSt: 'Немов', sex: 'true', age: '23', idGroup: '2' }
7 1 Иванов
8 2 Петров
9 3 Сидов
10 4 Ушкина
11 5 Берг
12 6 Немов
13 { id: '7', nameSt: 'Мамина', sex: 'true', age: '17', idGroup: '2' }
14 7 Мамина
15
```

Читатель #2

```
id,nameSt,sex,age,idGroup
1,Иванов,true,22,1
2,Петров,true,17,1
3,Сидов,true,17,2
4,Ушкина,true,20,3
5,Берг,true,21,3
6,Немов,true,23,2
7,Мамина,true,17,2
```

```
1 const fs = require('fs');
2 const fastcsv = require('fast-csv');
3 function getArray(nameFile) {
4   let arrStudents = [];
5   fs.createReadStream(nameFile)
6     .pipe(fastcsv.parse({ headers: true }))
7     .on('data', row => arrStudents.push(row));
8   return arrStudents;
9 }
10 function printArray(arr) {
11   console.log(arr.length);
12   arr
13     .map(st => console.log(st));
14 }
15 let arr = getArray('data.csv');
16 printArray(arr);
```

0

Читать #2

```
id,nameSt,sex,age,idGroup
1,Иванов,true,22,1
2,Петров,true,17,1
3,Сидов,true,17,2
4,Ушкина,true,20,3
5,Берг,true,21,3
6,Немов,true,23,2
7,Мамина,true,17,2
```

```
1 function getArray(nameFile) {
2     ...
3     return arrStudents;
4 }
5 function printArray(arr) {
6     arr.map(st => console.log(st));
7 }
8 function sleep(ms) {
9     return new Promise(resolve => {
10         setTimeout(resolve, ms)
11     })
12 }
13 async function init() {
14     let arr = getArray('data.csv');
15     await sleep(100);
16     printArray(arr);
17 }
18
19 init();
```

```
{ id: '1', nameSt: 'Иванов', sex: 'true', age: '22', idGroup: '1' }
{ id: '2', nameSt: 'Петров', sex: 'true', age: '17', idGroup: '1' }
{ id: '3', nameSt: 'Сидов', sex: 'true', age: '17', idGroup: '2' }
{ id: '4', nameSt: 'Ушкина', sex: 'true', age: '20', idGroup: '3' }
{ id: '5', nameSt: 'Берг', sex: 'true', age: '21', idGroup: '3' }
{ id: '6', nameSt: 'Немов', sex: 'true', age: '23', idGroup: '2' }
{ id: '7', nameSt: 'Мамина', sex: 'true', age: '17', idGroup: '2' }
```

Читатель #2+

```
1 const fs = require('fs');
2 const fastcsv = require('fast-csv');
3 let text = "";
4 let stream = new fs.ReadStream('data.csv');
5
6 stream.on('readable', function() {
7   let block = stream.read();
8   if (block != null) text += block;
9 });
10 stream.on('end', function(){
11   console.log(text);
12 });
13 stream.on('error', function(){
14   console.log('Ошибка чтения данных...');
15 });
```

```
id,nameSt,sex,age,idGroup
1,Иванов,true,22,1
2,Петров,true,17,1
3,Сидов,true,17,2
4,Ушкина,true,20,3
5,Берг,true,21,3
6,Немов,true,23,2
7,Мамина,true,17,2
```

```
id,nameSt,sex,age,idGroup
1,Иванов,true,22,1
2,Петров,true,17,1
3,Сидов,true,17,2
4,Ушкина,true,20,3
5,Берг,true,21,3
6,Немов,true,23,2
7,Мамина,true,17,2
```

Читать #3

```
id,nameSt,sex,age,idGroup
1,Иванов,true,22,1
2,Петров,true,17,1
3,Сидов,true,17,2
4,Ушкина,true,20,3
5,Берг,true,21,3
6,Немов,true,23,2
7,Мамина,true,17,2
```

```
1 const fs = require('fs');
2 const csvjson = require('csvjson');
3
4 function csv_to_json(nameFile) {
5     let textCSV = fs.readFileSync(nameFile, 'utf-8');
6     return csvjson.toObject(textCSV, { delimiter: ',' });
7 }
8
9 let json = csv_to_json('data.csv');
10 json
11     .sort((a, b) => b.age - a.age)
12     .slice(0,3)
13     .map(obj => console.log(obj.age + '\t' + obj.nameSt));
14
15
```

```
23 Немов
22 Иванов
21 Берг
```

Читатель #4

```
id,nameSt,sex,age,idGroup
1,Иванов,true,22,1
2,Петров,true,17,1
3,Сидов,true,17,2
4,Ушкина,true,20,3
5,Берг,true,21,3
6,Немов,true,23,2
7,Мамина,true,17,2
```

```
1 const ut = require('./utils');
2 const fs = require('fs');
3
4
5 let lines = ut.getLines('data.csv');
6 console.log(lines);
7
8 let arrStudents = ut.getArray(lines, ',');
9
10 arrStudents
11     .sort((a, b) => b.age - a.age)
12     .slice(0,3)
13     .map(obj => console.log(obj.age + '\t' + obj.nameSt));
14
15
```

```
23 Немов
22 Иванов
21 Берг
```

ЧИТАТЬ #4 utils.js

```
1 class Student {
2   constructor(id, nameSt, sex, age, idGroup) {
3     this.id = +id;
4     this.nameSt = nameSt;
5     this.sex = !!sex;
6     this.age = +age;
7     this.idGroup = +idGroup;
8   }
9 }
10
11 function getArray(lines, del = '\t') {
12   return lines
13     .slice(1)
14     .map(line => line.split(del))
15     .map(arr => new Student(arr[0], arr[1], arr[2], arr[3], arr[4]));
16 }
17
18 function getLines(nameFile, coding = 'utf-8') {
19   return require('fs')
20     .readFileSync(nameFile, coding)
21     .split(/\r?\n/);
22 }
23
24 module.exports.Student = Student;
25 module.exports.getArray = getArray;
26 module.exports.getLines = getLines;
```

Write #1

```
1  const fs = require('fs');
2  const csvjson = require('csvjson');
3  const fastcsv = require('fast-csv');
4
5  function csv_to_json(nameFile) {
6      let textCSV = fs.readFileSync(nameFile, 'utf-8');
7      return csvjson.toObject(textCSV, { delimiter: ',' });
8  }
9
10 function writeToCsv(array, nameFile) {
11     let fw = fs.createWriteStream(nameFile);
12     fastcsv
13         .write(array, { headers: true })
14         .pipe(fw);
15 }
16
17 let arrayStudents = csv_to_json('data.csv');
18 writeToCsv(arrayStudents, 'output.csv');
19
20
```


SELECT, INSERT, UPDATE, DELETE

Вывести на экран все данные из указанной таблицы - имя таблицы вводит пользователь.

Добавить нового студента/хобби/куратора.

Изменить возраст указанного по фамилии студента.

Удалить куратора из списка по фамилии.

Найти всех студентов, указанного по фамилии куратора.

Выбрать всех студентов, указанного возраста/пола/группы.

Найти куратора по фамилии студента.

Сменить куратора у группы (на вход подаётся имя группы и имя нового куратора).

<https://qps.ru/CVhxi>