

**Универсальное автоматизированное рабочее место сбора и учета  
данных для учителей школы на базе GOOGLE таблиц**

**Е.В. Челябинов, Т.Ю. Наумова,  
муниципальное автономное общеобразовательное учреждение  
муниципального образования город Краснодар средняя  
общеобразовательная школа № 73 имени Александра Васильевича  
Молчанова**

**Аннотация.** В статье актуализируется необходимость использования автоматизированного рабочего места, как основного инструмента оперативного сбора данных и построения отчетов для учителей в повседневной профессиональной деятельности.

На основании поставленной задачи выбирается система, соответствующая основным требованиям. Она должна быть достаточно простой и быстрой в использовании, но в тоже время иметь широкие возможности, не требовать установки и сопровождения, располагать знакомым и интуитивно понятным интерфейсом, обновлять данные в режиме реального времени для всех пользователей и иметь защищенный доступ. В качестве такой системы были выбраны Google таблицы.

В статье показано создание универсального автоматизированного рабочего места сбора и учета данных для учителей школы на базе Google таблиц. Рассмотрено, как от электронных таблиц, позволяющих заносить данные в ряды и столбцы, а также производить не очень сложные вычисления перейти к полноценной автоматизированной проблемно-ориентированной системе с помощью написания скриптов на языке GoogleAppsScript.

Нами также были показаны ключевые проблемы, возникшие при разработке, и способы устранения каждой из них в ходе оптимизации для соответствия поставленным требованиям.

**Ключевые слова:** Google таблицы, GoogleApps Script, проблемно-

ориентированная система, ИТ-проект, образовательный стандарт, универсальность, журнал учёта посещаемости.

## **Universal automated workplace for data collection and accounting for school teachers based on GOOGLE tables**

**E.V. Chelyapov, T.U. Naumova,**

Municipal autonomous educational institution of the municipality of the city of Krasnodar, secondary school No. 73 named after Alexander Vasilyevich Molchanov

**Annotation.** The article actualizes the need to use an automated workplace as the main tool for operational data collection and reporting for teachers in their daily professional activities.

Based on the task, a system is selected that meets the basic requirements. It should be fairly simple and fast to use, but at the same time have ample opportunities, do not require installation and maintenance, have a familiar and intuitive interface, update data in real time for all users and have secure access. Google Tables were chosen as such a system.

The article shows the creation of a universal automated workplace for data collection and accounting for school teachers based on Google tables. It is considered how to move from spreadsheets that allow entering data into rows and columns, as well as performing not very complex calculations, to a full-fledged automated problem-oriented system by writing scripts in the Google Apps Script language.

We also showed the key problems that arose during development, and ways to eliminate each of them during optimization to meet the requirements.

**Keywords:** Google Sheets, Google Apps Script, problem-oriented system, IT project, Educational standard, Universality, Attendance log.

В современной школе количество отчетов увеличивается с каждым годом. В интернете формируется рынок специализированных и адаптированных программ и приложений, которые рекомендуются к использованию в школьном

учебном процессе. Поэтому учителя вынуждены вести отчёты одновременно в различных приложениях, которые не похожи друг на друга. В связи с этим возникают некоторые сложности, а именно: техническая, функциональная, различный интерфейс[1]. Поэтому нами было разработано универсальное автоматизированное рабочее место сбора и учета данных для учителей школы на базе Google таблиц[2, 3].

### **Постановка задачи на разработку универсального автоматизированного рабочего места на базе Google таблиц.**

Главными задачами нами были поставлены универсальность, простота использования и интуитивно понятный интерфейс. Помимо внешних удобств пользователей необходимо было учесть и возможности разработчика, которые должны были позволять формировать самые разные отчёты. После проведения анализа, нами были выбраны Google таблицы[4]. С их интерфейсом знакомы многие учителя, что позволяет сократить время на обучение.

Google таблицы обладают рядом существенных преимуществ [5, 6]:

- Понятный интерфейс;
- хранение информации в облаке;
- сохранение истории изменений;
- защита данных;
- совместное использование. Этот пункт особо выделяется, так как именно он позволяет обеспечить работу всех школьных учителей в одной системе;
- различный уровень доступа. Данный пункт вытекает из предыдущего. Разный доступ позволяет обезопасить систему от незапланированных изменений.

GoogleSheets содержат таблицы, листы, ячейки. С помощью различных функций можно формировать необходимые данные[7,8]. Если нужной функции нет во встроенных, то её всегда можно создать с помощью написания скрипта на языке GoogleAppsScript[9].

### **Реализация и практическое применение.**



Система также направлена на формирование отчётов по внесённым данным. На листах к каждому месяцу учебного года добавлены по три таблицы итогов, которые рассчитывают показатели отсутствующим за один день, месяц и статистику по каждому ученику[12]. Также на отдельном листе представлена таблица итогов за всю школу, которая формирует отчёт по общему количеству присутствующих, отсутствующих и питающихся в разрезе конкретной даты[13].

D	E	F	G	H	I	J	K
Выберите дату:	15.09.2021			Убрать пустые строки:	<input checked="" type="checkbox"/>		
<b>Классы</b>	Присутствующие:	Отсутствующие:	Питающиеся:	Пит. из многодетной:	Пит. из малообеспеченной:	Загруженная дата:	15.09.2021
1А	27	2	16	3	1		
1Б	29	4	20	1	1		
1В	32	0	23	0	2		
1Г	25	5	19	2	0		
1Д	7	0	2	3	2		
2А	28	3	17	2	3		
2Б	27	2	18	2	1		
2В	33	1	24	0	1		
2Г	23	3	17	1	0		
3А	32	2	27	2	2		
3Б	25	3	20	0	3		
3В	27	2	21	4	1		
3Г	25	2	15	5	1		
4А	33	0	24	0	0		
4Б	23	10	13	1	4		
4В	32	0	28	3	0		
4Г	25	7	15	2	5		
	Присутствующие:	Отсутствующие:	Питающиеся:	Пит. из многодетной:	Пит. из малообеспеченной:		
<b>Итого</b>	453	46	319	31	27		

Рис.2 – таблица итогов за всю школу

При создании журнала, были рассмотрены различные способы запусков скриптов. Некоторые из них запускались при открытии, некоторые при нажатии флажка. Но всё имело свои значительные недостатки. В дальнейшем была найдена возможность изменить форму меню Google таблицы, добавляя собственные пункты и подменю. Во встроенное меню Goggle таблицы мы добавили новый пункт «Школа», состоящий из пяти глав: Навигация, Формат, Оформление, Расписание, Полная таблица[14]. Они отвечают за самые основные действия, которые учитель совершает ежедневно при работе с журналом.

Глава «Навигация», представленная на рисунке 3, содержит два действия: «Текущая дата» и «Вернуться в начало».

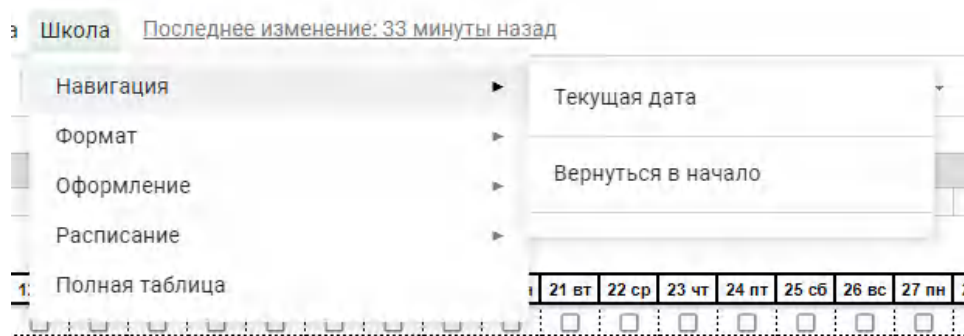


Рис.3 – глава «Навигация»

В связи с масштабностью таблицы и неудобством постоянного скроллинга, появилась необходимость в функции текущей даты. Она отправляет пользователя на столбец текущей даты. Функция «Вернуться в начало» отправляет в самое начало таблицы – на столбец первого сентября текущего учебного года.

Далее, на рисунке 4, представлена глава «Формат». Она содержит три варианта представления списка учащихся.

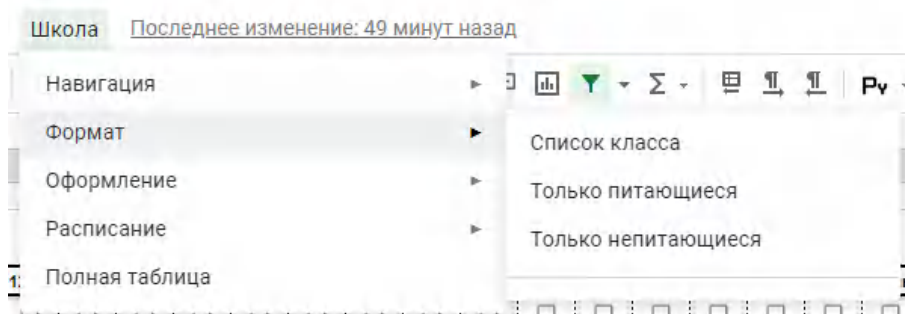


Рис.4 – глава «Формат»

При выборе первого формата будет отображаться весь список класса. При выборе второго – в списке класса останутся только питающиеся, то есть те, чей статус равен «П», «ПМД», «ПМО». При выборе третьего формата, соответственно, отобразятся только те ученики, чей статус равен «Н». Различные форматы необходимы для различных потребностей в работе с заполнением журнала питания.

Глава «Оформление», представлена на рисунке 5.

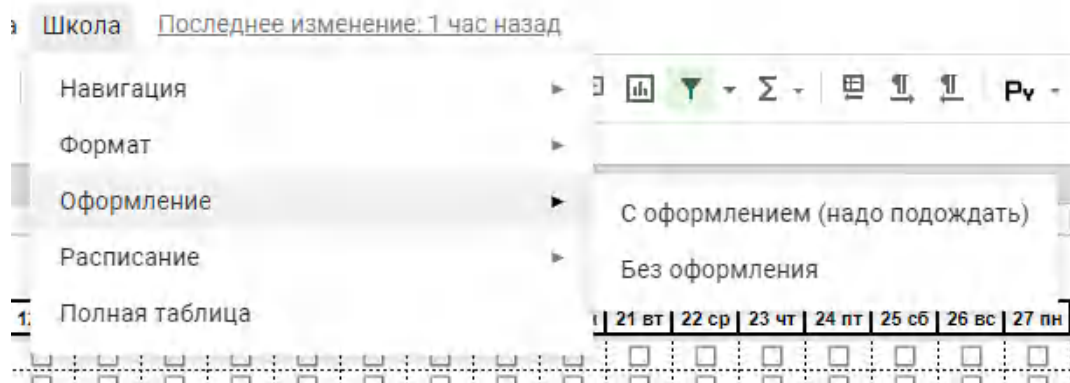


Рис.5 – глава «Оформление»

Она содержит два действия, первое из которых добавляет оформление на всех таблицах листа, а другое убирает его. В поле заполнения функция выделяет нижнюю границу каждой третьей строки, тем самым разбивая таблицу на триады. Это помогает отслеживать строку каждого ученика.

На рисунке 6 представим главу «Расписание».

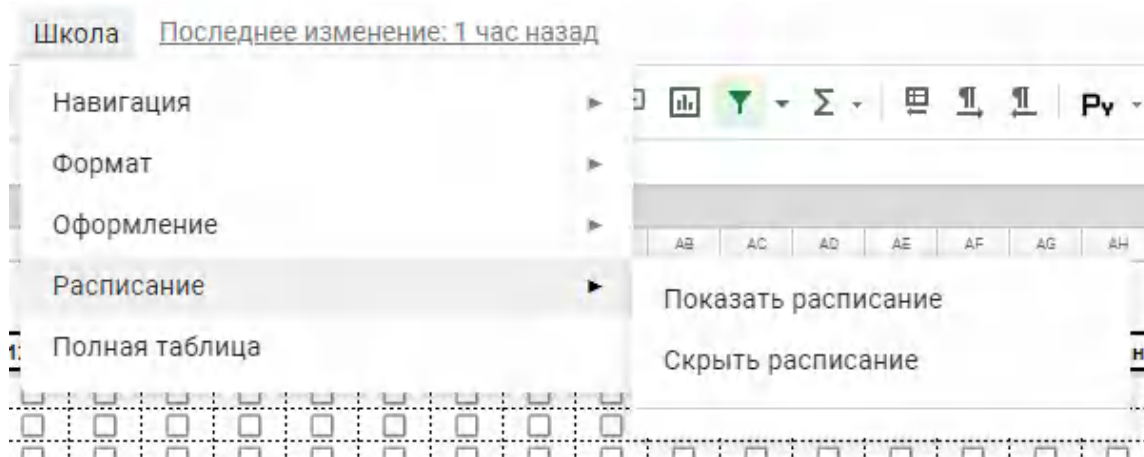


Рис.6 – глава «Расписание»

Она содержит два действия: Показать и скрыть расписание. После нажатия на первое действие над датами появляется скрытая строка с отмеченными флажками. Если убрать некоторые из флажков, то эти даты перестанут учитываться в итоговых таблицах.

Пункт «Полная таблица» полностью открывает форму заполнения списка класса, предоставляя возможность внести дополнительные данные. Как

правило, работа с этой главой происходит в начале учебного года, когда происходит формирование и внесение списков учащихся.

### **Оптимизация первых версий.**

После тестирования первых версий системы нами был выявлен и в дальнейшем устранен ряд проблем.

Проблема 1. После опробования самых различных способов запуска скриптов был заключен вывод о недостатках многих из них.

Решение: Для удобного и единообразного запуска всех скриптов был создан новый пункт меню «Школа».

Проблема 2. Время на выполнение многих скриптов начало превышать допустимое значение. Причиной стало обильно использование цветowych маркеров, условное форматирование и избыточное количество столбцов и строк.

Решение: В связи с этим было принято решение убрать большую часть маркеров, отказаться от возможностей условного форматирования, перейдя на главу нового пункта меню «Оформление». Также были убраны лишние столбцы и строки и оптимизированы скрипты[15].

Проблема 3. Два формата представления данных не удовлетворяли все потребности пользователя.

Решение: Скрипт «Только питающиеся» заменен на четыре новых: «Список класса», «Только питающиеся», «Только не питающиеся», «Полная таблица»[16].

### **Написание функций, обеспечивающих работоспособность системы.**

В конечной версии мы получили следующие скрипты, написанные на языке программирования GoogleAppsScript[17, 18].

- 1) Текущая дата (рис. 7)

```
1 function todtd() {
2   var ss = SpreadsheetApp.getActiveSpreadsheet();
3   var sheet = ss.getActiveSheet();
4   ss.toast('Выполняется скрипт', 'Подождите...', 3);
5   sheet.getRange('M61:MK46').activate();
6   dt = sheet.getRange('A5').getValue();
7   sheet.getRange('S{dt}').activate()
8 }
```



```

1 function sep() {
2   var ss = SpreadsheetApp.getActiveSpreadsheet();
3   var sheet = ss.getActiveSheet();
4   ss.toast('Выполняется скрипт', 'Подождите...', 3);
5   sheet.getRange('H6').activate()
6 }

```

Рис.7 – скрипт действия «Текущая дата»

Скрипт активирует диапазон, находящийся за последним месяцем – «Май», тем самым задавая программе маршрут поиска нужной даты, справа налево. Прежде чем найти нужную дату, таблица открывает перед пользователем текущий месяц, и останавливается на текущей дате, располагая актуальный столбец вплотную с фамилиями класса.

Текущая дата и её положение в таблице определяется с помощью встроенных функций =СЕГОДНЯ(), =ПОИСКПОЗ(), =АДРЕС()[19, 20].

## 2) С оформлением (рис. 8)

```

1 function oform() {
2   var ss = SpreadsheetApp.getActiveSpreadsheet();
3   var sheet = ss.getActiveSheet();
4   ss.toast('Выполняется скрипт', 'Подождите...', 3);
5   bezofrm();
6   ss.getRange('F7:G46').activate();
7   ss.getActiveRangeList().setBorder(null, null, null, null, true, true, '#000000', SpreadsheetApp.BorderStyle.DASHED);
8   var k = 0
9   var i = 7
10  while(i <= 46) {
11    if (sheet.isRowHiddenByFilter(i) == false) {
12      if (k%3 == 2){
13        ss.getRangeList(['D$S(i):EWS(i)', 'GS$S(i):HTS(i)', 'IC$S(i):JGS(i)', 'JP$S(i):KSS(i)', 'LB$S(i):MFS(i)', 'ES(i):AKS(i)', 'AT$S(i):BXS(i)', 'CG$S(i):DJS(i)', 'FF$S(i):GJS(i)', 'ANS(i):APS(i)', 'CAS(i):CCS(i)', 'DMS(i):DOS(i)', 'EZ$S(i):FBS(i)', 'GMS(i):GOS(i)', 'HMS(i):HVS(i)', 'JS$S(i):JLS(i)', 'KVS(i):KXS(i)', 'MIS(i):MKS(i)']).activate()
14        .setBorder(null, null, true, null, null, null, '#000000', SpreadsheetApp.BorderStyle.SOLID_MEDIUM);
15      }
16    }
17    k = k + 1
18  }
19  i = i + 1
20 }
21 sep()
22 }

```

Рис.8 – скрипт действия «С оформлением»

После выполнения функций главы формата, когда отображается только нужная часть класса, часть строк скрыта и из-за этого происходит некорректное выделение, так как выделяются скрытые строки тоже. Для решения проблемы были созданы две переменные и цикл while, которые создали два уровня таблицы: все строки и отображаемые строки.

## 3) Без оформления (рис. 9)

```

1 function bezofrm() {
2   var ss = SpreadsheetApp.getActiveSpreadsheet();
3   var sheet = ss.getActiveSheet();
4   ss.toast('Выполняется скрипт', 'Подождите...', 3);
5
6   ss.getRangeList(['DS7:EW46', 'GS7:HT46', 'IC7:JG46', 'JP7:KS46', 'LB7:MF46', 'ET:AK46', 'AT7:BX46', 'CG7:DJ46', 'FF7:GJ46', 'AN7:AP46', 'CA7:CC46', 'DM7:DO46', 'EZ7:FB46', 'GM7:GO46', 'HW7:HY46', 'JJ7:JL46', 'KV7:KX46', 'MI7:MK46']).activate()
7   .setBorder(null, null, null, null, true, true, '#000000', SpreadsheetApp.BorderStyle.DASHED);
8
9   ss.getRange('E7:G46').activate();
10  ss.getActiveRangeList().setBorder(null, null, null, null, true, true, '#000000', SpreadsheetApp.BorderStyle.SOLID);
11
12  sep()
13 }

```

Рис.9 – скрипт действия «Без оформления»

Скрипт на всём листе убирает выделение строк, заменяя всё пунктирной решёткой. После завершения скрипта автоматически отправляет пользователя на первое сентября

4) Открыть расписание (рис. 10)

```
1 function pk() {
2   var ss = SpreadsheetApp.getActiveSpreadsheet();
3   var sheet = ss.getActiveSheet();
4   ss.toast('Выполняется скрипт', 'Подождите...', 3);
5   sheet.showRows(5);
6   sheet.getRange('5:5').activate();
7 }
```

Рис.10 – скрипт действия «Открыть расписание»

В первых версиях журнала строка исключения дат из расчёта открывалась по нажатию соответствующего плюса в таблице. Но было выявлено неудобство такого способа в использовании, а также раскрытие соседних скрытых строк. Поэтому было принято решение автоматизировать данный процесс и занести его в меню “Школа”. Итогом стали два скрипта в одно действие, скрывающие и раскрывающие строку исключения дат соответственно.

5) Список класса (рис. 11)

```
1 function sheetnames() {
2   var ss = SpreadsheetApp.getActiveSpreadsheet();
3   var sheet = ss.getActiveSheet();
4   var out = new Array();
5   var sheets = SpreadsheetApp.getActiveSpreadsheet().getSheets();
6
7   ss.getRange('A1:A10').activate();
8   ss.getActiveRangeList().clear({contentsOnly: true, skipFilteredRows: true});
9
10  for (var i=0 ; i<sheets.length ; i++) {
11    a1 = sheets[i].getName();
12    var ddd = ss.getSheetByName('${a1}').getSheetId();
13    sheet.getRange(1 + i,1).setValue(a1);
14    sheet.getRange(1 + i,3).setValue(ddd);
15  }
16 }
17 }
```

Рис.11 – скрипт действия «Список класса»

Выводит в столбец ячеек список классов параллели, а также id листов, на которых расположены последние.

## б) Имя класса (рис. 12)

```
1 function newName(){
2   var ss = SpreadsheetApp.getActiveSpreadsheet();
3   var sheet = ss.getActiveSheet();
4   t = sheet.getSheetName()
5   sheet.getRange('F6').setValue(t)
6 }
```

Рис.12 – скрипт действия «Имя класса»

Помещает в указанную ячейку имя класса указанное в названии листа, исключая несоответствия и оптимизируя ручной ввод.

Следующие скрипты выполняются при выборе соответствующий действий главы «Оформление» в пункте меню «Школа»: «Список классов», «Только питающиеся», «Только не питающиеся», «Полная таблица» (рис. 13).

```
1 function spisok() {
2   var ss = SpreadsheetApp.getActiveSpreadsheet();
3   var sheet = ss.getActiveSheet();
4   ss.toast('Выполняется скрипт', 'Подождите...', 3);
5   bezdt()
6   sheet.getRange('G6').activate()
7
8   var criteria = SpreadsheetApp.newFilterCriteria()
9     .setHiddenValues([''])
10    .build();
11   ss.getActiveSheet().getFilter().setColumnFilterCriteria(7, criteria);
12
13   var spreadsheet = SpreadsheetApp.getActive();
14   spreadsheet.getRange('F2').activate();
15   spreadsheet.getCurrentCell().setValue('Список класса:');
16   spreadsheet.getRange('F6').activate();
17 }
18 }
```

Список класса

```
1 function pit() {
2   var ss = SpreadsheetApp.getActiveSpreadsheet();
3   var sheet = ss.getActiveSheet();
4   ss.toast('Выполняется скрипт', 'Подождите...', 3);
5   bezdt()
6   sheet.getRange('G6').activate()
7
8   var criteria = SpreadsheetApp.newFilterCriteria()
9     .setHiddenValues(['', 'н'])
10    .build();
11   ss.getActiveSheet().getFilter().setColumnFilterCriteria(7, criteria);
12
13   var spreadsheet = SpreadsheetApp.getActive();
14   spreadsheet.getRange('F2').activate();
15   spreadsheet.getCurrentCell().setValue('Только питающиеся:');
16   spreadsheet.getRange('F6').activate();
17 }
18 }
```

Только питающиеся

```
1 function nepit() {
2   var ss = SpreadsheetApp.getActiveSpreadsheet();
3   var sheet = ss.getActiveSheet();
4   ss.toast('Выполняется скрипт', 'Подождите...', 3);
5   bezdt()
6   sheet.getRange('G6').activate()
7
8   var criteria = SpreadsheetApp.newFilterCriteria()
9     .setHiddenValues(['', 'н', 'ПМД', 'ПМО'])
10    .build();
11   ss.getActiveSheet().getFilter().setColumnFilterCriteria(7, criteria);
12
13   var spreadsheet = SpreadsheetApp.getActive();
14   spreadsheet.getRange('F2').activate();
15   spreadsheet.getCurrentCell().setValue('Только не питающиеся:');
16   spreadsheet.getRange('F6').activate();
17 }
18 }
```

Только не питающиеся

```
1 function polnaya() {
2   var ss = SpreadsheetApp.getActiveSpreadsheet();
3   var sheet = ss.getActiveSheet();
4   ss.toast('Выполняется скрипт', 'Подождите...', 3);
5   bezdt()
6   sheet.getRange('G6').activate()
7
8   var criteria = SpreadsheetApp.newFilterCriteria()
9     .build();
10   ss.getActiveSheet().getFilter().setColumnFilterCriteria(7, criteria);
11
12   var spreadsheet = SpreadsheetApp.getActive();
13   spreadsheet.getRange('F2').activate();
14   spreadsheet.getCurrentCell().setValue('Полная таблица:');
15   spreadsheet.getRange('F6').activate();
16 }
17 }
```

Полная таблица

Рис.13 – скрипт действий «Список классов», «Только питающиеся», «Только не питающиеся», «Полная таблица»

## 6. Заключение

В заключении отметим, что мы получили универсальную проблемно-ориентированную систему, созданную с помощью Googleтаблиц. Она позволяет формировать отчёты на основе входных данных, которые в свою очередь могут принимать различную специализацию.

Также нами была доказана возможность использования Google таблиц при создании универсальных систем, ориентированных на решение ключевых проблем в самых разных сферах. Для этого можно легко преобразовывать таблицы с помощью языка программирования Google Apps Script. Главным преимуществом этой среды является бесплатное использование, что делает её общедоступной.

### **Литература**

1. Батракова И.С., Глубокова Е.Н., Писарева С.А., Тряпицына А.П. Изменения педагогической деятельности преподавателя вуза в условиях цифровизации образования // Высшее образование в России. 2021. Т. 30. № 8-9. С. 9-19. DOI: 10.31992/0869-3617-2021-30-8-9-9-19.
2. Google Таблицы. Режим доступа:  
[https://support.google.com/docs/topic/9054603?hl=ru&ref\\_topic=1382883](https://support.google.com/docs/topic/9054603?hl=ru&ref_topic=1382883)
3. Khalil Maad Ma, Adadurov S.E.B., Mahmood M.Sh.c. Mastering Google cloud: building the platform that serves your needs. // Models and Methods for Researching Information Systems in Transport. 2020, no. 11-12, p. 41–45.
4. Янковская, А. В. Поиск, создание и использование smart-технологий в практике работы педагогов ГУО «Лидский районный центр технического творчества» // Педагогическое образование в условиях трансформационных процессов: пространство самореализации личности: сб. науч. ст. IX междунар. науч.-практ. конф., Минск, 11 нояб. 2020 г. / Белорус. гос. пед. ун-т ; под науч. ред. А. В. Позняк. Минск, 2020. С. 421–426.
5. «Google Таблицы»: большой и простой гайд. Режим доступа:  
<https://texterra.ru/blog/google-tablitsy-bolshoy-gayd-dlya-novichkov.html#h2-2>
6. Google Таблицы: большой гайд по работе с инструментом. Режим доступа: <https://askusers.ru/blog/obuchenie/google-tablitsy-bolshoy-gayd-po-rabote-s-instrumentom/#anchor-3-1>
7. Roberts B. Google Sheet Functions: A step-by-step guide; 2016. 184 p.

8. Материалы по Google Таблицам. Режим доступа: <https://sreda31.ru/blog/google/sheets-lessons/>
9. Основы скрипта приложений с GoogleSheets. Режим доступа: <https://developers.google.com/codelabs/apps-script-fundamentals-1?hl=ru#0>
10. Бухарова Г. Д., Козлова А. В. Модель методики внедрения web 2.0-технологий в организацию самостоятельной работы студентов. // Образование и наука. 2012. Т. 94. № 5. С. 96–106.
11. Кувшинов Н.Н. Использование Excel и Google-таблиц для создания открытых электронных рабочих журналов преподавателя // Лучшие практики электронного обучения. Материалы III методической конференции. 2017. С. 47–53.
12. Намоконов Е., Шагабутдинов Р., Google Таблицы. Это просто. Функции и приемы – М.: Манн, Иванов и Фербер; 2018. 400 с.
13. Соболева Е. В., Суворова Т. Н., Ниматулаев М. М., Новосёлова С. Ю. Возможности интерактивных сервисов для совершенствования подготовки будущих педагогов цифровой школы // Перспективы науки и образования. 2020. № 3 (45). С. 441-458. DOI: 10.32744/pse.2020.3.32.
14. Guay M., Grubbs M., DuVall J., Bouman J. The Ultimate Guide to Google Sheets: Everything you need to build powerful spreadsheet workflows in Google Sheets; 2016. 203p.
15. Делаем документы Google Таблиц и Excel легче и быстрее. Режим доступа: [https://shagabutdinov.ru/excel\\_optimization/](https://shagabutdinov.ru/excel_optimization/)
16. Google Docs, Google Drive, Google Scripts: как писать скрипты, макросы и код – часть 0. Режим доступа: <https://sonikelf.ru/rabotaem-s-guglotablicami-pri-pomoshhi-skriptov/>
17. McPherson B Going GAS: от VBA к Google Apps Script; 2016. 724 p.
18. Справочник по Google Apps Script: основные команды для автоматизации Гугл Таблиц. Режим доступа: <https://megabyte.ga/spravochnik-po-google-apps-script-osnovnye-komandy-dlya-gugl-tablic>

19. 20+ продвинутых функций Google Таблиц. Режим доступа:

<https://blog.cybermarketing.ru/20-prodvinytyx-funkcij-google-tablic-spreadsheets/>

20. Несколько полезных функций Google Spreadsheets: IMPORTRANGE,

VLOOKUP, MATCH, INDEX. Режим доступа:

[https://shagabutdinov.ru/googledocs\\_functions/](https://shagabutdinov.ru/googledocs_functions/)

ФИО (полностью)	Челяпов Егор Витальевич
Ученая степень, звание	нет
Должность и место работы (полное название организации по Уставу ОО)	директор, учитель информатики и ИКТ высшей категории, муниципального автономного общеобразовательного учреждения муниципального образования город Краснодар средней общеобразовательной школы № 73 имени Александра Васильевича Молчанова
Муниципальное образование (район, город), в котором располагается ОО	Муниципальное автономное общеобразовательное учреждение муниципального образования город Краснодар средняя общеобразовательная школа № 73 имени Александра Васильевича Молчанова
E-mail	020887.87@mail.ru
Телефон для связи (указать – моб., дом., раб.)	+7 (918) 089-95-63 (моб.) 8(861)232-88-17 (раб.)
Адрес проживания с индексом	350001, г. Краснодар ул. Ставропольская, д.89, кв. 44.

ФИО (полностью)	Наумова Татьяна Юрьевна
Ученая степень, звание	нет
Должность и место работы (полное название организации по Уставу ОО)	учитель математики, муниципального автономного общеобразовательного

	учреждения муниципального образования город Краснодар средней общеобразовательной школы № 73 имени Александра Васильевича Молчанова
Муниципальное образование (район, город), в котором располагается ОО	Муниципальное автономное общеобразовательное учреждение муниципального образования город Краснодар средняя общеобразовательная школа № 73 имени Александра Васильевича Молчанова
Е-mail	Ivleva03101985@yandex.ru
Телефон для связи (указать – моб., дом., раб.)	+7 (988) 55-95-323 (моб.) 8(861)232-88-17 (раб.)
Адрес проживания с индексом	350055, г.Краснодар, ж/р. Новознаменский, ул.Войсковая 4 кор.3, кв. 88.