

Частное Общеобразовательное Учреждение
«Лицей Томского государственного университета»

НАУЧНО-ИССЛЕДОВАТЕЛЬСКАЯ РАБОТА
РАЗРАБОТКА ПРОГРАММНОГО КОМПЛЕКСА ДЛЯ
АВТОМАТИЗИРОВАННОГО СОСТАВЛЕНИЯ ШКОЛЬНЫХ
РАСПИСАНИЙ

Мелехин Александр Александрович

Руководитель

Научный сотрудник

лаборатории проектирования

рабочих элементов ракетно-

космической техники

НИИПММ ТГУ

_____ К.В. Костюшин

подпись

« _____ » _____ 2021 г.

Ученик 10 «Б» класса

_____ А.А. Мелехин

подпись

ОГЛАВЛЕНИЕ

Введение	3
Глава I. Анализ методов и инструментов	4
1.1. Анализ предметной области	4
1.2. Базы данных	5
1.3. Язык программирования Python	6
1.4. ORM (Object–Relational Mapping)	6
1.5. Моделирование и создание структур данных	7
1.6. DBeaver Community	7
1.7. Графический интерфейс пользователя	7
Глава II – Разработка программного комплекса	9
2.1. Моделирование структур данных	9
2.2. Создание структуры базы данных.	11
2.3. Ввод справочной информации	11
2.4. Формирование расписания	15
Заключение	16
Литература	17
Приложения	18
Приложение 1 Модель данных	18
Приложение 2 Структура базы данных	19
Приложение 3 Расписание уроков	27

ВВЕДЕНИЕ

Как показал мониторинг, на сегодняшний день большинство школ составляет расписание занятий учеников вручную.

Совместить все классы, учителей и предметы очень сложно. А если учесть, что в разных классах на одной параллели количество часов по каждому предмету могут различаться (так как программы обучения отличаются), то задача усложняется во много раз.

При количестве учеников в среднем 1000 в каждой школе, этот процесс занимает в среднем 1-2 рабочих дня (8-16 часов). По статистике в России в 2020 году числилось 40 000 школ [19]. Это составляет примерно 400 000 часов рабочего времени для составления расписания для всех школ России. Рот средних зарплатах 100-250 в час (в зависимости от региона и школы), государство тратит на составление расписания примерно 40 000 000 рублей. Напомни так же что расписание в среднем меняется 2-4 раза в год. Это 80-160 миллионов рублей в год.

Возникла необходимость автоматизации процесса составления расписания, что сильно сэкономит время, затраченное на выполнение данной задачи, и минимизирует вероятность возникновения ошибок.

Была поставлена задача разработать программное обеспечение для автоматизирования составления школьного расписания.

Для ее реализации необходимо проанализировать методы составления расписания, применяемые в Лицее ТГУ. Так же необходимо изучить технические методы решения подобного рода задач.

ГЛАВА I. АНАЛИЗ МЕТОДОВ И ИНСТРУМЕНТОВ

1.1. Анализ предметной области

На данный момент для составления расписания подготавливаются информационные списки:

- Список классов и подгрупп;
- Список предметов для каждого класса/подгруппы;
- Какие предметы делятся на подгруппы;
- Количество часов каждого предмета для каждого класса/подгруппы;
- Список аудиторий с привязкой к классу или предмету;
- Список всех преподавателей
- Список доступного времени преподавателя;
- Список какой преподаватель в каком классе/подгруппе какой предмет ведет.

На основе анализа полученной информации составляется расписание в виде таблицы (Рис.1).

10 классы
Лидей ТТУ > 10 классы

28 уч. неделя	10А – ПП - 146	10Б – ТЕХН. - 147	10В – СЗ - 150	10Г – ХБ - 150
Понедельник	08.45-10.20 146, Чернышова Ю.О.	информатика	физика	физика
	10.30-12.00 146, Чернышова Ю.О.	физика	литература	литература
	12.30-14.05 146, Чернышова Ю.О.	МХК	английский язык	английский язык
	14.15-15.50 146, Чернышова Ю.О.			
Вторник	08.45-10.20 147, Чернышова Ю.О.	геометрия		
	10.30-12.00 147, Чернышова Ю.О.	алгебра	история	химия
	12.30-14.05 147, Чернышова Ю.О.	физика	алгебра	алгебра
	14.15-15.50 147, Чернышова Ю.О.			
Среда	08.45-10.20 148, Чернышова Ю.О.	химия	английский язык	английский язык
	10.30-12.00 148, Чернышова Ю.О.	литература	экономика	химия
	12.30-14.05 148, Чернышова Ю.О.	английский язык	биология	обществознание
	14.15-15.50 148, Чернышова Ю.О.			
Четверг	08.45-10.20 149, Чернышова Ю.О.	русский язык	алгебра	алгебра
	10.30-12.00 149, Чернышова Ю.О.	алгебра	геометрия	геометрия
	12.30-14.05 149, Чернышова Ю.О.	английский язык	русский язык	русский язык
	14.15-15.50 149, Чернышова Ю.О.			
Пятница	08.45-10.20 150, Чернышова Ю.О.	биология	литература	литература
	10.30-12.00 150, Чернышова Ю.О.	обществознание	обществознание	биология
	12.30-14.05 150, Чернышова Ю.О.	литература	история	биология
	14.15-15.50 150, Чернышова Ю.О.	история	информатика	информатика
Суббота	08.45-10.20			
	10.30-12.00			
	12.30-14.05			
	14.15-15.50			

Рис. 1

1.2. Базы данных

Практически все современные информационные системы используют базы данных. База данных — совокупность данных, хранимых в соответствии со схемой данных, манипулирование которыми выполняют в соответствии с правилами средств моделирования данных. Базы данных классифицируются по применяемой модели данных: [1]

- Иерархическая
- Объектная
- Объектно-реляционная
- Реляционная
- Сетевая
- Функциональная

Управлением базой данных занимается система управления базами данных (СУБД) (англ. Database Management System, сокр. DBMS). СУБД – совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных [2].

Наибольшее практическое применение получили реляционные системы управления базами данных. Практически все разработчики современных приложений, предусматривающих связь с системами баз данных, ориентируются на реляционные СУБД. По данным аналитиков на 2010 год, реляционные СУБД используются в абсолютном большинстве крупных проектов по разработке информационных систем. По результатам исследований компании IDC 2009 года всего около 7% составляют проекты, в которых используются СУБД нереляционного типа [3].

Учитывая популярность использования реляционных СУБД и то, что мой проект не предполагает больших нагрузок, для реализации проекта на мой взгляд подойдет легкая, но в то же время достаточно производительная СУБД SQLite [4].

1.3. Язык программирования Python

Одним из самых популярных современных языков программирования является Python. Python используется в анализе данных, машинном обучении, DevOps [5] и веб-разработке, а также в других сферах, включая разработку игр. За счёт читабельности, простого синтаксиса и отсутствия необходимости в компиляции язык хорошо подходит для обучения программированию, позволяя концентрироваться на изучении алгоритмов, концептов и парадигм. Отладка же и экспериментирование в значительной степени облегчаются тем фактом, что язык является интерпретируемым.

Применяется язык многими крупными компаниями, такими как Google и Facebook. По состоянию на апрель 2021 года Python занимает третье место в рейтинге TIOBE популярности языков программирования с показателем 11,03% [6] и объявлялся «Языком года» по версии TIOBE в 2007, 2010, 2018 и 2020 году [7]. Поэтому в качестве языка программирования я выбираю Python.

1.4. ORM (Object–Relational Mapping)

Для упрощения выполнения операций CRUD (create, read, update, delete) [8] с базой данных можно использовать технологию ORM (Object–Relational Mapping). Это технология программирования, которая связывает базы данных с концепциями объектно-ориентированных языков программирования, создавая «виртуальную объектную базу данных» [9].

В языке Python наиболее популярной библиотекой ORM является Peewee, она очень проста в изучении и использовании. Все возложенные на нее функции эта библиотека выполняет быстро и с высокой стабильностью [10].

1.5. Моделирование и создание структур данных

Для моделирования структур данных используются различные специализированные программные средства. Но, непосредственно для СУБД SQLite, такого бесплатного программного обеспечения не существует, поэтому я буду использовать средство для СУБД MySQL – MySQL Workbench [11]. А разработанную с помощью неё модель данных создам в своей СУБД SQLite при помощи языка SQL.

Язык SQL – декларативный язык программирования, применяемый для создания, модификации и управления данными в реляционной базе данных, управляемой соответствующей системой управления базами данных [12].

1.6. DBeaver Community

Для оперативного просмотра и изменения объектов базы данных существует много различных утилит. Я выбрал универсальную бесплатную утилиту DBeaver Community. Она предоставляет весь спектр необходимых функций для разработчика баз данных [13]:

- Исполнение SQL запросов
- Менеджер данных
- Подсветка синтаксиса и авто-завершение кода запросов SQL
- Менеджер структуры базы данных
- Менеджер SQL скриптов
- Генератор языка описания данных (Data Definition Language)

1.7. Графический интерфейс пользователя

В Python существует много различных библиотек для создания графического интерфейса пользователя [14]:

- Tkinter

- Flexx
- CEF Python
- Dabo
- Kivy
- Pyforms
- PyGObject
- PyQt
- PySide
- PyGUI
- PyGTK
- wxPython

Для своего проекта я выбрал библиотеку Tkinter [15], которая изначально встроена в сам язык, является простой для изучения и предоставляет все основные возможности для создания полноценного приложения.

Кроме того, эта библиотека является кросс-платформенной [16], что обеспечит стабильную работу моей программы во всех наиболее популярных операционных системах [17].

ГЛАВА II – РАЗРАБОТКА ПРОГРАММНОГО КОМПЛЕКСА

2.1. Моделирование структур данных

В результате анализа исходных данных и требуемого результата, я спроектировал модель данных (Приложение 1).

options – таблица настроек

options_id	Идентификатор параметра
option_value	Значение параметра

call_table – таблица звонков

lesson_namber	Идентификатор пары
lesson_begin	Начало пары
lesson_end	Окончание пары

teachers_has_subjects – таблица отношений учителей, предметов и классов

teacher_id	Идентификатор преподавателя
subject_id	Идентификатор предмета
class_id	Идентификатор класса

timetable – таблица расписания

classroom_id	Идентификатор кабинета
class_id	Идентификатор класса
subject_id	Идентификатор предмета
group_id	Идентификатор подгруппы класса
teacher_time_id	Идентификатор расписания преподавателя

teachers_time – таблица расписания преподавателя

teacher_time_id	Идентификатор расписания преподавателя
teacher_id	Идентификатор преподавателя
day_name	Номер дня недели
lesson_number	Идентификатор пары

classes – таблица классов

class_id	Идентификатор класса
class_name	Название класса
classroom_id	Идентификатор аудитории
separate	Признак разделения класса на подгруппы

groups – таблица подгрупп классов

group_id	Идентификатор подгруппы класса
group_name	Название подгруппы класса
class_id	Идентификатор класса

teachers – таблица преподавателей

teacher_id	Идентификатор преподавателя
teacher_name	Название преподавателя

lesson_hours – таблица распределения часов

subject_id	Идентификатор предмета
class_id	Идентификатор класса
hours	Количество часов
group_id	Идентификатор подгруппы класса

classrooms – таблица кабинетов

classroom_id	Идентификатор аудитории
room_number	Номер аудитории

classroom_time – таблица занятости аудиторий

lesson_number	Идентификатор пары
day_name	Номер дня недели
classroom_id	Идентификатор аудитории

subjects_has_classrooms – таблица отношений предметов и аудиторий

subject_id	Идентификатор предмета
classroom_id	Идентификатор аудитории

subjects – таблица отношений предметов и аудиторий

subject_id	Идентификатор предмета
subject_name	Название предмета
separate	Признак разделения предмета на подгруппы
has_room	Признак привязки кабинета к аудитории

2.2. Создание структуры базы данных.

На основании спроектированной модели данных была сгенерирована структура базы данных на языке SQL с помощью DBeaver Community (Приложение 2).

2.3. Ввод справочной информации

Раздел «Предметы» (Рис. 2)

Номер предмета	Предмет	Разделён на подгруппы	Привязан к аудитории
1	английский	да	нет
2	русский	нет	нет
3	биология	нет	нет
4	химия	нет	нет
5	литература	нет	нет
6	история	нет	нет
7	обществознание	нет	нет
8	МХК	нет	нет
9	физкультура	нет	да
10	алгебра	нет	нет

Введите название предмета

добавить

Убрать выбранный предмет

Разделить выбранный предмет на подгруппы

Объединить выбранный предмет

Связать выбранный предмет с аудиторией

Отвязать выбранный предмет от аудитории

Привязанность предметов к аудиториям

Предмет	Аудитория
информатика	148
физкультура	стадион

Рис. 2

В данном разделе вводятся все предметы, указываются привязки к аудиториям и деление на подгруппы.

Раздел «Преподаватели» (Рис. 3) предназначен для хранения списка всех преподавателя лицея.

Номер преподавателя	Имя преподавателя
1	Романович О.В.
2	Сорокин В.А
3	Кучерова А.И.
4	Колпакова Л.В
5	Зырянова Е.А.
6	Гердт Ю.Р.
7	Карнаухов С.М.
8	Третьяков Е.О.
9	Матвеева П.Ф.
10	Найбороденко М.С.

Введите имя преподавателя

Рис. 3

Раздел «Аудитории» перечисляет все аудитории (Рис. 4).

Номер аудитории	Аудитория
1	142
2	143
3	146
4	147
5	148
6	150
7	152
8	стадион

Введите номер аудитории

Рис. 4

Раздел «Расписание звонков» содержит расписание звонков всех пар, дату понедельника и номер учебной недели (Рис. 5).

Предметы	Преподаватели	Аудитории	Классы и группы	Расписание звонков	Информация об преподавателях	Количество часов
			Начало	Окончание		
Первая пара:			8:45	10:20		
Вторая пара:			10:30	12:00		
Третья пара:			12:30	14:05		
Четвёртая пара:			14:15	15:50		
Изменить расписание звонков:						
Дата понедельника: 12.04			Изменить дату понедельника			
Номер недели: 25			Изменить номер недели			

Рис. 5

Раздел «Классы и группы» содержит список всех классов с разделением на подгруппы и привязкой к аудитории (Рис. 6).

Предметы	Преподаватели	Аудитории	Классы и группы	Расписание звонков	Информация об преподавателях	Количество часов																								
<table border="1"> <thead> <tr> <th>Номер класса</th> <th>Класс</th> <th>Аудитория</th> <th>Разделён на подгруппы</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>10А - ГП</td> <td>146</td> <td>нет</td> </tr> <tr> <td>2</td> <td>10Б - ФМ</td> <td>147</td> <td>нет</td> </tr> <tr> <td>3</td> <td>11А - ГП</td> <td>142</td> <td>нет</td> </tr> <tr> <td>4</td> <td>11Б</td> <td>143</td> <td>да</td> </tr> <tr> <td>5</td> <td>10В</td> <td>150</td> <td>да</td> </tr> </tbody> </table>							Номер класса	Класс	Аудитория	Разделён на подгруппы	1	10А - ГП	146	нет	2	10Б - ФМ	147	нет	3	11А - ГП	142	нет	4	11Б	143	да	5	10В	150	да
Номер класса	Класс	Аудитория	Разделён на подгруппы																											
1	10А - ГП	146	нет																											
2	10Б - ФМ	147	нет																											
3	11А - ГП	142	нет																											
4	11Б	143	да																											
5	10В	150	да																											
<div> Добавить класс Убрать выбранный класс Разделить выбранный класс на подгруппы Объединить выбранный класс </div>																														
<p>Группы</p> <table border="1"> <thead> <tr> <th>Номер Группы</th> <th>Группа</th> <th>Класс</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>11Б - ФМ</td> <td>11Б</td> </tr> <tr> <td>2</td> <td>11Б - ХБ</td> <td>11Б</td> </tr> <tr> <td>3</td> <td>10В - ХБ</td> <td>10В</td> </tr> <tr> <td>4</td> <td>10В - СЦ</td> <td>10В</td> </tr> </tbody> </table>							Номер Группы	Группа	Класс	1	11Б - ФМ	11Б	2	11Б - ХБ	11Б	3	10В - ХБ	10В	4	10В - СЦ	10В									
Номер Группы	Группа	Класс																												
1	11Б - ФМ	11Б																												
2	11Б - ХБ	11Б																												
3	10В - ХБ	10В																												
4	10В - СЦ	10В																												

Рис. 6

Раздел «Информация об преподавателях» необходим для указания какой предмет в каком классе ведет преподаватель, а так же его расписание, когда он будет находится в лицее и сможет вести уроки (Рис. 7).

Предметы Преподаватели Аудитории Классы и группы Расписание звонков Информация об преподавателях Количество часов

Преподаватель	Предмет	Ведёт предмет в классе
Романович О.В.	биология	10Б - ФМ
Романович О.В.	информатика	11Б

Выберите преподавателя:
 Романович О.В.

	ПН	ВТ	СР	ЧТ	ПТ	СБ
первая пара	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
вторая пара	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
третья пара	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
четвёртая пара	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Рис. 7

Раздел «Количество часов» хранит информацию о том сколько часов в неделю выделено на каждый предмет для каждого класса (Рис. 8).

Timetable Maker

Справочники Сгенерировать

Предметы Преподаватели Аудитории Классы и группы Расписание звонков Информация об преподавателях Количество часов

Класс/группа	Предмет	Количество пар в неделю
10В - СЦ	история	7
11Б - ФМ	обществознание	4
11Б - ХБ	обществознание	5
10В - ХБ	обществознание	3
10В - СЦ	обществознание	2
11Б - ФМ	МХК	1
11Б - ХБ	МХК	2
10В - ХБ	МХК	4
10В - СЦ	МХК	5
11Б - ФМ	физкультура	2

Введите количество пар
 2

Рисунок. 8

2.4. Формирование расписания

Раздел «Сгенерировать» предназначен для формирования расписания (Рис. 9). Расписание выводится в браузер по умолчанию. Пример готового расписания представлен в Приложении 3.

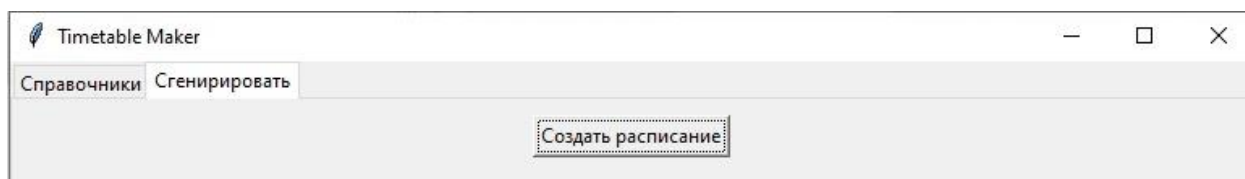


Рисунок. 9

ЗАКЛЮЧЕНИЕ

Работа выполнена в полном объеме и в соответствие с календарным планом, техническим заданием и прочими поставленными условиями. В ходе выполнения работ получены следующие результаты:

1. Проведен анализ требований к автоматизированной системе планирования рабочего расписания учеников и преподавателей Лицея ТГУ.
2. Проведено изучение баз данных и методов работы с ними.
3. Проведено изучение существующих СУБД и методик работы с ними.
4. Проведен анализ требований к СУБД при разработке автоматизированной системы составления школьных расписаний.
5. Спроектирована СУБД для планирования рабочего расписания учеников и преподавателей Лицея ТГУ.
6. Проведено изучение современных методик разработки программного обеспечения на языке Python, в том числе с использованием ORM (Object–Relational Mapping) подходов.
7. Проведено изучение современных методик разработки графического интерфейса на языке Python.
8. Разработан алгоритм автоматизированного составления расписаний расписания учеников и преподавателей.
9. Разработан программный комплекс позволяющий составлять рабочее расписание учеников и преподавателей Лицея ТГУ в автоматизированном режиме.

Разработанный программный комплекс может быть использован для составления расписания в Лицеи ТГУ, а также в других образовательных учреждениях.

ЛИТЕРАТУРА

1. https://ru.wikipedia.org/wiki/База_данных
2. https://ru.wikipedia.org/wiki/Система_управления_базами_данных
3. https://ru.wikipedia.org/wiki/Реляционная_СУБД
4. <https://www.sqlite.org/about.html>
5. <https://ru.wikipedia.org/wiki/DevOps>
6. https://ru.wikipedia.org/wiki/Индекс_ТЮВЕ
7. <https://ru.wikipedia.org/wiki/Python>
8. <https://ru.wikipedia.org/wiki/CRUD>
9. [https://ru.bmstu.wiki/ORM_\(Object-Relational_Mapping\)](https://ru.bmstu.wiki/ORM_(Object-Relational_Mapping))
10. <http://docs.peewee-orm.com/en/latest/>
11. <https://www.mysql.com/products/workbench/>
12. <https://ru.wikipedia.org/wiki/SQL>
13. <https://dbeaver.io/>
14. <https://techrocks.ru/2018/04/26/13-python-gui-frameworks/>
15. <https://docs.python.org/3/library/tkinter.html>
16. <https://ru.wikipedia.org/wiki/Кроссплатформенность>
17. https://ru.wikipedia.org/wiki/Операционная_система
18. <https://maxpark.com/community/8/content/7207619>

Приложение 1 Модель данных



Приложение 2 Структура базы данных

```
CREATE TABLE subjects (  
    subject_id INTEGER PRIMARY KEY NOT NULL,  
    subject_name TEXT NOT NULL);
```

```
CREATE TABLE teachers (  
    teacher_id INTEGER PRIMARY KEY NOT NULL,  
    teacher_name TEXT NOT NULL);
```

```
CREATE TABLE teachers_has_subjects (  
    teacher_id INT NOT NULL,  
    subject_id INT NOT NULL,  
    class_id INT NOT NULL,  
    PRIMARY KEY (teacher_id, subject_id, class_id),  
    CONSTRAINT fk_teachers_has_subjects_teachers  
        FOREIGN KEY (teacher_id)  
        REFERENCES teachers (teacher_id)  
        ON DELETE NO ACTION  
        ON UPDATE NO ACTION,  
    CONSTRAINT fk_teachers_has_subjects_subjects1  
        FOREIGN KEY (subject_id)  
        REFERENCES subjects (subject_id)  
        ON DELETE NO ACTION  
        ON UPDATE NO ACTION,  
    CONSTRAINT fk_teachers_has_subjects_classes1  
        FOREIGN KEY (class_id)  
        REFERENCES classes (class_id)  
        ON DELETE NO ACTION  
        ON UPDATE NO ACTION)
```

```
CREATE TABLE classrooms (  
    classroom_id INT NOT NULL,  
    room_number TEXT NOT NULL,  
    PRIMARY KEY (classroom_id))
```

```
CREATE TABLE lesson_hours (  
    subject_id INT NOT NULL,  
    class_id INT,  
    hours INT NOT NULL,  
    group_id INT,  
    PRIMARY KEY (subject_id, class_id),  
    CONSTRAINT fk_lesson_hours_subjects1  
        FOREIGN KEY (subject_id)  
        REFERENCES subjects (subject_id)  
        ON DELETE NO ACTION  
        ON UPDATE NO ACTION,  
    CONSTRAINT fk_lesson_hours_classes1  
        FOREIGN KEY (class_id)  
        REFERENCES classes (class_id)  
        ON DELETE NO ACTION  
        ON UPDATE NO ACTION,  
    CONSTRAINT fk_lesson_hours_groups1  
        FOREIGN KEY (group_id)  
        REFERENCES groups (group_id)  
        ON DELETE NO ACTION  
        ON UPDATE NO ACTION)
```

```
CREATE TABLE timetable (  
    classroom_id INT NOT NULL,  
    class_id INT NOT NULL,
```

```

subject_id INT NOT NULL,
group_id INT NOT NULL,
teacher_time_id INT NOT NULL,
PRIMARY KEY (classroom_id, class_id, subject_id, teacher_time_id),
CONSTRAINT fk_timetable_classrooms1
    FOREIGN KEY (classroom_id)
    REFERENCES classrooms (classroom_id)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
CONSTRAINT fk_timetable_classes1
    FOREIGN KEY (class_id)
    REFERENCES classes (class_id)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
CONSTRAINT fk_timetable_subjects1
    FOREIGN KEY (subject_id)
    REFERENCES subjects (subject_id)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
CONSTRAINT fk_timetable_groups1
    FOREIGN KEY (group_id)
    REFERENCES groups (group_id)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
CONSTRAINT fk_timetable_table21
    FOREIGN KEY (teacher_time_id)
    REFERENCES teachers_time (teacher_time_id)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)

```

```

CREATE TABLE teachers_time (
    teacher_time_id INTEGER PRIMARY KEY NOT NULL,
    teacher_id INT NOT NULL,
    day_name TEXT NOT NULL,
    lesson_number TEXT NOT NULL,
    CONSTRAINT fk_table2_teachers1
        FOREIGN KEY (teacher_id)
        REFERENCES teachers (teacher_id)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION)

```

```

CREATE TABLE timetable (
    classroom_id INT NOT NULL,
    class_id INT NOT NULL,
    subject_id INT NOT NULL,
    group_id INT NULL,
    teacher_time_id INT NOT NULL,
    PRIMARY KEY (classroom_id, class_id, subject_id, teacher_time_id),
    CONSTRAINT fk_timetable_classrooms1
        FOREIGN KEY (classroom_id)
        REFERENCES classrooms (classroom_id)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION,
    CONSTRAINT fk_timetable_classes1
        FOREIGN KEY (class_id)
        REFERENCES classes (class_id)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION,
    CONSTRAINT fk_timetable_subjects1
        FOREIGN KEY (subject_id)

```

```

REFERENCES subjects (subject_id)
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT fk_timetable_groups1
FOREIGN KEY (group_id)
REFERENCES groups (group_id)
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT fk_timetable_table21
FOREIGN KEY (teacher_time_id)
REFERENCES teachers_time (teacher_time_id)
ON DELETE NO ACTION
ON UPDATE NO ACTION);

```

```

CREATE TABLE subjects (
    subject_id INTEGER PRIMARY KEY NOT NULL,
    subject_name TEXT NOT NULL,
    separate INT NOT NULL,
    has_room INT NOT NULL,
    UNIQUE (subject_name COLLATE NOCASE) ON CONFLICT FAIL
);

```

```

CREATE TABLE teachers (
    teacher_id INTEGER PRIMARY KEY NOT NULL,
    teacher_name TEXT NOT NULL,
    UNIQUE (teacher_name COLLATE NOCASE) ON CONFLICT FAIL
);

```

```

CREATE TABLE classrooms (
    classroom_id INTEGER PRIMARY KEY NOT NULL,

```

```
room_number TEXT NOT NULL,  
UNIQUE (room_number COLLATE NOCASE) ON CONFLICT FAIL  
);
```

```
CREATE TABLE classes (  
class_id INTEGER PRIMARY KEY NOT NULL,  
class_name TEXT NOT NULL,  
classroom_id INT NOT NULL,  
separate INT NOT NULL,  
UNIQUE (class_name COLLATE NOCASE) ON CONFLICT FAIL  
UNIQUE (classroom_id COLLATE NOCASE) ON CONFLICT FAIL  
CONSTRAINT fk_classes_classrooms1  
FOREIGN KEY (classroom_id)  
REFERENCES classrooms (classroom_id)  
ON DELETE NO ACTION  
ON UPDATE NO ACTION)
```

```
CREATE TABLE subjects_has_classrooms (  
subject_id INT NOT NULL,  
classroom_id INT NOT NULL,  
UNIQUE (subject_id COLLATE NOCASE) ON CONFLICT FAIL,  
UNIQUE (classroom_id COLLATE NOCASE) ON CONFLICT FAIL,  
CONSTRAINT fk_subjects_has_classrooms_subjects1  
FOREIGN KEY (subject_id)  
REFERENCES subjects (subject_id)  
ON DELETE NO ACTION  
ON UPDATE NO ACTION,  
CONSTRAINT fk_subjects_has_classrooms_classrooms1  
FOREIGN KEY (classroom_id)  
REFERENCES classrooms (classroom_id)
```


ON DELETE NO ACTION
ON UPDATE NO ACTION)

```
CREATE TABLE groups (  
  group_id INTEGER PRIMARY KEY NOT NULL,  
  group_name TEXT NOT NULL,  
  class_id INTEGER NOT NULL,  
  UNIQUE (group_name COLLATE NOCASE) ON CONFLICT FAIL,  
  CONSTRAINT fk_groups_classes1  
  FOREIGN KEY (class_id)  
  REFERENCES classes (class_id)  
  ON DELETE NO ACTION  
  ON UPDATE NO ACTION)
```

```
CREATE TABLE call_table (  
  lesson_number INTEGER PRIMARY KEY NOT NULL,  
  lesson_begin TEXT NOT NULL,  
  lesson_end TEXT NOT NULL,  
  UNIQUE (lesson_begin COLLATE NOCASE) ON CONFLICT FAIL,  
  UNIQUE (lesson_end COLLATE NOCASE) ON CONFLICT FAIL  
);
```

```
CREATE TABLE options (  
  option_id INTEGER PRIMARY KEY NOT NULL,  
  option_value TEXT NOT NULL  
);
```

```
CREATE TABLE groups (  
  group_id INTEGER PRIMARY KEY NOT NULL,  
  group_name TEXT NOT NULL,
```

```
class_id INTEGER NOT NULL,  
UNIQUE (group_name COLLATE NOCASE) ON CONFLICT FAIL,  
CONSTRAINT fk_groups_classes1  
FOREIGN KEY (class_id)  
REFERENCES classes (class_id)  
ON DELETE CASCADE  
ON UPDATE NO ACTION);
```

```
CREATE TABLE classroom_time (  
lesson_number INT NOT NULL,  
day_name INT NOT NULL,  
classroom_id INT NOT NULL,  
CONSTRAINT fk_classroom_time_classrooms1  
FOREIGN KEY (classroom_id)  
REFERENCES classrooms (classroom_id)  
ON DELETE NO ACTION  
ON UPDATE NO ACTION)
```

Приложение 3 Расписание уроков

32 Уч. недели		10А - ГП	10Б - ТЕХН	10В	11А - ГП	11Б
24.05 Пн	8:45-10:20	.	физика 147, Пак В.В.	.	.	.
	8:45-12:00	.	физика 147, Пак В.В.	.	русский язык 142, Колпакова Л.В.	химия 143, Коротченко Н.М.
	8:45-14:05	.	физика 147, Пак В.В.	.	русский язык 142, Колпакова Л.В.	.
	8:45-15:50	.	физика 147, Пак В.В.	.	.	.
25.05 Вт	8:45-10:20	.	.	.	математика (проф) 142, Гердт Ю.Р.	информатика ф-в 143, Романович О.В.
	8:45-12:00	.	.	.	математика (проф) 142, Гердт Ю.Р.	информатика ф-п 143, Романович О.В.
	8:45-14:05	.	.	.	история 142, Коршунова А.А.	математика (проф) 143, Гердт Ю.Р.
	8:45-15:50	.	.	.	математика (проф) 142, Гердт Ю.Р.	информатика ф-п 143, Романович О.В.
26.05 Ср	8:45-10:20
	8:45-12:00
	8:45-14:05	.	английский язык 147, Полонская М.С.	.	.	.
	8:45-15:50	информатика 146, Алигасанова К.Л.	английский язык 147, Полонская М.С.	.	.	.
27.05 Чт	8:45-10:20	.	физика 147, Пак В.В.	русский язык 150, Навро Р.И.	.	.
	8:45-12:00	.	физика 147, Пак В.В.	русский язык 150, Навро Р.И.	русский язык 142, Колпакова Л.В.	химия 143, Коротченко Н.М.
	8:45-14:05	информатика 146, Алигасанова К.Л.	физика 147, Пак В.В.	алгебра 150, Нармалнов Р.К.	русский язык 142, Колпакова Л.В.	.
	8:45-15:50	.	физика 147, Пак В.В.	.	.	.
28.05 Пт	8:45-10:20	геометрия 146, Некрач Е.Н.	литература 147, Третьяков Е.О.	.	математика (проф) 142, Гердт Ю.Р.	.
	8:45-12:00	геометрия 146, Некрач Е.Н.	биология 147, Сорокин В.А.	.	литература 142, Третьяков Е.О.	математика (проф) 143, Гердт Ю.Р.
	8:45-14:05	геометрия 146, Некрач Е.Н.	биология 147, Сорокин В.А.	.	литература 142, Третьяков Е.О.	математика (проф) 143, Гердт Ю.Р.
	8:45-15:50	геометрия 146, Некрач Е.Н.	биология 147, Сорокин В.А.	.	литература 142, Третьяков Е.О.	математика (проф) 143, Гердт Ю.Р.
29.05 Сб	8:45-10:20	алгебра 146, Ромашева Т.Н.	физкультура 147, Крушицкая О.Н.	география 150, Жилина Т.Н.	.	.
	8:45-12:00	алгебра 146, Ромашева Т.Н.	физкультура 147, Крушицкая О.Н.	география 150, Жилина Т.Н.	.	.
	8:45-14:05	физкультура 146, Крушицкая О.Н.
	8:45-15:50	физкультура 146, Крушицкая О.Н.