

Сидоренко В.А., старший преподаватель
ФГБОУ ВО «КалмГУ»,
Гузенко А.С., обучающийся 3 курса
ФМФИТ ФГБОУ ВО «КалмГУ»,
г. Элиста, Российская Федерация

Lazarus и SQLite – особенности работы с типами DATE и TIME

Аннотация

В статье показаны некоторые проблемы использования нестандартных для SQLite типов полей DATE и TIME в интегрированной среде разработки Lazarus и показаны пути их решения. Рассмотрены практические примеры, а также даны рекомендации по работе с указанными типами полей. Данная статья будет полезна для разработчиков баз данных, использующих среду программирования Lazarus.

Ключевые слова: типы DATE и TIME в SQLite и Lazarus, SQLite в Lazarus – проблемы с DATE и TIME, не редактируется тип DATE SQLite в Lazarus.

В Linux из инструментов разработки приложений имеется бесплатный свободно распространяемый Lazarus. Эта среда программирования была задумана как аналог Delphi. Она так же использует язык программирования Object Pascal, который здесь преобразовался в Free Pascal [1]. Содержание и интерфейс Lazarus так же очень близки к Delphi. Основной набор компонентов совпадает – и по названиям и по работе. Причём такой же Lazarus практически без каких-либо изменений работает и в Windows. И это в некоторых ситуациях удобно. Так что в учебном процессе это оптимальный вариант для разработки приложений.

Так же как и в Delphi, в Lazarus есть возможность работать с различными типами баз данных. Однако технологии разработки баз данных здесь отличны. И при изучении баз данных в Lazarus приходится довольно часто сталкиваться с различными сложностями. Неудобство ещё в том, что информации, литературы по работе с базами данных в Lazarus очень мало. Видимо из-за малой популярности этой среды среди программистов. Поэтому за основу в изучении берём Delphi и пробуем применять в Lazarus похожие приёмы программирования баз данных. Экспериментируя, добиваемся решения возникающих иногда проблем.

В этой статье разберём нюансы работы с типами данных DATE и TIME в SQLite. Некоторые особенности поведения типа DATE уже обсуждались в статье [4], здесь же попробуем подробнее изучить практическое применение этого типа и близкого ему TIME. Определим надёжные способы работы с ними в Lazarus, так как указанные типы для SQLite нестандартные, то неминуемо возникают странности их поведения. Напомним, что объявление этих типов в

таблицах SQLite допускает, но сам их воспринимает как текстовые, а Lazarus пытается с ними работать как с обычным типом DateTime в Pascal.

Возьмём некую таблицу, покажем её в Lazarus с помощью компонента DBGrid.

ID	Name	DR	TT
1	Попов А.С.	23.12.1980	12:00:00
2	Моцарт В.А.	30.04.1999	13:12:05
4	Волков	30.12.1899	
5	Волков А.	30.12.1899	00:00:00
6	Горбунков С.	12.02.1951	00:10:00
7	Моргунов Е.	11.10.1935	

Рис.1. Исходная таблица показана в Lazarus

Теперь та же самая таблица – через SQLiteStudio.

ID	Name	DR	TT
1	1 Попов А.С.	2444596.5	2415019
2	2 Моцарт В.А.	36280	2415019.0500578703
3	4 Волков	27.01.1756	NULL
4	5 Волков А.		
5	6 Горбунков С.	2433689.5	2415018.5069444445
6	7 Моргунов Е.	2428086.5	NULL

Рис.2. Исходная таблица показана в SQLiteStudio

С таблицей уже работали разными методами и в разных средах, поэтому дата и время представлены здесь в таком разнообразии. Здесь есть и пустые поля, и с пометкой NULL, и как бы с нормальной датой (текстовой), и странные числа, но нормальные в Lazarus. Напомним, что дата 30.12.1899 в Pascal это число 0. И эта дата показана как при пустом поле в SQLiteStudio, так и при текстовом виде даты – это строки 4 и 5. Там же, где мы видим NULL в SQLiteStudio, в Lazarus мы видим пустое поле (строки 4 и 7, поле TIME).

Попробуем редактировать эту таблицу в Lazarus.

ID	Name	DR	TT
1	Попов	15.12.1966	00:25:00
2	Моцарт	30.04.1756	10:00:00
4	Волков Сергей	15.01.1990	
5	Волков Антон	18.12.1970	11:00:00
6	Горбунков С.С.	12.02.1949	00:10:00
7	Моргунов Евгени	11.10.1930	11:15:00

Рис.3. Попытка изменений таблицы в Lazarus

Перезагрузим эту таблицу и посмотрим результат:

ID	Name	DR	TT
1	Попов	15.12.1966	00:25:00
2	Моцарт В.А.	30.04.1999	13:12:05
4	Волков	30.12.1899	
5	Волков А.	30.12.1899	00:00:00
6	Горбунков С.С.	12.02.1949	00:10:00
7	Моргунов Евгений	11.10.1930	11:15:00

Рис.4. Проверка результата корректировки данных

Мы видим, что где-то корректировки удались (где ID=1,6,7), а где-то ничего не изменилось (2,4,5). То есть, просто корректировать через компонент DBGrid без дополнительных приёмов – дело ненадёжное. Поэтому будем разбираться, как решить эту проблему.

В первую очередь можно надеяться на правильные SQL-запросы INSERT и UPDATE. Использовать, конечно, будем встроенный язык SQL в среде Lazarus.

К примеру, возьмём запрос

```
UPDATE Proba2
SET NAME='Веселов', DR='02.02.1992', TT='11:11:11'
WHERE ID=1');
UPDATE Proba2
SET NAME='Зайцев', DR='01.01.1991', TT=NULL
WHERE ID=4;
```

После выполнения этого запроса получаем:

ID	Name	DR	TT
1	Веселов	30.12.1899	11:11:11
2	Моцарт В.А.	30.04.1999	13:12:05
4	Зайцев	30.12.1899	

Рис.5. Результат корректировки в ID=1 и ID=4 – Lazarus

ID	Name	DR	TT
1	1 Веселов	02.02.1992	11:11:11
2	2 Моцарт В.А.	36280	2415019.0500578703
3	4 Зайцев	01.01.1991	NULL

Рис.6. Результат корректировки в ID=1 и ID=4 – SQLiteStudio

С точки зрения SQLiteStudio здесь всё правильно – все тексты изменились согласно запросу. Lazarus воспринимает даты как тексты и выдаёт 0-число, т.е. 30.12.1899. Интересно, что и время теперь стало текстом, и Lazarus считает его текстом. Следовательно, поступать так с датой и временем, как привыкли в Delphi с правильным типом DATE в SQL-запросе, не получится – брать в кавычки нельзя. Другие какие-то варианты записи даты без кавычек (02/02/1992 или 1992/02/02 или 1992-02-02 и т.д.) положительного результата тоже не дали. Поэтому ясно, что надо идти классическим путём – применять в запросе параметры. Например:

```

D:= EncodeDate(1992, 3, 31); // var D:double
T:= EncodeTime(14, 01, 27, 0); // var T:double
SQLQueryX.SQL.Clear;
SQLQueryX.SQL.Add('UPDATE Proba2 SET DR=:D, TT=:T');
SQLQueryX.SQL.Add('WHERE ID=1');
SQLQueryX.Params.Clear;
SQLQueryX.Params.Add;
SQLQueryX.Params[0].Name:= 'D';
SQLQueryX.Params[0].Value:= D;
SQLQueryX.Params.Add;
SQLQueryX.Params[1].Name:= 'T';
SQLQueryX.Params[1].Value:= T;
SQLQueryX.ExecSQL;
SQLTransaction1.Commit;

```

ID	Name	DR	TT
1	Веселов	31.03.1992	14:01:27

Рис.7. Результат выполнения запроса с параметрами – Lazarus

ID	Name	DR	TT
1	Веселов	33694	0.584340277777778

Рис.8. Результат выполнения запроса с параметрами – SQLiteStudio

Здесь уже видно, что сработано всё правильно: дата – это целое число, а время – часть суток, т.е. дробное число меньше 1.

Отдельными SQL-запросами редактировать записи не всегда удобно. Посмотрим, как можно всё же использовать DBGrid для редактирования таблицы. У объекта SQLQuery есть свойство UpdateSQL. В Инспекторе Объектов запишем в UpdateSQL следующий запрос с параметрами Name, DR, TT и ID:

```

UPDATE Proba2
SET NAME=:Name, DR=:DR, TT=:TT
WHERE ID=:ID

```

Свойство Params заполнять не надо, с параметрами будем работать программно. В событии OnBeforePost объекта SQLQuery1 запишем следующий обработчик (процедуру):

```

procedure TForm1.SQLQuery1BeforePost(DataSet: TDataSet);
begin
  with SQLQuery1 do begin
    Params.Clear;
    Params.Add;
    Params.Add;
    Params.Add;
    Params.Add;
    Params[0].Name:='ID';
    Params[0].DataType:= ftInteger;
    Params[1].Name:='Name';

```

```

Params[1].DataType:= ftString;
Params[2].Name:='DR';
Params[2].DataType:= ftDate;
Params[3].Name:='TT';
Params[3].DataType:= ftTime;
ParamByName('ID').Value:= FieldByName('ID').Value;
ParamByName('DR').Value:= FieldByName('DR').Value;
ParamByName('TT').Value:= FieldByName('TT').Value;
ParamByName('Name').Value:= FieldByName('Name').Value;
end;
end;

```

Теперь можно работать с таблицей через DBGrid. Добавка новой записи тоже работает правильно.

ID	Name	DR	TT
1	Веселов Олег	03.03.1980	15:00
2	Моцарт Вольфган	27.01.1756	06:00
4	Зайцев Сергей	30.12.1990	12:00
5	Волков Антон	23.01.1950	09:00
6	Горбунков Семён	12.02.1949	00:10
7	Моргунов Евгений	11.10.1930	11:15
8	Вицын Георгий	12.05.1933	15:00

Рис.9. Конечный результат работы через DBGrid с использованием запроса с параметрами UpdateSQL.

Кстати, здесь время показано уже в более удобном формате. Это достигается следующим образом: в настройке столбцов DBGrid есть свойство DisplayFormat, в котором указываем **hh : mm**.

Список использованной литературы

1. Википедия: Lazarus <https://ru.wikipedia.org/wiki/Lazarus>
2. Алексеев Е. Р., Чеснокова О. В., Кучер Т. В. Free Pascal и Lazarus: Учебник по программированию. – М.: Альт Линукс, ДМК Пресс, 2010. 440 с.
3. Архангельский А.Я. Программирование в Delphi 7. – М.: ООО Бином-Пресс, 2003 г.–1152 с.
4. Сидоренко В.А. Проблемы несогласованности типа DATE в связке СУБД SQLite и среды разработки Lazarus. Сборник статей научно-практического Круглого стола «Адаптация человека к вызовам цифровой эпохи», с.111–114. – Элиста, 2023

Сидоренко В.А.,
Гузенко А.С.

2024