ChatGPT

ПОДРОБНАЯ ШПАРГАЛКА C# + SQLite + WinForms (с объяснениями)

---

1. КАК ПОДКЛЮЧИТЬСЯ К БАЗЕ ДАННЫХ SQLite

SQLite — это файл-база. Всё хранится в файле `materials.db`. Чтобы подключиться:

```
using System.Data.SQLite;

string config = "Data Source=materials.db;Version=3;";
using (var connection = new SQLiteConnection(config))
{
    connection.Open();
    // здесь ты можешь выполнять запросы
}
```

`using` нужен, чтобы соединение автоматически закрылось после блока.

---

2. ЗАПРОСЫ С ПОМОЩЬЮ SQLiteCommand

SELECT — чтение данных:

```
string query = "SELECT name FROM table WHERE id = @id";
using (var command = new SQLiteCommand(query, connection))
{
    command.Parameters.AddWithValue("@id", 1); // безопасная подстановка
    using (var reader = command.ExecuteReader())
    {
        while (reader.Read())
        {
            string name = reader.GetString(0); // читаем результат
        }
    }
}
```

INSERT / UPDATE / DELETE — изменение данных:

```
string query = "INSERT INTO table (name) VALUES (@name)";
using (var command = new SQLiteCommand(query, connection))
```

```
{
    command.Parameters.AddWithValue("@name", "Глина");
    command.ExecuteNonQuery(); // выполняем (не возвращает результат)
}
```

Получить одно значение (например, id):

```
var result = command.ExecuteScalar();
int id = Convert.ToInt32(result);
```

JOIN — соединение таблиц:

```
SELECT m.id, t.material_type
FROM materials m
JOIN material_types t ON m.material_type_id = t.id
```

JOIN связывает таблицы по общему полю (например, `id`). Используется, чтобы достать связанные данные (типы, имена и т.д.).

---

3. ПРЕОБРАЗОВАНИЯ ТИПОВ + ПРОВЕРКИ

int.TryParse — проверить, можно ли строку превратить в число:

```
if (int.TryParse(textBox1.Text, out int number))
{
    // number теперь это int
    if (number < 0) { ... }
}
else
{
    // не получилось
}
```

- `out int number` — создаёт переменную внутри условия.

string.IsNullOrWhiteSpace — пустое ли поле:

```
if (string.IsNullOrWhiteSpace(textBox1.Text)) { ... }
```

Convert.ToInt32 / ToDouble

```
int x = Convert.ToInt32("5");
double d = Convert.ToDouble("5.23");
```

---

4. ЦИКЛЫ

```
for (int i = 0; i < 10; i++) { }
foreach (var item in list) { }
while (условие) { }
```

---

5. WinForms: элементы управления (Controls)

- `Controls.Add(...)` — добавить элемент на форму или панель
- `Controls.Clear()` — очистить всё (например, все карточки)
- `AutoSize = true` — чтобы размер подстраивался под текст

```
Label l = new Label();
l.Text = "Пример";
flowLayoutPanel1.Controls.Add(l);
```

Также важно: `comboBox1.SelectedValue` — получить значение, выбранное в выпадающем списке.

---

6. ФОРМЫ: переход, ожидание, закрытие

Открыть другую форму и ждать:

```
var f = new AddEditForm();
if (f.ShowDialog() == DialogResult.OK)
{
    // пользователь сохранил
    LoadData();
}
```

Внутри AddEditForm:

```
this.DialogResult = DialogResult.OK;
this.Close();
```

---

7. SQLiteDataAdapter и DataTable — для ComboBox / таблиц

```
SQLiteDataAdapter adapter = new SQLiteDataAdapter("SELECT id, name FROM table",
connection);
DataTable table = new DataTable();
adapter.Fill(table);

comboBox1.DataSource = table;
comboBox1.DisplayMember = "name"; // что видно
comboBox1.ValueMember = "id";     // что получаешь

int selectedId = Convert.ToInt32(comboBox1.SelectedValue);
```

Это удобно для выпадающих списков (типы, материалы и т.д.).

---

8. ДОКУМЕНТАЦИЯ В VISUAL STUDIO (без интернета)

- Наведи мышку на метод — появится краткое описание
- Нажми `Ctrl + клик` — переход к определению
- Если включено: `F1` откроет оффлайн-доку

---

9. GIT — минимальный набор

```
git init             # инициализация

# добавить файлы и закоммитить
git add .
git commit -m "Первый коммит"
git push origin main

# отменить коммит (оставить файлы)
git reset --soft HEAD~1

# удалить файл из коммита
```

```
git rm --cached путь/к/файлу

# удалить файл полностью
git rm путь/к/файлу
```

Если репозиторий уже есть:

```
git clone https://адрес_репозитория.git
```

---

10. .EXE файл в C#

- Собери проект (`Ctrl+Shift+B`)
- Ищи `.exe` в папке `/bin/Debug/`

---

11. ДЛЯ МОДУЛЯ 4: Расчёт количества продукции

```
public int CalculateProductAmount(int productTypeId, int materialTypeId, int
materialUsed, double param1, double param2)
{
    if (productTypeId <= 0 || materialTypeId <= 0 || param1 <= 0 || param2 <= 0)
        return -1;

    // Заглушки — в реальности данные подгружаются из БД
    double coefficient = 1.5; // коэффициент типа продукции
    double lossPercent = 10.0; // процент потерь сырья

    double materialPerUnit = param1 * param2 * coefficient;
    double totalNeeded = materialPerUnit * (1 + lossPercent / 100);

    int amount = (int)(materialUsed / totalNeeded);
    return amount;
}
```

Если данные не найдены или параметры некорректны — возвращает `-1`.

У тебя получится. Ты уже умеешь больше, чем ты думаешь