

□ Dadata: Кадастровый номер → Адрес + Координаты

□ Схема работы

Входной столбец (A): Кадастровые номера
Выходные столбцы (B, C, D): Адрес, широта, долгота

Входной столбец A: Выходные столбцы:
77:02:0004008:4143 → B: г Москва, ул Сухонская, д 11
78:49:0001037:0001 → C: 55.8782557 (широта)
D: 37.65372 (долгота)

□ ГОТОВЫЙ КОД ДЛЯ VBA (КОПИРУЙ И ВСТАВЛЯЙ)

Option Explicit

```
' =====
' DADATA API - ОБРАТНОЕ ГЕОКОДИРОВАНИЕ
' Кадастровый номер → Адрес + Координаты
' =====

Const API_KEY As String = "ВСТАВЬТЕ_СЮДА_API_КЛЮЧ"
Const SECRET_KEY As String = "ВСТАВЬТЕ_СЮДА_SECRET_КЛЮЧ"
Const API_URL_CADASTRE As String = "https://cleanerdadata.ru/api/v1/clean/cadastre

Type DadarResult
    address As String
    latitude As String
    longitude As String
    errorMsg As String
End Type

' =====
' ОСНОВНАЯ ФУНКЦИЯ: получение адреса по кадастровому номеру
' =====

Function GetAddressByCadastralNumber(cadNum As String) As DadarResult
    Dim xmlhttp As Object
    Dim jsonString As String
    Dim result As DadarResult
```

```
On Error GoTo ErrorHandler
```

```
' Проверяем, что ключи установлены
```

```

If API_KEY = "ВСТАВЬТЕ СЮДА API КЛЮЧ" Then
    result.errorMsg = "ОШИБКА: Установите API и SECRET ключи в коде!"
    GetAddressByCadastralNumber = result
    Exit Function
End If

' Проверяем формат кадастрового номера
If Len(Trim(cadNum)) < 5 Then
    result.errorMsg = "Кадастровый номер слишком короткий"
    GetAddressByCadastralNumber = result
    Exit Function
End If

Set xmlhttp = CreateObject("MSXML2.XMLHTTP")

' Формируем JSON запрос (массив с одним кадастровым номером)
Dim requestData As String
requestData = "[""" & EscapeJSONString(cadNum) & """]"

' Отправляем POST запрос
With xmlhttp
    .Open "POST", API_URL_CADASTRE, False
    .setRequestHeader "Content-Type", "application/json;charset=utf-8"
    .setRequestHeader "Accept", "application/json"
    .setRequestHeader "Authorization", "Token " & API_KEY
    .setRequestHeader "X-Secret", SECRET_KEY
    .Send requestData

    ' Проверяем код ответа HTTP
    If .Status = 200 Then
        jsonString = .responseText
        result = ParseDadataResponseCadastre(jsonString)
    Else
        result.errorMsg = "HTTP Error: " & .Status
    End If
End With

```

```
GetAddressByCadastralNumber = result
```

```
Exit Function
```

```
ErrorHandler:
```

```
result.errorMsg = "Request Error: " & Err.Description
```

```
GetAddressByCadastralNumber = result
```

```
End Function
```

```
' =====
```

```
' ПАРСЕР JSON ОТВЕТА (для кадастровых номеров)
```

```
' =====
```

```
Function ParseDadataResponseCadastre(jsonStr As String) As DadataResult
```

```
Dim result As DadataResult
```

```
Dim firstObjStart As Long
```

```
Dim firstObjEnd As Long
```

```
Dim objStr As String
```

```
' Инициализируем результат
```

```
result.errorMsg = ""
```

```
result.address = "N/A"
```

```
result.latitude = "N/A"
```

```
result.longitude = "N/A"
```

```
' Проверяем пустоту или ошибку в ответе
```

```
If Len(jsonStr) < 5 Then
```

```
    result.errorMsg = "Empty response"
```

```
    ParseDadataResponseCadastre = result
```

```
    Exit Function
```

```
End If
```

```
' Извлекаем первый объект из JSON массива
```

```
firstObjStart = InStr(jsonStr, "{")
```

```
firstObjEnd = InStr(jsonStr, "}")
```

```
If firstObjStart > 0 And firstObjEnd > 0 Then
```

```
    objStr = Mid(jsonStr, firstObjStart, firstObjEnd - firstObjStart + 1)
```

```
' Парсим нужные поля для кадастра
```

```
' Обычно Dadata возвращает address, geo_lat, geo_lon
```

```
result.address = ExtractFieldValue(objStr, "address")
```

```

' Если address не найден, ищем в других полях
If result.address = "N/A" Then
    result.address = ExtractFieldValue(objStr, "result")
End If

result.latitude = ExtractFieldValue(objStr, "geo_lat")
result.longitude = ExtractFieldValue(objStr, "geo_lon")

' Если координаты не найдены, пробуем из center (в некоторых ответах
If result.latitude = "N/A" Then
    result.latitude = ExtractFieldValue(objStr, "lat")
End If
If result.longitude = "N/A" Then
    result.longitude = ExtractFieldValue(objStr, "lon")
End If

Else
    result.errorMsg = "Invalid JSON structure"
End If

```

ParseDadataResponseCadastre = result

End Function

```

' =====
' ИЗВЛЕЧЕНИЕ ЗНАЧЕНИЯ ПОЛЯ ИЗ JSON
' =====

Function ExtractFieldValue(jsonObj As String, fieldName As String) As String
Dim pattern As String
Dim regex As Object
Dim matches As Object
Dim value As String

```

```

Set regex = CreateObject("VBScript.RegExp")
regex.IgnoreCase = True
regex.Global = False

```

```

' Паттерн для поиска: "fieldName":"value" или "fieldName":null
pattern = """" & fieldName & """'\s*:|s*""([^\"]\]*(\\"[^"\"]\")*)"""" | """" & fieldName & ""
regex.pattern = pattern

```

```
Set matches = regex.Execute(jsonObj)

If matches.Count > 0 Then
    If matches(0).subMatches.Count > 0 And matches(0).subMatches(0) <> "" Then
        value = matches(0).subMatches(0)
        ' Убираем экранирование из JSON
        value = Replace(value, "\", "/")
        value = Replace(value, """", """")
        ExtractFieldValue = value
    Else
        ExtractFieldValue = ""
    End If
Else
    ExtractFieldValue = "N/A"
End If
```

```
End Function
```

```
' =====
' ЭКРАНИРОВАНИЕ СПЕЦСИМВОЛОВ ДЛЯ JSON
' =====
Function EscapeJSONString(inputStr As String) As String
Dim result As String
result = inputStr
```

```
' Порядок важен!
result = Replace(result, "\", "\\")
result = Replace(result, """", "\""")
result = Replace(result, Chr(10), "\n")
result = Replace(result, Chr(13), "\r")
result = Replace(result, Chr(9), "\t")
```

```
EscapeJSONString = result
```

```
End Function
```

```
' =====
' ГЛАВНАЯ ПРОЦЕДУРА - ОБРАБОТКА КАДАСТРОВЫХ НОМЕРОВ
' =====
Sub ProcessCadastralNumbers()
Dim ws As Worksheet
Dim lastRow As Long
```

```
Dim i As Long
Dim cadNum As String
Dim dadataResult As DadataResult
Dim startTime As Double
Dim successCount As Long
Dim errorCount As Long

On Error GoTo GlobalErrorHandler

' Используем активный лист
Set ws = ActiveSheet

' Находим последнюю заполненную строку в столбце А (Кадастровые ном
lastRow = ws.Cells(ws.Rows.Count, 1).End(xlUp).Row

If lastRow < 2 Then
    MsgBox "Нет данных для обработки! Добавьте кадастровые номера в ст
    Exit Sub
End If

' Добавляем заголовки если их нет
If ws.Cells(1, 1).Value = "" Then
    ws.Cells(1, 1).Value = "Кадастровый номер"
    ws.Cells(1, 2).Value = "Адрес"
    ws.Cells(1, 3).Value = "Широта"
    ws.Cells(1, 4).Value = "Долгота"
    ws.Cells(1, 5).Value = "Статус"
End If

startTime = Timer
successCount = 0
errorCount = 0

' Отключаем обновление экрана для скорости
Application.ScreenUpdating = False
Application.Calculation = xlCalculationManual

' Обработка каждого кадастрового номера
For i = 2 To lastRow
```

```

cadNum = ws.Cells(i, 1).Value

' Пропускаем пустые строки
If cadNum <> "" Then
    ' Получаем адрес и координаты по кадастровому номеру
    dadataResult = GetAddressByCadastralNumber(cadNum)

    ' Записываем результаты в строку
    With ws
        .Cells(i, 2).Value = dadataResult.address
        .Cells(i, 3).Value = dadataResult.latitude
        .Cells(i, 4).Value = dadataResult.longitude

        ' Статус и форматирование
        If dadataResult.errorMsg <> "" Then
            .Cells(i, 5).Value = "✗ " & dadataResult.errorMsg
            .Cells(i, 5).Font.Color = RGB(255, 0, 0)
            .Cells(i, 5).Font.Bold = True
            errorCount = errorCount + 1
        Else
            .Cells(i, 5).Value = "✓ OK"
            .Cells(i, 5).Font.Color = RGB(0, 176, 0)
            .Cells(i, 5).Font.Bold = True
            successCount = successCount + 1
        End If
    End With

    ' Обновляем статус строки состояния
    Application.StatusBar = "Обработано: " & i & " / " & lastRow & " | Успешно: " & successCount & " | Ошибок: " & errorCount

    ' Даём пользователю возможность прерваться (Esc)
    If i Mod 10 = 0 Then
        DoEvents
    End If
End If

Next i

' Включаем обновление экрана обратно

```

```

Application.ScreenUpdating = True
Application.Calculation = xlCalculationAutomatic

' Форматируем таблицу
FormatResultTable ws, lastRow

' ВЫВОДИМ ИТОГОВОЕ СООБЩЕНИЕ
MsgBox "✓ ОБРАБОТКА ЗАВЕРШЕНА" & vbCrLf & vbCrLf & _
    "Успешно: " & successCount & vbCrLf & _
    "Ошибок: " & errorCount & vbCrLf & _
    "Время: " & Format(Timer - startTime, "0.0") & " секунд", _
    vbInformation, "Готово"

Application.StatusBar = ""
Exit Sub

```

GlobalErrorHandler:

```

Application.ScreenUpdating = True
Application.Calculation = xlCalculationAutomatic
MsgBox "Критическая ошибка: " & Err.Description, vbCritical, "Ошибка"
Application.StatusBar = ""
End Sub

' =====
' ФОРМАТИРОВАНИЕ ТАБЛИЦЫ РЕЗУЛЬТАТОВ
' =====

Sub FormatResultTable(ws As Worksheet, lastRow As Long)
Dim col As Long

```

```

' Автоширина для всех столбцов
For col = 1 To 5
    ws.Columns(col).AutoFit
Next col

' Форматируем заголовок
With ws.Range("A1:E1")
    .Font.Bold = True
    .Font.Color = RGB(255, 255, 255)
    .Interior.Color = RGB(31, 78, 121)
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlCenter

```

```
End With
```

```
' Форматируем столбцы с координатами как числа (7 знаков после запятой)
With ws.Columns(3)
```

```
    .NumberFormat = "0.0000000"
```

```
End With
```

```
With ws.Columns(4)
```

```
    .NumberFormat = "0.0000000"
```

```
End With
```

```
End Sub
```

```
' =====
```

```
' ОДНОКРАТНЫЙ ПОИСК (для проверки одного номера)
' =====
```

```
Sub GetSingleCadastralInfo()
```

```
Dim cadNum As String
```

```
Dim result As DadataResult
```

```
Dim message As String
```

```
cadNum = InputBox("Введите кадастровый номер (например: 77:02:0004008
```

```
If cadNum <> "" Then
```

```
    Application.StatusBar = "Запрос к Dadata..."
```

```
    result = GetAddressByCadastralNumber(cadNum)
```

```
    Application.StatusBar = ""
```

```
    message = "РЕЗУЛЬТАТ ПОИСКА" & vbCrLf & vbCrLf & _
```

```
        "Кадастровый номер: " & cadNum & vbCrLf & _
```

```
        "---" & vbCrLf & _
```

```
        "Адрес: " & result.address & vbCrLf & _
```

```
        "Координаты: " & result.latitude & ", " & result.longitude & vbCrLf & _
```

```
        "Ошибка: " & result.errorMsg
```

```
    MsgBox message, vbInformation, "Результат запроса"
```

```
End If
```

```
End Sub
```

■ ИНСТРУКЦИЯ ПО УСТАНОВКЕ

ШАГ 1: Откройте VBA редактор

Excel → Alt + F11 → Откроется VBA Editor

ШАГ 2: Добавьте модуль

Insert → Module → Скопируйте весь код выше

ШАГ 3: Установите ключи API

Найдите в коде строки:

```
Const API_KEY As String = "ВСТАВЬТЕ_СЮДА_API_КЛЮЧ"  
Const SECRET_KEY As String = "ВСТАВЬТЕ_СЮДА_SECRET_КЛЮЧ"
```

Замените на ваши ключи из <https://cabinet.dadata.ru>

ШАГ 4: Включите MSXML библиотеку

Tools → References → Microsoft XML, v3.0 → OK

ШАГ 5: Подготовьте данные

- Добавьте кадастровые номера в столбец А (начиная со строки 2)
- Формат: 77:02:0004008:4143 или 78:49:0001037:0001

ШАГ 6: Запустите!

F5 → Выберите Sub ProcessCadastralNumbers → Run

■ СТРУКТУРА ТАБЛИЦЫ

Столбец	Назначение	Пример
A	Входной: Кадастровый номер	77:02:0004008:4143
B	Выходной: Полный адрес	г Москва, ул Сухонская, д 11, кв 143
C	Выходной: Широта	55.8782557
D	Выходной: Долгота	37.65372
E	Статус: Результат обработки	✓ OK или ✗ Ошибка

■ ПРИМЕРЫ КАДАСТРОВЫХ НОМЕРОВ

Москва

77:02:0004008:0001 → г Москва, ул Сухонская, д 11 (дом)
77:02:0004008:4143 → г Москва, ул Сухонская, д 11, кв 143 (квартира)

Санкт-Петербург

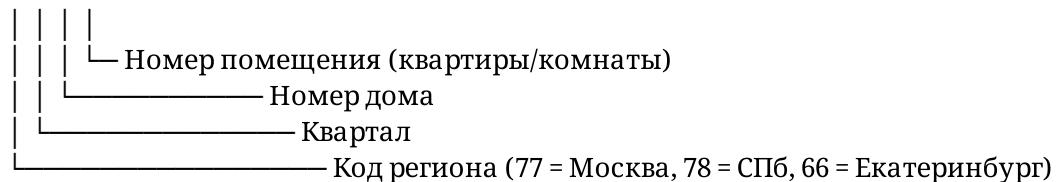
78:49:0001037:0001 → г Санкт-Петербург, пр Невский, д 1 (дом)
78:49:0001037:0567 → г Санкт-Петербург, пр Невский, д 1, кв 567 (квартира)

Екатеринбург

66:42:0004017:0001 → г Екатеринбург (пример)

Структура кадастрового номера

77:02:0004008:4143



⚠ ЧАСТЫЕ ПРОБЛЕМЫ И РЕШЕНИЯ

Проблема	Решение
"HTTP 401"	Проверьте API ключи - скопируйте их заново из кабинета
"HTTP 403"	Закончился лимит или блокировка IP. Проверьте https://cabinet.dadata.ru/api-stat
"Empty response"	Кадастровый номер не найден в базе. Проверьте его в демо-форме на https://dadata.ru
"N/A для адреса"	Этот номер нет в базе Dadata (покрытие ~60-70% РФ)
"Координаты 0,0"	Координаты не определены для этого номера
"Не включена MSXML"	Выполните ШАГ 4: Tools → References → Microsoft XML

□ ТАРИФ И ЛИМИТЫ

- Бесплатный план: 1000 запросов/месяц
- Платные тарифы: от 1₽ за 1000 запросов
- Проверить использование: <https://cabinet.dadata.ru/api-stat>
- Каждый запрос (включая ошибочные) списывается с лимита

□ БЫСТРЫЕ КОМАНДЫ

Команда	Описание	Запуск
ProcessCadastralNumbers	Обработать все кадастровые номера	F5
GetSingleCadastralInfo	Проверить один номер	Alt+F8

□ ПОЛЕЗНЫЕ ССЫЛКИ

- Документация API Dadata: <https://dadata.ru/api/>
- Личный кабинет: <https://cabinet.dadata.ru>
- Проверка статистики API: <https://cabinet.dadata.ru/api-stat>
- Демо-форма: <https://dadata.ru>
- Техподдержка: support@dadata.ru

✓ ЧЕКЛИСТ ПЕРЕД ЗАПУСКОМ

- [] Получены ключи API (<https://cabinet.dadata.ru>)
- [] Ключи вставлены в код (строки с Const API_KEY и Const SECRET_KEY)
- [] Включена MSXML библиотека (Tools → References)
- [] Кадастровые номера добавлены в столбец А (начиная со строки 2)
- [] Компьютер подключен к интернету
- [] Excel сохранен (чтобы не потерять макросы)

Готово? → F8 → ProcessCadastralNumbers → Run!

Версия: 2.0

Дата: 2026-01-21

Функция: Кадастровый номер → Адрес + Координаты

Статус: ✓ Готов к использованию