

**QUICKFIX FIELD GUIDE**

Power Query Recipes

Fast fixes, practical patterns, and troubleshooting moves for Excel and Power BI refresh problems.

Error fixes

Step-level errors, cell-level errors, type conversion, missing columns, and Excel error values.

Refresh fixes

File paths, credentials, gateway behavior, privacy levels, and service-ready sources.

Performance fixes

Fold early, reduce early, delay expensive operations, and avoid accidental full scans.

Model fixes

Merge safety, append drift, sort reliability, folder combine, and reusable functions.

START HERE

Quick Diagnostic Map



If you see...	Go to...	First thing to check
Yellow error pane, query will not load	Step-level error recipes	Error reason, message, detail, and the first failing Applied Step.
Error in a few cells	Cell-level error recipes	Column Quality, Keep Errors, and the exact value causing conversion failure.
Refresh works on your machine but not someone else's	Source and credential recipes	Local paths, permissions, privacy levels, and gateway mapping.
Merges multiply rows	Join safety recipes	Duplicate keys on the lookup side and mismatched data types.
Slow preview or refresh	Performance recipes	Filters/columns too late, sort too early, lost query folding, custom functions over large rows.
Folder combine breaks after new files arrive	Folder recipe	Hidden/temp files, sample file assumptions, header drift, and missing columns.

Golden triage

- 1. Find the first failing step.
- 2. Check source shape before that step.
- 3. Fix the cause, not just the symptom.
- 4. Refresh all, not only preview.

Use profiling early

Enable Column quality, Column distribution, and Column profile. For smaller datasets, switch profiling to the entire dataset before trusting error/null percentages.

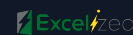
Recipe rule

After every fix, recheck row count, duplicate key count, error count, and a known source total. A query can refresh and still be wrong.

Fast mental model: each Applied Step returns a new table. When something breaks, inspect the table immediately before the failing step and ask: did the column, type, row order, or source availability change?

RECIPES 01-06

Source and Refresh Fixes



01 DataSource.NotFound

File moved, inaccessible path, wrong permissions, or user-specific drive.

FIX

- Replace local paths like `C:\Users\Name\...` with SharePoint/OneDrive/network paths.
- Use a parameter for the root folder and reference it in `File.Contents` or `Folder.Files`.
- Clear and reset credentials from Data source settings.

```
Source = Folder.Files(p_RootFolder)
```

02 Power BI Service Refresh Fails

Desktop refresh is not the final test.

FIX

- Check dataset credentials after publishing.
- Map on-premises sources through the gateway.
- Avoid sources that only your desktop session can see.
- Confirm privacy levels match the intended data combination.

TIP

Publish once early, then validate refresh before building many reports on top.

03 Formula.Firewall

Power Query is preventing accidental data leakage between sources.

FIX

- Set compatible privacy levels: Public, Organizational, or Private.
- Stage each source separately, then combine in a reference query.
- Do not pass private values into public web requests unless that is intentional and governed.

COMMON TRIGGER

A query both references another query and directly accesses a source in the same step chain.

04 Remote Host Closed Connection

Often API timeout, proxy, service limit, or large response problem.

FIX

- Ask whether the API supports paging and request smaller chunks.
- Filter by date/range parameters at the API level where possible.
- Retry later only after confirming it is not a repeatable data-size issue.

```
// Pattern: generate pages then combine
Pages = List.Generate(() => 1, each _ <= PageCount,
each _ + 1),
Tables = List.Transform(Pages, each fn_GetPage(_)),
Combined = Table.Combine(Tables)
```

05 Credential Prompts Keep Returning

Source identity changed or auth scope does not match.

FIX

- Data source settings > Clear Permissions, then reconnect.
- Use consistent source roots, especially for web and SharePoint URLs.
- For APIs, avoid concatenating the whole URL; use `Web.Contents` with `RelativePath` and `Query`.

06 Excel Workbook Bloats or Gets Slow

Too many worksheet loads and duplicate staging outputs.

FIX

- Load staging queries as connection only.
- Load large model tables to the Data Model instead of sheets where appropriate.
- Remove columns before loading; Power Query should not carry audit clutter unless needed.

RECIPES 07-13

Error and Type Fixes



07 Column Was Not Found

A step references a column that no longer exists.

FIX

- Click the step before the error and inspect actual column names.
- Remove stale rename/remove/type steps.
- For allowed schema drift, use missing-field options.

```
Table.SelectColumns(Source, {"Date", "Amount"},
MissingField.UseNull)
```

08 Wrong Type Inferred

Power Query can infer from early rows and miss later patterns.

FIX

- Delete automatic Changed Type if it chose incorrectly.
- Set types explicitly after cleanup.
- Use data profiling over the entire dataset for small/medium sources.

```
Typed = Table.TransformColumnTypes(Cleaned, {"Qty",
type number})
```

09 Date Locale Errors

Day/month and month/day are being interpreted with the wrong culture.

FIX

- Use Change Type > Using Locale.
- Specify culture in M when source format is stable.
- Validate by extracting day, month, and year after conversion.

```
Table.TransformColumnTypes(Source, {"InvoiceDate",
type date}, "en-GB")
```

10 Excel Errors Imported as Error

Cells contain #REF! , #DIV/0! , #NULL! , etc.

FIX

- Use Keep Errors first to audit affected rows.
- Replace errors only when a fallback value is valid.
- Use try to preserve business logic.

```
FinalRate = try [Standard Rate] otherwise [Special
Rate]
```

11 Need Custom Error Logic

Different errors require different handling.

FIX

Use try to turn the value into a record, then inspect HasError , Value , and Error .

```
Checked = try [Standard Rate],
Output =
if Checked[HasError]
then [Special Rate]
else Checked[Value]
```

12 Text Plus Number Fails

M does not silently concatenate mixed types.

FIX

Convert the number to text intentionally, preferably with culture if display matters.

```
"Total Sales: " & Number.ToText([Sales], "N2", "en-
US")
```

13 Non-breaking Spaces Break Matching

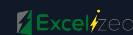
Trim/Clean may not remove every whitespace character.

FIX

```
Table.TransformColumns(
Source,
{"Customer", each Text.Trim(Text.Replace(_,
Character.FromNumber(160), " ")), type text}
)
```

RECIPES 14-20

Merge, Append, and Shape Fixes



14 Merge Multiplies Rows

The lookup table is not unique on the join key.

FIX

- Group lookup table by key and count rows.
- Resolve duplicates before merge or accept one-to-many intentionally.
- Make join keys the same data type on both sides.

```
Dupes = Table.SelectRows(
    Table.Group(Customers, {"CustomerID"}, {"Rows",
        each Table.RowCount(_, Int64.Type)}),
        each [Rows] > 1
    )
```

15 Merge Finds No Matches

Keys look similar but are not truly equal.

FIX

- Trim and clean both key columns.
- Check type: text "001" is not number 1 .
- Normalize case when matching text codes.

```
CleanKey =
    Text.Upper(Text.Trim(Text.From([CustomerID])))
```

16 Append Creates Null Columns

Append aligns by column name, not position.

FIX

- Standardize column names before append.
- Keep a schema mapping table for recurring vendor files.
- Use `Table.SelectColumns` to force expected output columns.

17 Remove Duplicates Keeps Wrong Row

Sort order is not guaranteed through duplicate removal.

FIX

- Rank by your tie-breaker and keep rank 1.
- Or sort within grouped nested tables before choosing first row.
- Buffer only after testing memory and folding impact.

```
Grouped = Table.Group(Source, {"ID"}, {"Rows", each
    Table.Sort(_, {"Modified", Order.Descending})})
```

18 Unpivot Breaks When New Months Arrive

Hard-coded selected period columns age badly.

FIX

Select stable identifier columns, then use **Unpivot Other Columns**. New month columns will flow into rows automatically.

19 Pivot Fails or Aggregates Unexpectedly

More than one value exists per row/category intersection.

FIX

- Group first to the intended grain.
- Choose an explicit aggregation when pivoting.
- Check duplicate combinations before pivot.

20 Nested Values Show as Errors

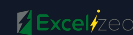
Table/list/record values can fail on load or after firewall buffering.

FIX

- Expand nested tables/records before loading.
- Remove columns containing lists/tables if not needed.
- Set non-Any data types where possible.

RECIPES 21-27

Performance and Folding Fixes

**21 Refresh Is Too Slow**

Power Query is doing too much work locally.

FIX

- Filter rows early.
- Choose/remove columns early.
- Delay expensive operations like sort, group, pivot, and custom functions.
- Use the best connector for the source, such as SQL Server connector over generic ODBC when possible.

22 View Native Query Disappears

A step probably broke query folding.

FIX

- Move foldable filters and column selection before the breaking step.
- Push complex logic into a database view when shared or large.
- Use query folding indicators where available.

REMEMBER

Partial folding can still be useful. Get the biggest filters folded first.

23 Native SQL Makes Later Steps Slow

Later transformations may not fold after native query use.

FIX

- Put as much transformation logic as possible in the SQL statement or a view.
- Use `Value.NativeQuery` with parameters where appropriate.
- Use read-only least-privilege database credentials.

24 Table.Buffer Made It Worse

Buffering can block folding and consume memory.

FIX

- Remove buffer before foldable filters.
- Buffer small lookup lists only after testing.
- Do not buffer large source tables casually.

```
LookupIDs = List.Buffer(DimCustomer[CustomerID])
```

25 Preview Is Painfully Slow

The editor is scanning too much while authoring.

FIX

- Temporarily Keep First Rows while developing.
- Remove that temporary step before final refresh.
- Develop expensive custom parsing on a small reference sample.

26 DirectQuery Feels Broken

Non-folding steps are usually the culprit.

FIX

- For DirectQuery and Dual tables, design for folding from the start.
- Move transformations to SQL views or source-side logic.
- Avoid local-only transformations in DirectQuery tables.

27 Incremental Refresh Pulls Too Much Data

Date filter does not fold or uses the wrong field.

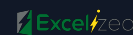
FIX

- Use `RangeStart` and `RangeEnd` Date/Time parameters.
- Filter on an unchanging Date/Time field.
- Confirm the filter folds before publishing.

```
Table.SelectRows(Source, each [OrderDate] >= RangeStart and [OrderDate] < RangeEnd)
```

RECIPES 28-32

Folder, Function, and API Fixes



28 Folder Combine Reads Bad Files

Temporary, hidden, old, or wrong-extension files entered the pipeline.

FIX

```
Files =
    Table.SelectRows(
        Folder.Files(p_FolderPath),
        each [Extension] = ".xlsx"
            and [Attributes]?[Hidden]? <> true
            and not Text.StartsWith([Name], "~$")
    )
```

29 Transform Sample File Is Weird

The sample file controls the generated function.

FIX

- Choose a representative sample file.
- Move header cleanup, type fixes, and sheet/table navigation into the transform function.
- Keep source file name in final output when audit matters.

30 Custom Function Is Slow

It may be invoked row-by-row against too much data.

FIX

- Filter rows before invoking the function.
- Invoke once per file/page/object, not once per fact row.
- Return only required columns from the function.

31 Dynamic Web URL Breaks Credentials

Power Query cannot classify a constantly changing full URL cleanly.

FIX

```
Web.Contents(
    "https://api.contoso.com",
    [
        RelativePath = "v1/orders",
        Query = [from = Date.ToText(p_StartDate,
            "yyyy-MM-dd")]
    ]
)
```

32 Parameters Are Scattered

Paths, servers, date windows, and thresholds are hard-coded.

FIX

- Create parameters from Manage Parameters.
- Name them with `p_`, such as `p_Server` or `p_FolderPath`.
- Use parameters in connector dialogs where supported.

Reusable function habit: build the transformation as a normal query first, test it on one representative input, then convert to a function once the output shape is stable.

BONUS PAGES

Ready-to-Use Patterns



Safe select columns

```
Table.SelectColumns(
    Source,
    {"CustomerID", "OrderDate", "Amount"},
    MissingField.UseNull
)
```

Safe remove columns

```
Table.RemoveColumns(
    Source,
    {"Temp", "Notes"},
    MissingField.Ignore
)
```

Anti-join reconciliation

```
MissingProducts =
    Table.NestedJoin(
        Sales, {"ProductID"},
        Products, {"ProductID"},
        "Product",
        JoinKind.LeftAnti
    )
```

Keep only error rows

```
ErrorsOnly = Table.SelectRowsWithErrors(Source,
{"Amount"})
```

Normalize text key

```
Text.Upper(
    Text.Trim(
        Text.Replace(Text.From([Key]),
            Character.FromNumber(160), " ")
    )
)
```

Append many staged tables

```
Table.Combine({stg_Jan, stg_Feb, stg_Mar})
```

Final QA Checklist

- First failing Applied Step identified and fixed.
- Data types explicitly set after cleanup.
- Required columns selected intentionally.
- Missing columns handled intentionally.
- Merge keys trimmed, cleaned, typed, and duplicate-checked.
- Append column names standardized.
- Folder combines ignore hidden and temp files.
- File paths are parameterized or shared-access paths.
- Privacy levels and credentials checked.
- Power BI service refresh tested after publish.
- Query folding checked for database/DirectQuery sources.
- Row counts and totals reconcile to source.

REFERENCE

Sources Checked Online



These Microsoft Learn pages were checked while preparing this edition. They are useful follow-up references when you need product-specific details.

Topic	Source
Power Query errors	https://learn.microsoft.com/en-us/power-query/dealing-with-errors
try , otherwise , catch	https://learn.microsoft.com/en-us/power-query/error-handling
Best practices	https://learn.microsoft.com/en-us/power-query/best-practices
Common authoring issues	https://learn.microsoft.com/en-us/power-query/common-issues
Query folding overview	https://learn.microsoft.com/en-us/power-query/query-folding-basics
Data Privacy Firewall	https://learn.microsoft.com/en-us/power-query/data-privacy-firewall
Power BI folding guidance	https://learn.microsoft.com/en-us/power-bi/guidance/power-query-folding
Incremental refresh	https://learn.microsoft.com/en-us/power-bi/connect-data/incremental-refresh-configure

Excelized branding
Uses the local logo file `Excelized r3-01_white.png` from this project folder.

Format
A4 PDF, quick-reference layout, recipe numbers, and copyable M snippets.

Edition note
This is a rebuilt branded edition, not a direct edit of the older PDF file.