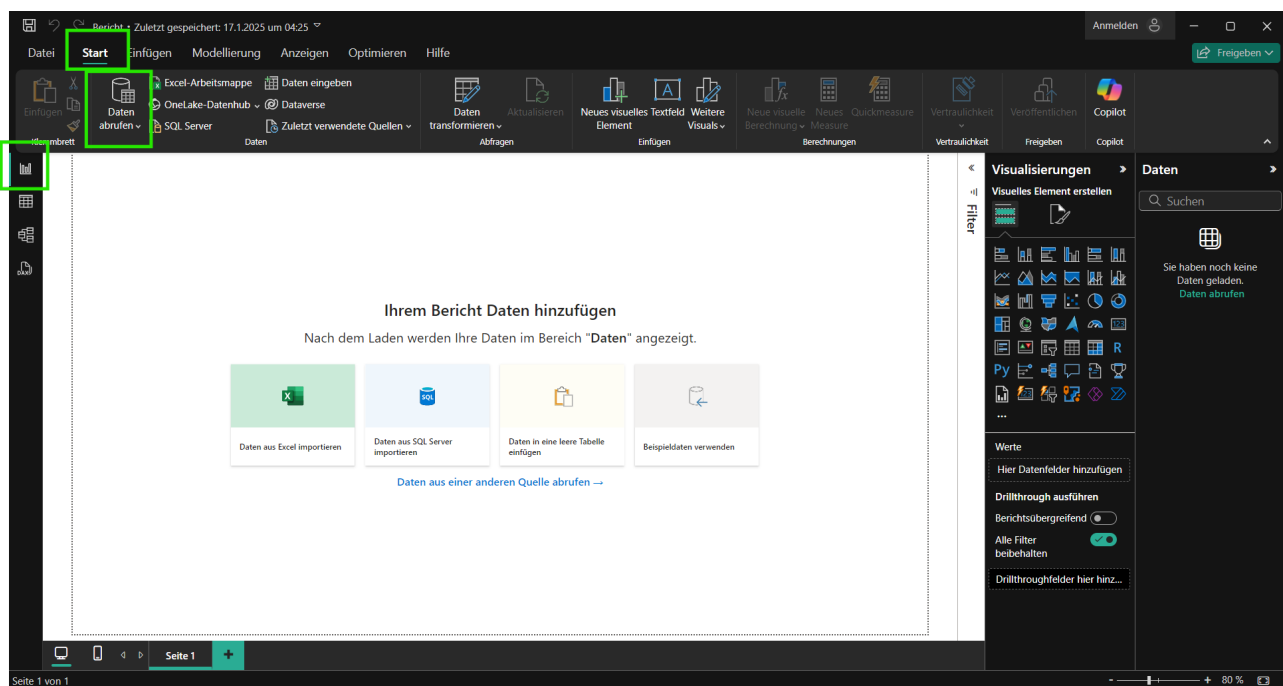


Daten aus einer XML-Datei abrufen

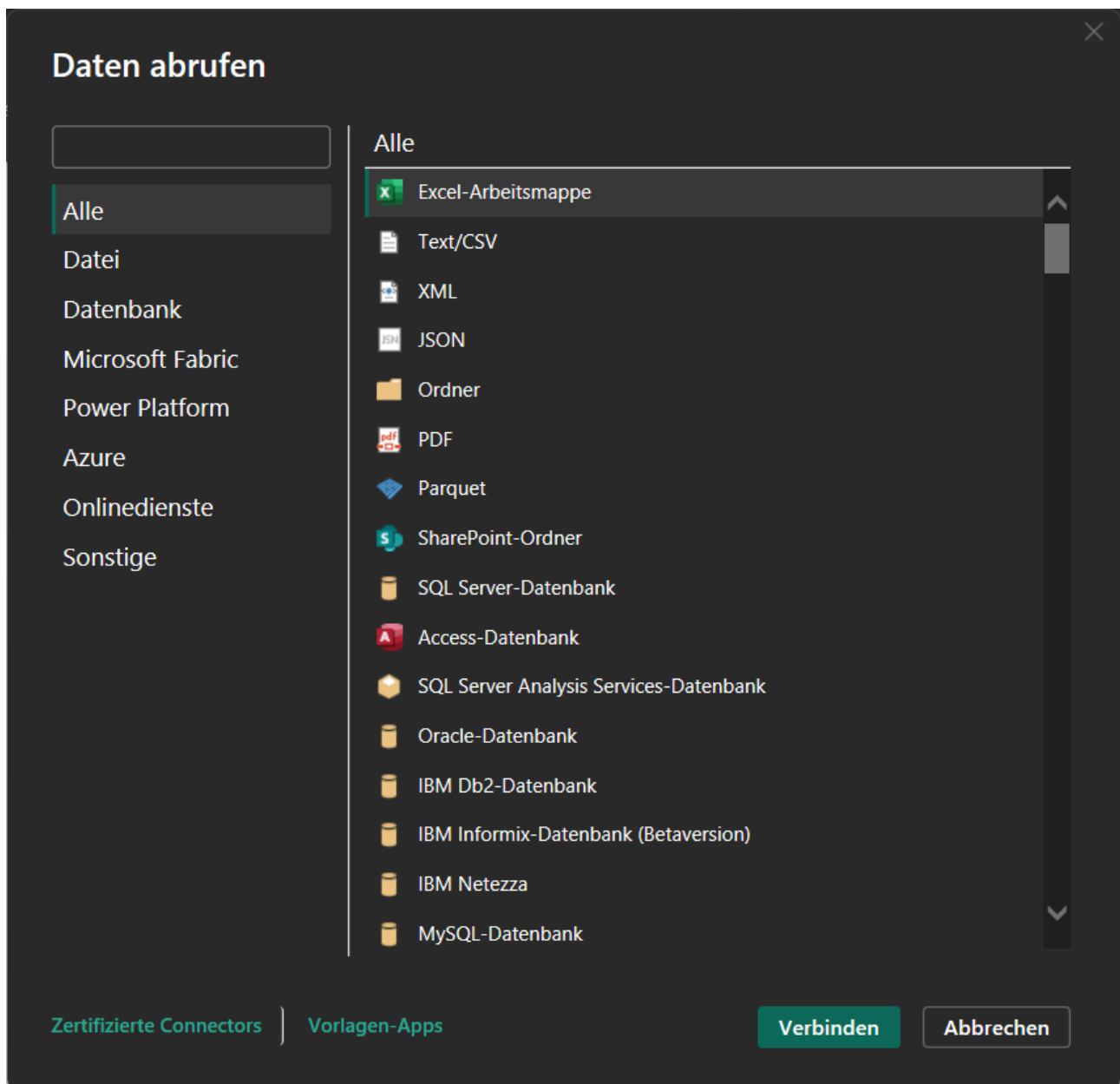
Inhalt: In diesem Dokument wird beschrieben, wie man Daten, die in einer XML-Datei gespeichert sind, in Power BI Desktop abrufen kann.

Bemerkung: Der im Folgenden beschriebene Weg funktioniert in der **Berichtsansicht**, **Tabellenansicht** und **Modellansicht** auf die gleiche Weise. Der Weg wird exemplarisch aus der **Berichtsansicht** heraus beschrieben.

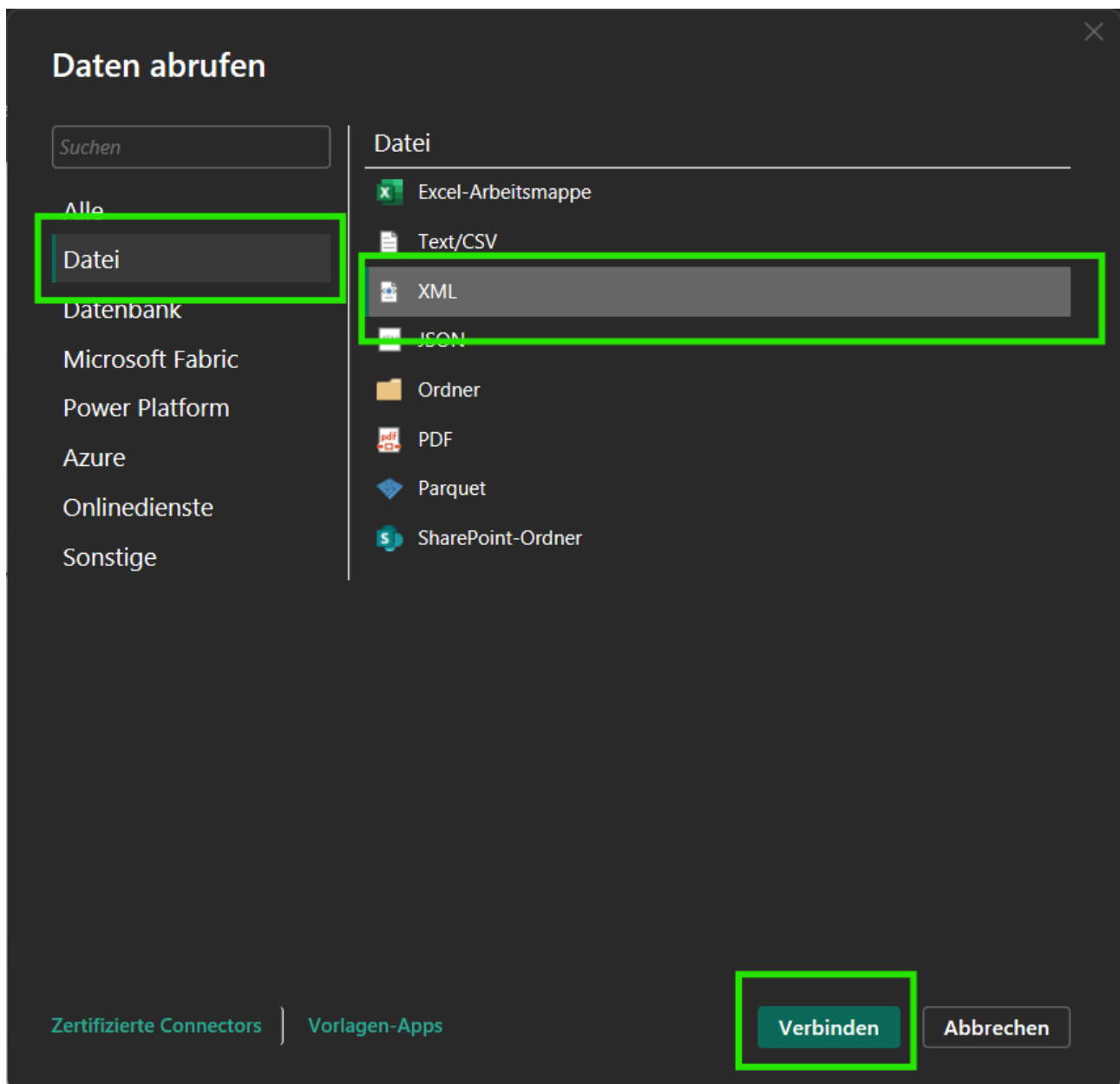
1. In der **Berichtsansicht**, wähle man in der Registerkarte **Start** die Option **Daten abrufen**.



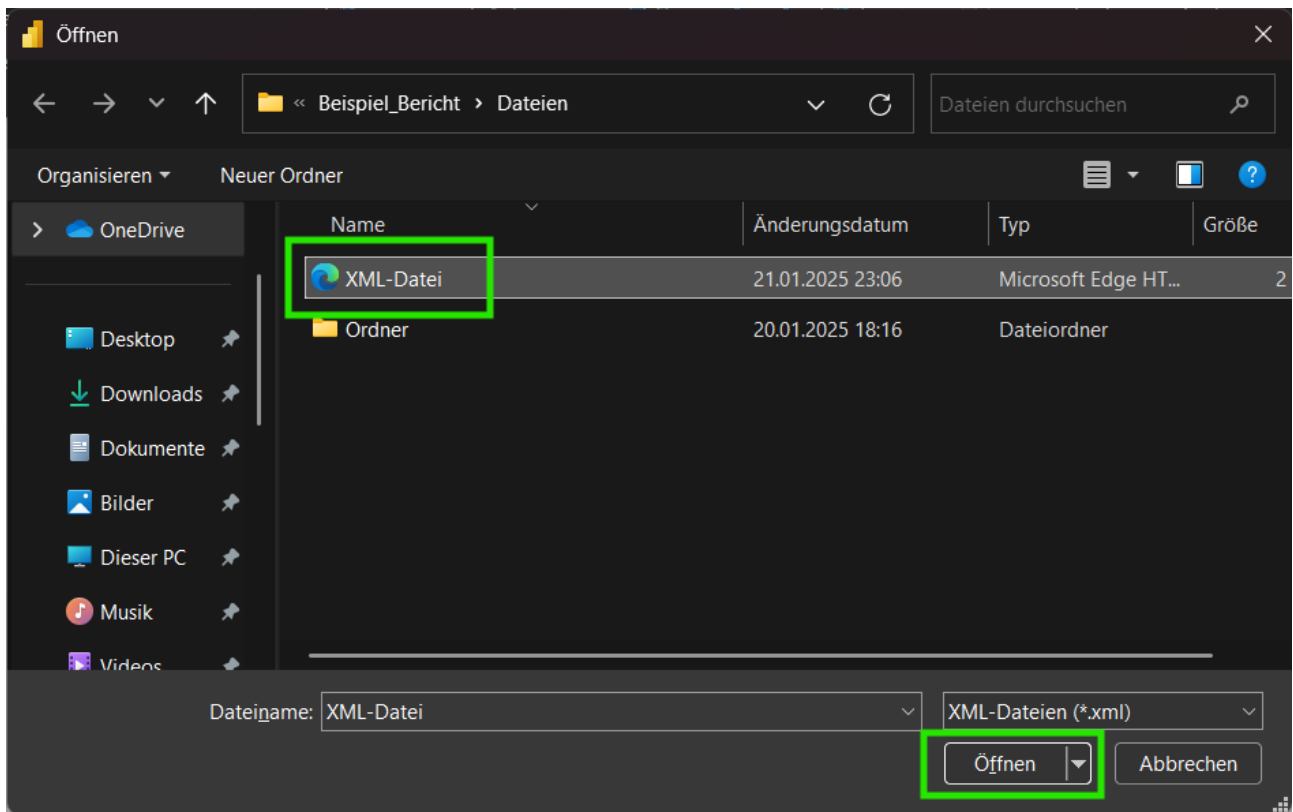
2. Es öffnet sich das Fenster **Daten abrufen**.



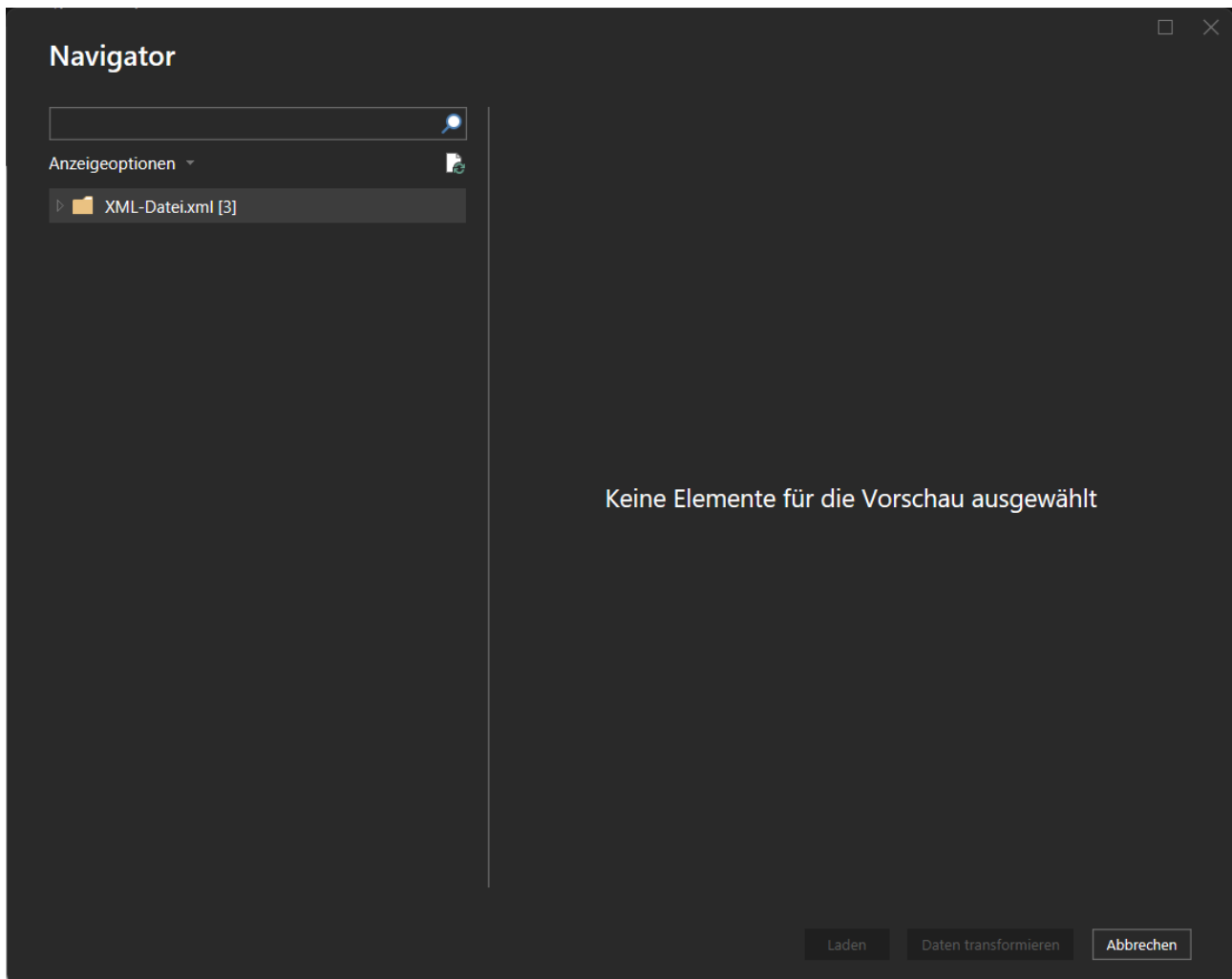
3. In der Kategorie **Datei** wählt man die Option **XML** aus und klickt anschließend auf **Verbinden**. Alternativ kann man die Option **XML** in der Kategorie **Alle** auswählen.



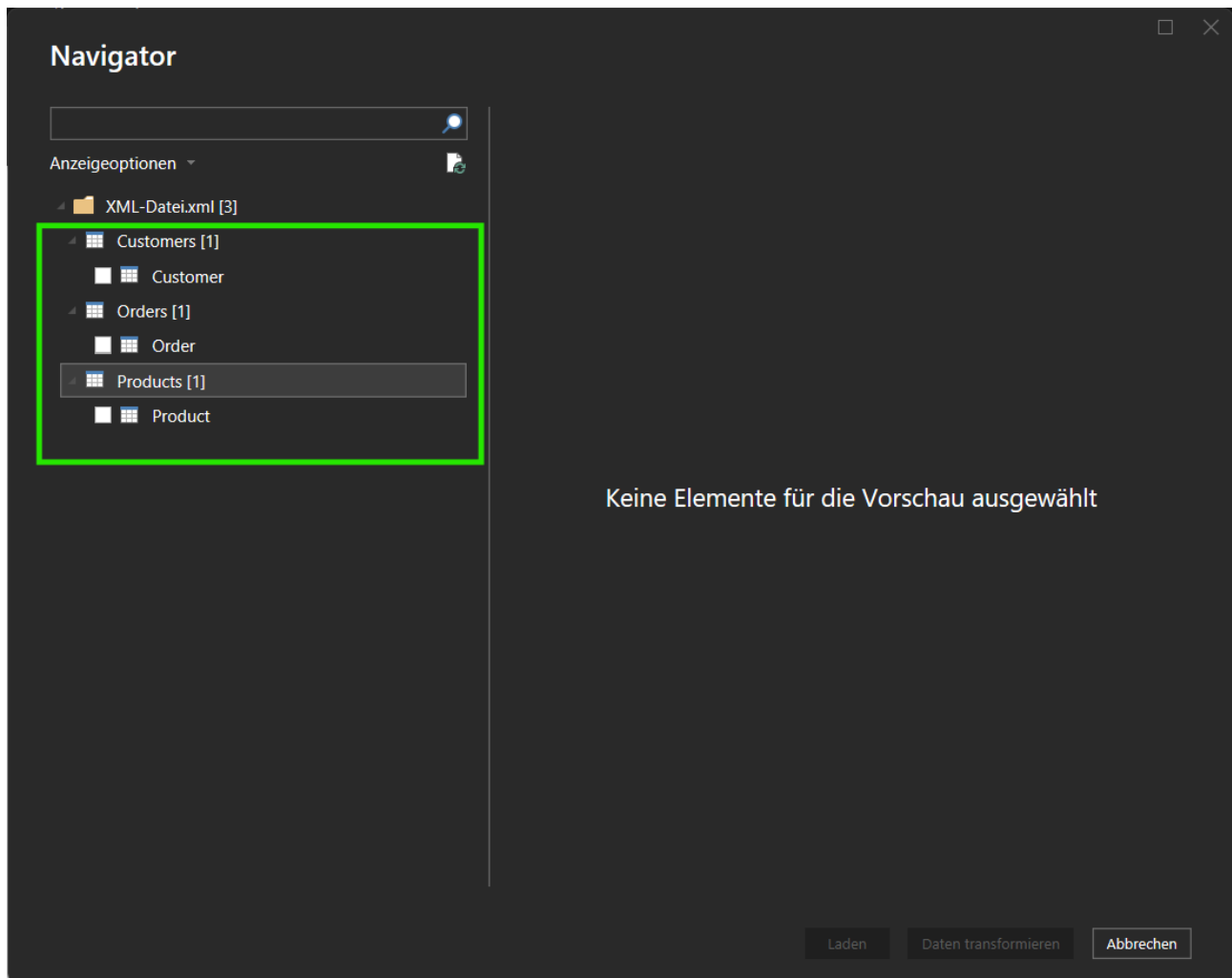
4. Es öffnet sich das Fenster **Öffnen**. In diesem navigiert man zu dem Speicherort, an dem die XML-Datei abgelegt ist und wählt anschließend **Öffnen**.



5. Es öffnet sich das Fenster **Navigator**.



6. Links im Fenster **Navigator** sieht man die XML-Datei mit der man sich verbinden möchte. Wenn man den Pfeil links vom Ordner Symbol wählt, dann werden einem alle (von Power BI erkannten) Tabellen angezeigt. Jede der Tabellen kann durch das wählen des Kontrollkästchens ausgewählt werden. Man kann auch alle Tabellen auswählen, dann wird jede einzelne Tabelle in das Power BI-Modell geladen. Hat man eine Tabelle ausgewählt, erhält man rechts eine Vorschau der Daten, wie sie in Power BI geladen werden würden, wenn man keine weitere Verarbeitung der Daten mehr vornimmt. Wählt man hingegen mehrere Tabellen gleichzeitig aus, so wird einem in der Vorschau die zuletzt ausgewählte Tabelle angezeigt.



Navigator

Anzeigeoptionen ▾

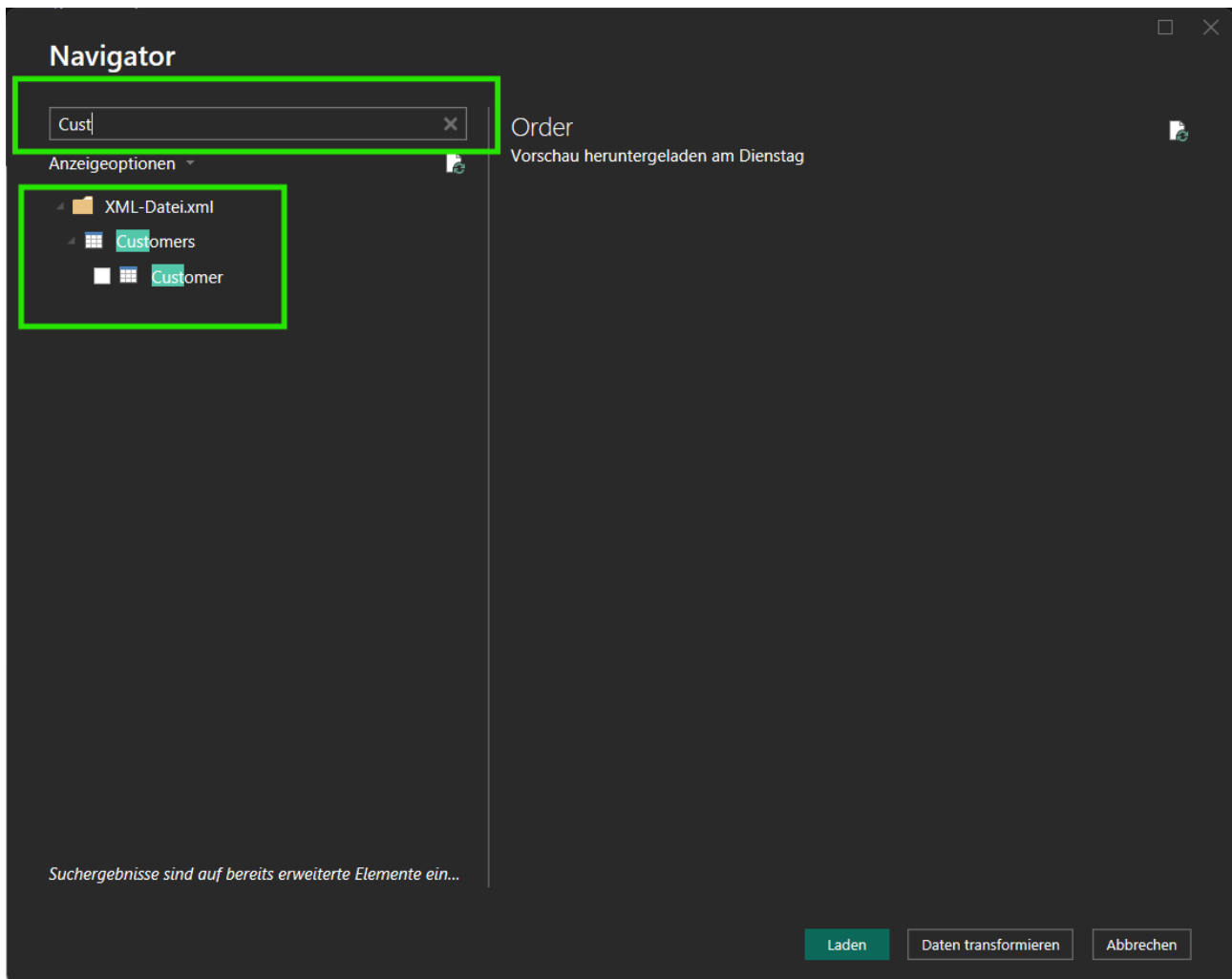
- XML-Datei.xml [3]
- Customers [1]
- Customer
- Orders [1]
 - ☒ Order
- Products [1]
- Product

Order
Vorschau heruntergeladen am Dienstag

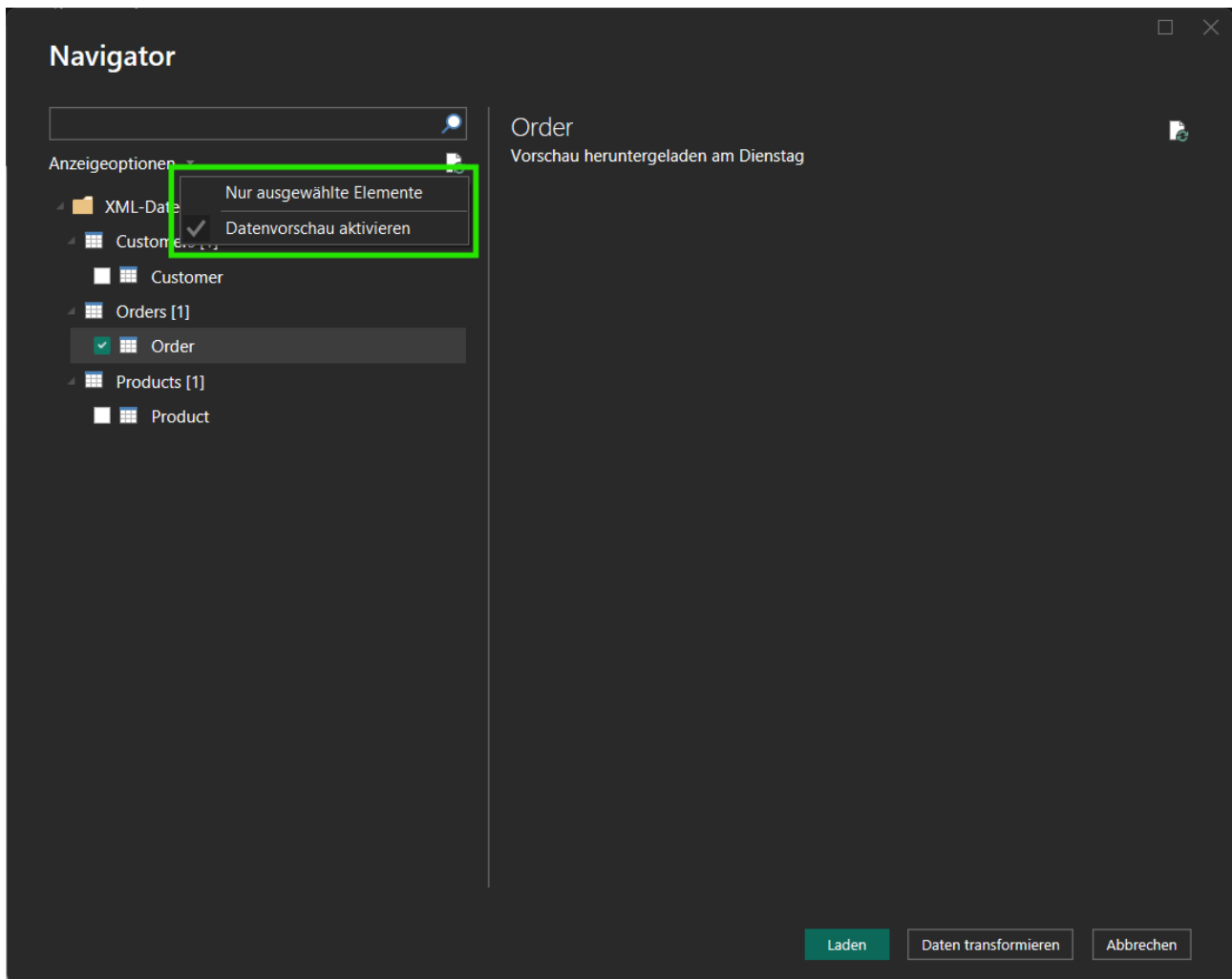
OrderID	OrderDate	CustomerID	TotalAmount
1	01.01.2025	101	15000
2	02.01.2025	102	20000
3	03.01.2025	101	30000

Laden Daten transformieren Abbrechen

7. Hat man eine große Anzahl an Tabellen in der Datei, so kann man mit Hilfe des Suchfelds nach der Tabelle suchen, die man in sein Modell importieren möchte.



8. Unter **Anzeigeoptionen** kann man einstellen, dass nur die Elemente angezeigt werden, die man ausgewählt hat. Ebenfalls lässt sich hier drüber, die Vorschau der Daten deaktivieren.



9. Klickt man das Symbol unter rechts unter dem Suchfeld, so wird die XML-Datei aktualisiert.
10. Wählt man **Laden** aus, so wird/werden die Tabelle/Tabellen, wie sie in der Vorschau zu sehen sind, in das Power BI-Modell geladen. Die Tabelle/Tabellen findet man in der **Berichtsansicht** anschließend im Bereich **Daten**.

Navigator

Anzeigeoptionen

- XML-Datei.xml [3]
- Customers [1]
 - ☒ Customer
- Orders [1]
 - ☒ Order
- Products [1]
 - ☒ Product

Order

OrderID	OrderDate	CustomerID	TotalAmount
1	01.01.2025	101	15000
2	02.01.2025	102	20000
3	03.01.2025	101	30000

Laden Daten transformieren Abbrechen

Bericht • Zuletzt gespeichert: 20.1.2025 um 17:59

Datei **Start** Einfügen Modellierung Anzeigen Optimieren Hilfe

Visualisierungen

Visuelles Element erstellen

Filter

Werte

Hier Datenfelder hinzufügen

Drillthrough ausführen

Berichtsübergreifend

Alle Filter beibehalten

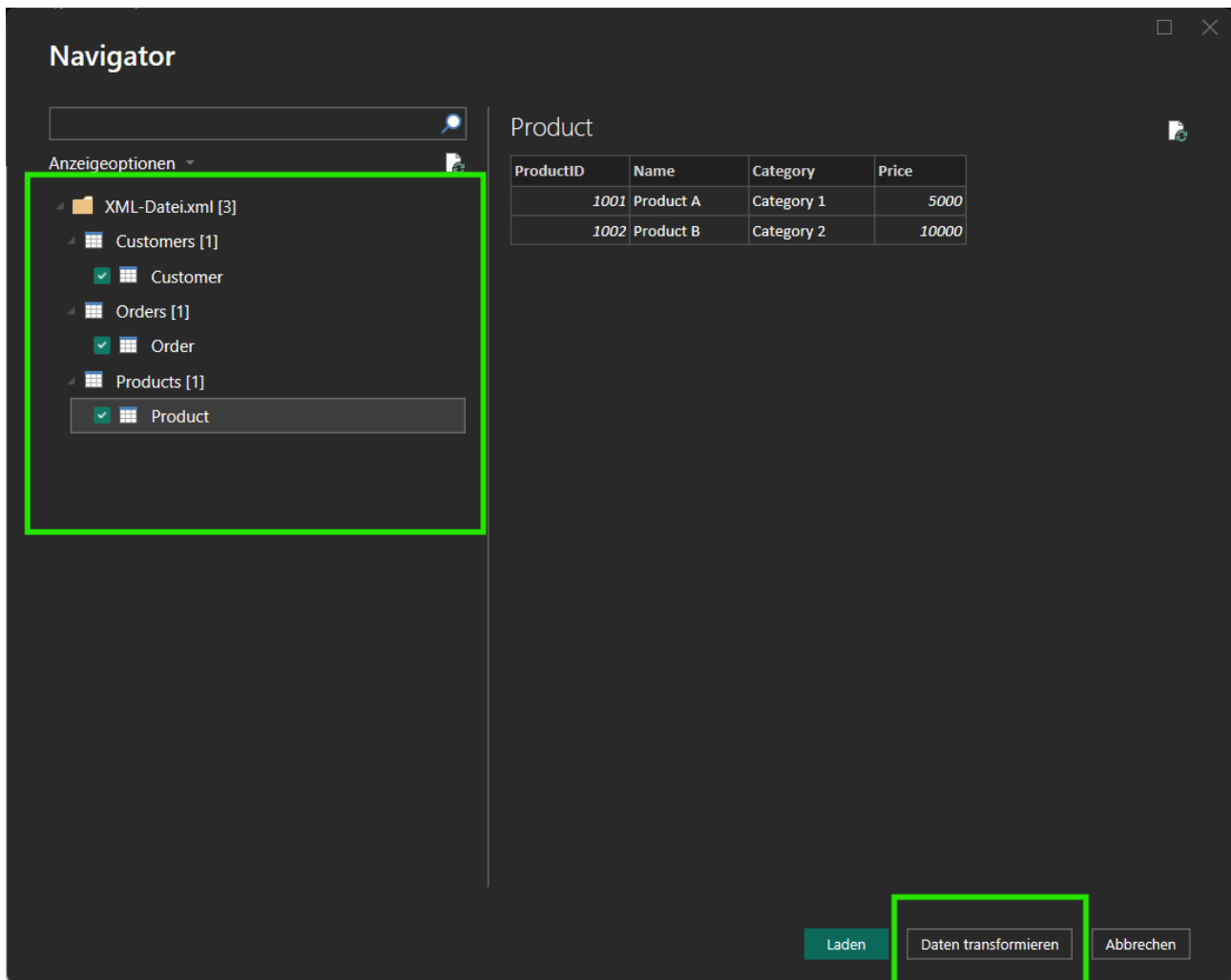
Drillthroughfelder hier hinz...

Daten

- Customer
 - ☐ CustomerID
 - ☐ Email
 - ☐ Name
 - ☐ Phone
- JSON-Datei
- Order
 - ☐ CustomerID
 - ☐ OrderDate
 - ☐ OrderID
 - ☐ TotalAmount
- Page001
- Product
 - ☐ Category
 - ☐ Name
 - ☐ Price
 - ☐ ProductID
- random_integers
- Sheet1
- Tab_Datei

Seite 1 von 1

11. Sollten die Daten hingegen noch weitere Verarbeitung benötigen, so wählt man **Daten transformieren** aus und es öffnet sich der **Power Query-Editor**. Die Abfrage für die Tabellen findet man links im **Power Query-Editor** im Bereich **Abfragen**. Hat man die Daten in die gewünschte Form gebracht, wählt man in der Registerkarte **Start** des **Power Query-Editors** die Option **Schließen und übernehmen** aus. Die korrigierten Tabellen werden dann in das Power BI-Datenmodell geladen. Man findet die Tabellen wieder in der **Berichtsansicht** im Bereich **Daten**.



Bericht

Start Transformieren Spalte hinzufügen Ansicht Extras Hilfe

Schließen und übernehmen
Schließen

Neue Quelle
Neue Abfrage

Zuletzt verwendete Quellen
Neue Abfrage

Daten eingeben

Datenquelleneinstellungen

Parameter verwalten
Parameter

Vorschau aktualisieren
Abfrage

Eigenschaften
Erweiterter Editor
Verwalten

Spalten auswählen
Spalten verwalten

Spalten entfernen
Spalten verwalten

Zeilen beibehalten
Zeilen verringern

Zeilen entfernen
Zeilen verringern

Sortieren

Spalte teilen
Gruppieren nach

Datentyp: Ganze Zahl
Erste Zeile als Überschriften verw.
Werte ersetzen
Transformieren

Abfragen [8]

random_integers

Sheet1

Txt_Datei

Page001

JSON-Datei

Customer

Order

Product

Table.TransformColumnTypes(Table9,({"ProductID", Int64.Type}, {"Name", type text}, {"Category", type text}, {"Price", Int64.Type}))

ProductID	Name	Category	Price
1	1001 Product A	Category 1	5000
2	1002 Product B	Category 2	10000

Abfrageeinstellungen

EIGENSCHAFTEN

Name
Product

Alle Eigenschaften

ANGEWENDETE SCHRITTE

Quelle

Navigation

Geänderter Typ

4 SPALTEN, 2 ZEILEN Spaltenprofilierung basierend auf obersten 1000 Zeilen

VORSCHAU HERUNTERGELADEN UM 05:55

Bericht

Start Transformieren Spalte hinzufügen Ansicht Extras Hilfe

Schließen und übernehmen
Schließen

Neue Quelle
Neue Abfrage

Zuletzt verwendete Quellen
Neue Abfrage

Daten eingeben

Datenquelleneinstellungen

Parameter verwalten
Parameter

Vorschau aktualisieren
Abfrage

Eigenschaften
Erweiterter Editor
Verwalten

Spalten auswählen
Spalten verwalten

Spalten entfernen
Spalten verwalten

Zeilen beibehalten
Zeilen verringern

Zeilen entfernen
Zeilen verringern

Sortieren

Spalte teilen
Gruppieren nach

Datentyp: Ganze Zahl
Erste Zeile als Überschriften verw.
Werte ersetzen
Transformieren

Abfragen [8]

random_integers

Sheet1

Txt_Datei

Page001

JSON-Datei

Customer

Order

Product

Table.TransformColumnTypes(Table9,({"ProductID", Int64.Type}, {"Name", type text}, {"Category", type text}, {"Price", Int64.Type}))

ProductID	Name	Category	Price
1	1001 Product A	Category 1	5000
2	1002 Product B	Category 2	10000

Abfrageeinstellungen

EIGENSCHAFTEN

Name
Product

Alle Eigenschaften

ANGEWENDETE SCHRITTE

Quelle

Navigation

Geänderter Typ

4 SPALTEN, 2 ZEILEN Spaltenprofilierung basierend auf obersten 1000 Zeilen

VORSCHAU HERUNTERGELADEN UM 05:55

Report - Last saved: 20.1.2025 at 17:59

File Start Einfügen Modellierung Anzeigen Optimieren Hilfe

Excel-Arbeitsmappe Daten eingeben
OneLake-Datenhub Dataverse
SQL Server Daten
Zuletzt verwendete Quellen

Daten transformieren Abfragen
Aktualisieren
Neues visuelles Textfeld Element
Weitere Visuals

Neue visuelle Berechnung Measure
Neues Quickmeasure
Berechnungen

Vertraulichkeit
Veröffentlichen
Copilot

Visualisierungen

Visuelles Element erstellen

Filter

Werte

Hier Datenfelder hinzufügen

Drillthrough ausführen

Berichtsübergreifend

Alle Filter beibehalten

Drillthroughfelder hier hinz...

Visuals mit Ihren Daten erstellen

Wählen Sie Felder aus, oder ziehen Sie sie aus dem Bereich "Daten" auf die Berichtscanvas.

Daten

Suchen

- Customer
 - CustomerID
 - Email
 - Name
 - Phone
- JSON-Datei
- Order
 - CustomerID
 - OrderDate
 - OrderID
 - TotalAmount
- Page001
- Product
 - Category
 - Name
 - Price
 - ProductID
- random_integers
- Sheet1
- Txt_Datei