

In-Class Activity: Building the Bookworm Books Database in Microsoft Access

Step-by-step student guide | Module: [code and title] | Name: [name and ID] | Date: [date]

1. Overview

In the design report you analysed the **Bookworm Books** scenario, drew an ERD and normalised the order data to Third Normal Form (3NF). Both routes gave the same four tables. In this activity you build that design as a working database in **Microsoft Access**, then test it to check that it behaves the way the design says it should.

Item	Details
Time	About 90 minutes. A suggested time is shown against each task.
Working	Individually, or in pairs with one keyboard. Swap roles after Task 3.
Software	Microsoft Access (Microsoft 365, 2021, 2019 or 2016). The ribbon looks slightly different in each version, but the steps are the same.
Starting point	The final relational schema from the design report: CUSTOMER, ORDER, BOOK and ORDER_ITEM.
What you hand in	The file <code>BookwormBooks.accdb</code> , eight screenshots (see Section 10) and a short written explanation of each screenshot.

By the end of the activity you will be able to

- turn a 3NF design into Access tables with suitable data types and primary keys;
- create a **composite primary key** for a linking table;
- create one-to-many relationships with **referential integrity** enforced;
- enter data in the correct order and explain why the order matters;
- test the design, and show with a query that no information was lost by normalising.

2. The design you are implementing

Table	Columns	Primary key	Foreign key
CUSTOMER	CustomerID, CustomerName, CustomerEmail	CustomerID	none
ORDER	OrderID, OrderDate, CustomerID	OrderID	CustomerID refers to CUSTOMER
BOOK	BookID, Title, Price	BookID	none
ORDER_ITEM	OrderID, BookID, Quantity	OrderID + BookID	OrderID refers to ORDER; BookID refers to BOOK

Naming and data-type decisions (already made for you)

- **Prefix table names with `tbl`** and queries with `qry`, so you can tell them apart in the Navigation Pane (for example `tblCustomer`).

- **ORDER becomes tblOrder.** ORDER is a reserved word in SQL (as in `ORDER BY`), and a table with that name has to be wrapped in square brackets in every query.
- **IDs are Short Text** (C1, B1, O101), so that your data matches the paper order forms exactly. Many real systems use AutoNumber instead.
- **A foreign key must have the same data type and size as the primary key it points to.** If they differ, Access will not let you create the relationship.
- Use **Currency** for Price, **Date/Time** for OrderDate and **Number** for Quantity.

3. Task 1: Create the database (5 minutes)

1. Open Microsoft Access. On the start screen, click **Blank database** (some versions call it Blank desktop database).
2. In the **File Name** box, type `BookwormBooks.accdb`. Click the folder icon and choose where to save it (your network drive, OneDrive or a USB drive, not the desktop of a shared computer).
3. Click **Create**. Access opens with an empty table called Table1 in Datasheet View.
4. We will build tables in Design View instead, so right-click the **Table1** tab and choose **Close**. If Access asks whether to save, click **No**.

Tip If a yellow Security Warning bar appears under the ribbon, click **Enable Content**.

4. Task 2: Create the four tables (25 minutes)

Build the tables **parents first**: `tblCustomer`, `tblBook`, `tblOrder`, then `tblOrderItem`. A foreign key can only point at a table that already exists. Use the same method for each table:

1. Click the **Create** tab, then **Table Design** (in the Tables group).
2. In the grid, type each **Field Name** and choose its **Data Type** from the drop-down list.
3. Click a field name, then set its **Field Properties** in the lower pane (Field Size, Required and so on).
4. To set the primary key, click the grey **row selector** to the left of the field name, then click **Primary Key** on the Table Design tab. A key icon appears.
5. Press **Ctrl+S**, type the table name, and click **OK**. Close the table.

4.1 tblCustomer

Field Name	Data Type	Field Size	Other settings
CustomerID	Short Text	10	Primary key. Required: Yes
CustomerName	Short Text	50	Required: Yes
CustomerEmail	Short Text	100	Required: Yes

Checkpoint A key icon is next to `CustomerID` only. Take **Screenshot 1** (the table in Design View) before you close it.

4.2 tblBook

Field Name	Data Type	Field Size / Format	Other settings
BookID	Short Text	10	Primary key. Required: Yes
Title	Short Text	100	Required: Yes
Price	Currency	Format: Currency, 2 decimal places	Required: Yes. Validation Rule: >=0

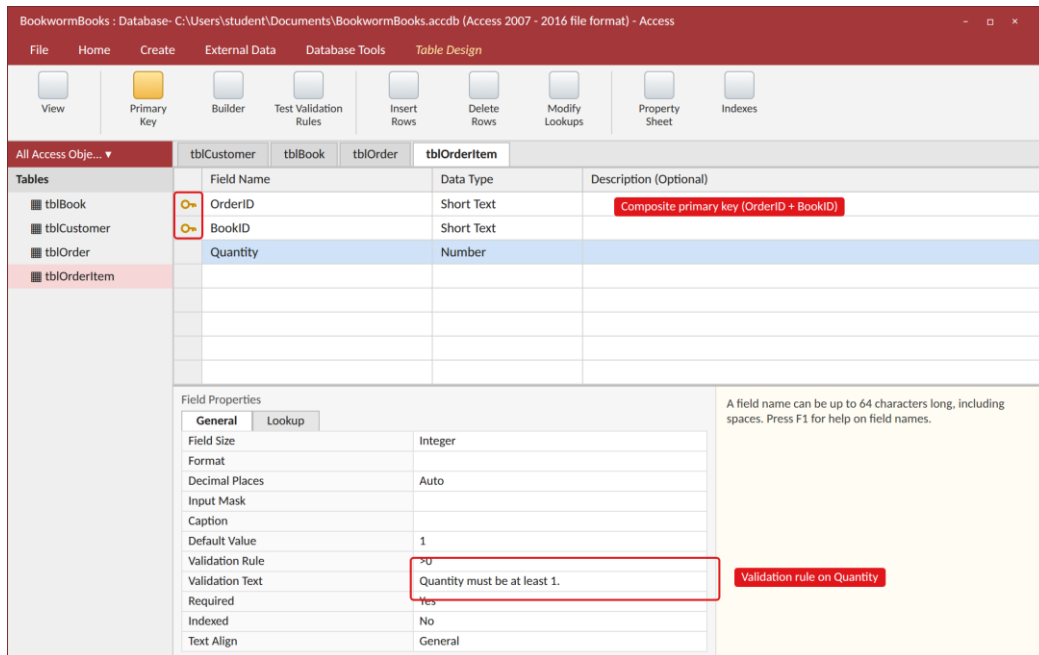
4.3 tblOrder

Field Name	Data Type	Field Size / Format	Other settings
OrderID	Short Text	10	Primary key. Required: Yes
OrderDate	Date/Time	Format: dd-mm-yyyy	Required: Yes
CustomerID	Short Text	10	Required: Yes. This is the foreign key , so its type and size must match tblCustomer.CustomerID

4.4 tblOrderItem (the linking table)

Field Name	Data Type	Field Size / Format	Other settings
OrderID	Short Text	10	Part of the primary key. Foreign key to tblOrder
BookID	Short Text	10	Part of the primary key. Foreign key to tblBook
Quantity	Number	Field Size: Integer	Required: Yes. Default Value: 1. Validation Rule: >0. Validation Text: Quantity must be at least 1.

Making a composite primary key. Click the row selector of `OrderID`, hold down **Ctrl**, click the row selector of `BookID`, then click **Primary Key**. Both rows should now show a key icon.

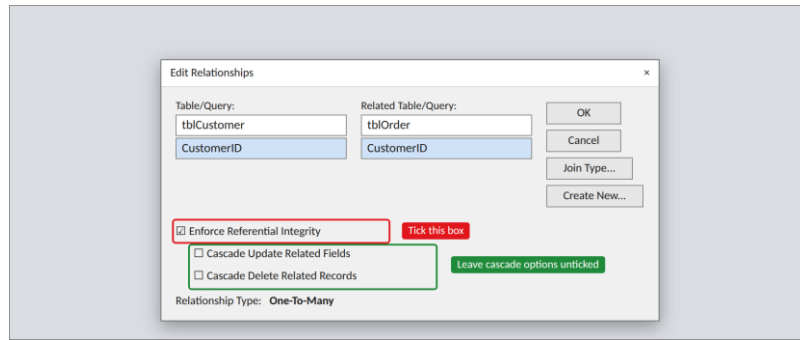


What your screen should look like: `tblOrderItem` with a composite primary key. (Illustration only. Your version of Access may look slightly different.)

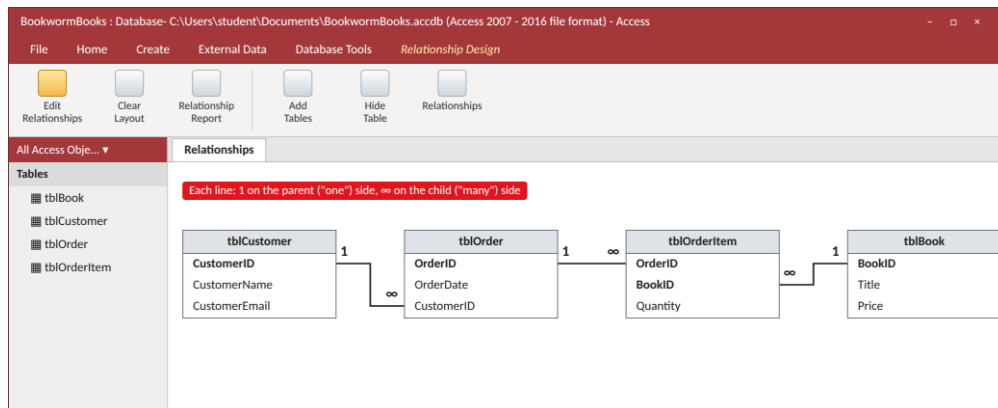
Checkpoint Why is the key made of **two** fields? Because the same book can appear on many orders, and an order can contain many books, but the same book should not appear twice on one order. Take **Screenshot 2** before closing the table.

5. Task 3: Create the relationships (15 minutes)

1. **Close all open tables.** Access cannot create relationships on a table that is open.
2. Click **Database Tools**, then **Relationships**. If no table list appears, click **Add Tables** (or **Show Table**) on the Relationship Design tab.
3. Select all four tables and click **Add Selected Tables**. Close the list. Drag the title bars so the tables sit left to right as: `tblCustomer`, `tblOrder`, `tblOrderItem`, `tblBook`.
4. Drag **CustomerID** from `tblCustomer` and drop it onto **CustomerID** in `tblOrder`. Always drag **from the "one" side (the primary key) to the "many" side (the foreign key)**. The Edit Relationships dialog opens.
5. Check that Table/Query shows `tblCustomer` and Related Table/Query shows `tblOrder`. Tick **Enforce Referential Integrity**. Leave both Cascade boxes **unticked**. Relationship Type should read **One-To-Many**. Click **Create**.
6. Repeat for `tblOrder.OrderID` onto `tblOrderItem.OrderID`, and for `tblBook.BookID` onto `tblOrderItem.BookID`. Tick **Enforce Referential Integrity** each time.
7. Check that every line has **1** at the parent end and **∞** at the child end. Take **Screenshot 3** (the whole Relationships window).
8. To get **Screenshot 4**, double-click any join line to reopen the Edit Relationships dialog, capture it, then click **Cancel**. Press **Ctrl+S** to save the layout and close the Relationships window.



What your screen should look like: the Edit Relationships dialog. (Illustration only.)



What your screen should look like: the finished Relationships window. (Illustration only.)

Checkpoint Compare the Relationships window with the ERD in your design report. The four boxes and the one-to-many lines should match.

Why enforce referential integrity? It stops an order pointing at a customer who does not exist, and stops you deleting a customer who still has orders. You will test both in Task 5.

6. Task 4: Enter the data (10 minutes)

Double-click a table in the Navigation Pane to open it in Datasheet View. Enter the data **in this order**: `tblCustomer`, `tblBook`, `tblOrder`, `tblOrderItem`. Press **Tab** to move between cells. Access saves a row as soon as you move to the next one.

Table	Data to enter (one row per line)
tblCustomer	C1, Aisha Khan, aisha@mail.com C2, Tom Evans, tom@mail.com
tblBook	B1, Python Basics, 20 B2, Web Design, 15 B3, Databases, 25
tblOrder	O101, 1-Oct-2026, C1 O102, 3-Oct-2026, C2 O103, 5-Oct-2026, C1
tblOrderItem	O101, B1, 1 O101, B2, 2 O102, B1, 1 O103, B3, 1

Tip Type prices as plain numbers (20) and Access shows them as £20.00. Type dates as 1-Oct-2026. If you type 01/10/2026 on a computer with US settings, Access reads it as 10 January.

Checkpoint Why this order? A row in `tblOrder` cannot be saved until its customer exists, and a row in `tblOrderItem` cannot be saved until both its order and its book exist. Take **Screenshot 5** showing all four tables in Datasheet View (arrange the windows side by side, or capture them one at a time).

7. Task 5: Test referential integrity and validation (10 minutes)

A good database refuses bad data. Try each test below and record what happens in the table at the end of this task.

Test A: an order line for an order that does not exist

1. Open `tblOrderItem`. In the new row, type O999, B1, 1, then move to another row.
2. Access shows an error message. Take **Screenshot 8** with the message on screen, click **OK**, then press **Esc** to cancel the row.

Test B: delete a customer who has orders

1. Open `tblCustomer`. Click the row selector for **C1** and press **Delete**. Note what Access says, and make sure C1 is still there afterwards.

Test C: break the validation rule

1. In `tblOrderItem` type O103, B1, 0 in the new row. Note the message, then press **Esc**.

Test D: the insert problem

1. In `tblBook` add B4, Networking, 30. Nobody has ordered this book, yet Access saves it.

Test E: a duplicate composite key

1. In `tblOrderItem` type O101, B1, 3. Note the message, then press **Esc**.

Test	What I did	What Access did (write the message in your own words)
A	Added O999 to <code>tblOrderItem</code>	
B	Tried to delete customer C1	
C	Entered Quantity 0	
D	Added book B4 with no orders	
E	Entered O101 + B1 a second time	

Think Back in section 3.1 of the design report you listed three problems of a single big table: repeated data, the insert problem and the delete problem. Which of these tests shows the insert problem solved? Which shows the delete problem solved?

8. Task 6: Rebuild the order forms with a query (15 minutes)

Splitting data across four tables is only useful if you can put it back together. You will write a query that joins all four tables and compare the result with the 1NF table in the design report.

1. Click **Create**, then **Query Design**. In the Show Table box add `tblCustomer`, `tblOrder`, `tblOrderItem` and `tblBook`, then click **Close**. Join lines appear automatically because you created the relationships.
2. Double-click these fields to add them to the grid, in this order: `OrderID` and `OrderDate` (from `tblOrder`), `CustomerName` (`tblCustomer`), `Title` (`tblBook`), `Quantity` (`tblOrderItem`), `Price` (`tblBook`).
3. Click in the next empty **Field** cell and type: `LineTotal: [Quantity]*[Price]`
4. In the **Sort** row under `OrderID`, choose **Ascending**. Take **Screenshot 6** (the query in Design View).
5. Click **Run** (the red exclamation mark). Take **Screenshot 7** (the result).
6. Press **Ctrl+S** and save the query as `qryOrderLines`.

OrderID	OrderDate	CustomerName	Title	Quantity	Price	LineTotal	Click to Add
O101	01-Oct-2026	Aisha Khan	Python Basics	1	£20.00	£20.00	
O101	01-Oct-2026	Aisha Khan	Web Design	2	£15.00	£30.00	
O102	03-Oct-2026	Tom Evans	Python Basics	1	£20.00	£20.00	
O103	05-Oct-2026	Aisha Khan	Databases	1	£25.00	£25.00	

What your screen should look like: the result of `qryOrderLines`. (Illustration only.)

Checkpoint You should see **4 rows** and line totals of £20.00, £30.00, £20.00 and £25.00. These are the same four rows as the 1NF table in your design report. If you see 12 or more rows, a join line is missing: check the Relationships window.

Test F: the update problem

1. Open `tblCustomer` and change **Aisha Khan** to **Aisha K. Khan** in one row. Run `qryOrderLines` again.
2. Both of Aisha's order lines (O101 and O103) now show the new name, although you typed it only once. Change the name back.

9. Extension tasks (if you finish early)

- **Order totals.** Create a new query based on `qryOrderLines`. Click **Totals** (the Σ button), set `OrderID`, `OrderDate` and `CustomerName` to **Group By**, and `LineTotal` to **Sum**. Expected result: O101 £50.00, O102 £20.00, O103 £25.00. Save it as `qryOrderTotals`.
- **Price history.** Section 5 of the design report notes that changing a book's price would change past orders. Add a **PricePaid** (Currency) field to `tblOrderItem` and fill it in. Then change `qryOrderLines` to use `PricePaid` instead of `Price`.

- **Order entry form.** Click **Create**, then **Form Wizard**. Choose fields from `tblOrder` and `tblOrderItem`, and choose to view the data **by tblOrder** with a **subform**. This gives a form that looks like the paper order form.

10. Evidence checklist for your report

Take screenshots with **Windows key + Shift + S** (Snipping Tool), paste them into Word and crop to the Access window. Under each one, add a caption (“Figure 1: ...”) and **one or two sentences** saying what it shows and why it matters.

Figure	What to capture	What your explanation should mention
1	tblCustomer in Design View	Data types, field size, and which field is the primary key.
2	tblOrderItem in Design View	Why the primary key has two fields. The validation rule on Quantity.
3	The Relationships window	The 1 and ∞ ends, and how it matches your ERD.
4	The Edit Relationships dialog	What Enforce Referential Integrity does, and why cascade is off.
5	All four tables in Datasheet View	Why the data was entered in this order. Each fact is stored once.
6	qryOrderLines in Design View	Which tables are joined, and what LineTotal calculates.
7	The result of qryOrderLines	How it matches the 1NF table, so no information was lost.
8	The referential integrity error message	What you tried, what Access refused, and why that protects the data.

Your finished report should also include a short **test table** (Tests A to F) and a brief **evaluation**: what the design does well, and at least two limitations (for example price history, or typing IDs by hand).

11. Troubleshooting

What you see	Likely cause	What to do
Enforce Referential Integrity is greyed out, or the message says no unique index was found.	The “one” side field is not a primary key.	Open the parent table in Design View, set the primary key, save, and try again.
“Data types or sizes of the fields do not match.”	The foreign key differs from the primary key.	Make both Short Text, size 10.
Cannot create the relationship because a table is in use.	A table is still open.	Close all tables and queries.
“Index or primary key cannot contain a Null value.”	A key field was left blank.	Fill in the key, or press Esc to cancel the row.
A relationship fails because some rows have no matching parent.	A typo in a foreign key, such as C11 instead of C1.	Correct the typo and try again.
The query returns far too many rows.	A join line is missing.	Check the Relationships window. In the query, drag the key field from one table to the other.
Dates show the wrong day or month.	Regional settings differ.	Type dates as 1-Oct-2026.

Tutor Notes (remove this page before issuing to students)

Suggested timing

Task	Minutes	What to check as you circulate
1. Create the database	5	File saved on a drive that students can reach next week.
2. Create the four tables	25	Foreign key type and size match the primary key. Composite key on tblOrderItem.
3. Relationships	15	Dragging from parent to child. Enforce Referential Integrity ticked.
4. Enter data	10	Parents entered before children.
5. Tests	10	Students can explain each error message.
6. Query	15	Four rows, not twelve.
Evidence and wrap-up	10	Eight screenshots with captions.

Common misconceptions

- Naming the table Order, then meeting syntax errors in queries later.
- Making OrderID the only primary key of tblOrderItem. Access then refuses a second book on the same order.
- Using AutoNumber for the foreign key. Access creates a Long Integer, which does not match a Short Text key.
- Dragging from child to parent. Access may still create the line but with the keys the wrong way round.
- Ticking Cascade Delete without thinking. Ask “do you want deleting a customer to delete their orders?”.

Note on the screenshots The Access screens in this guide and in the sample report are **illustrative recreations**, not captures from a live Access session. Menu names and ribbon layout differ between Access versions. If you want exact appearance for your lab version, replace them with your own captures before issuing.