



PRACTICAL FILE 1 TO 20 QUESTIONS



NAME: KARTIK RAWAT
School: Geeta Public School
Class: X (CBSE)
Section: 10th C
Roll Number: 73

Guided By: Sukhwinder Singh Sohal Sir



Q1 Consolidate data of two sheets given below in third sheet of Libre Office Calc.

Procedure

- Step 1: Open LibreOffice Calc.**
- Step 2: In Sheet1, create a table with headings Name and June Pocket Money and enter the data.**
- Step 3: Insert a new sheet (Sheet2) and create a table with headings Name and July Pocket Money, then enter the data.**
- Step 4: Insert another new sheet and rename it as Sheet3 (for consolidated result).**
- Step 5: Click on Data → Consolidate.**
- Step 6: Select Function = Sum.**
- Add the ranges Sheet1.\$A\$1:\$B\$6 and Sheet2.\$A\$1:\$B\$6.**
- Step 7: Tick Row labels under Options and set Copy results to: Sheet3.\$A\$1.**
- Step 8: Click OK to consolidate the data.**

	A	B	C
1	Name	June Pocket Money	
2	Amit	1200	
3	Sumit	1000	
4	Naina	1500	
5	Suman	1300	
6	Kamal	1400	

	A	B	C
1	Name	July Pocket Money	
2	Amit	900	
3	Sumit	1100	
4	Naina	1300	
5	Suman	1200	
6	Kamal	1250	

Consolidate dialog box details:

- Function: Sum
- Consolidation ranges: \$Sheet1.\$A\$1:\$B\$6, \$Sheet2.\$A\$1:\$B\$6
- Source data ranges: (empty)
- Copy results to: \$Sheet3.\$A\$1
- Options: Row labels, Column labels, Link to source data

Q2 Consider the Given Table Student mentioning Students details and perform the following:

Procedure



Step 1: Open LibreOffice Calc.

Step 2: In Sheet1, create a table with headings Rollno, Class, Name, Phone_no and enter the required data.

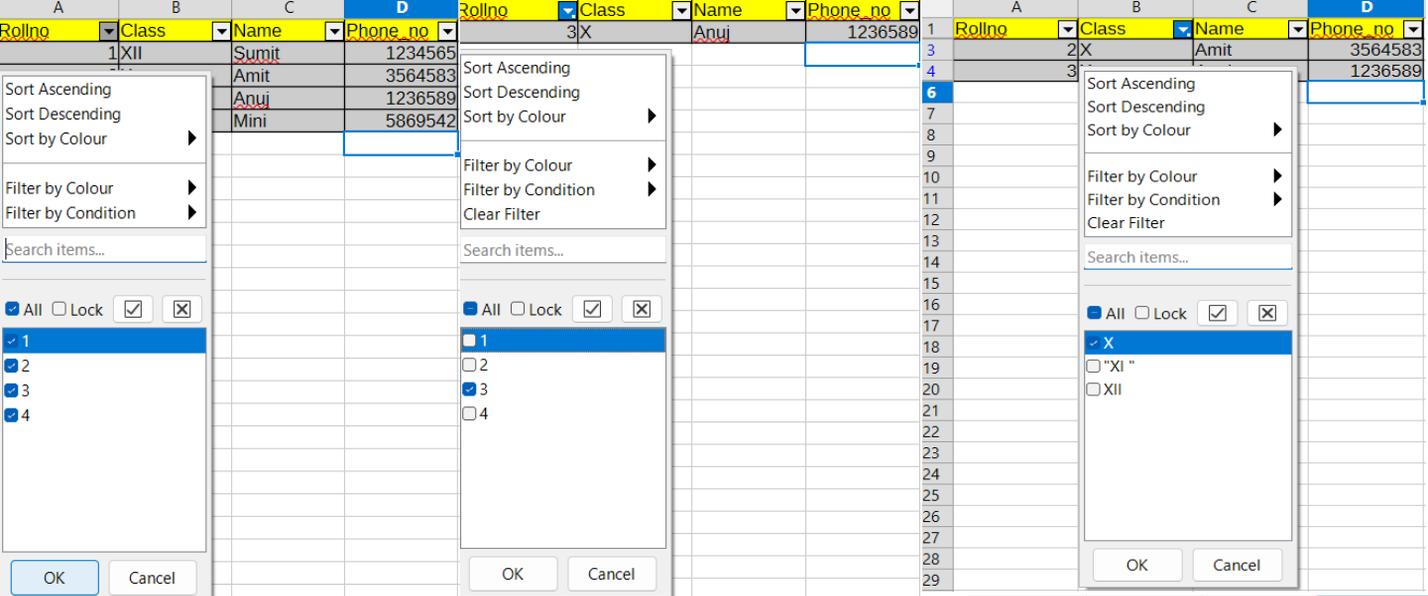
Step 3: Select the header row and apply a background color. Then select the data cells and apply another background color.

Step 4: Select the complete table, click on Borders, and apply Outer Border and All Inner Lines.

Step 5: Select the entire table, go to Data → AutoFilter.

Rollno	Class	Name	Phone_no
1	XII	Sumit	1234565
2	X	Amit	3564583
3	X	Anuj	1236589
4	XI	Mini	5869542

Rollno	Class	Name
1	XII	Sumit
2	X	Amit
3	X	Anuj
4	XI	Mini



Now Apply Different Filters

Filter 1: Display all records of the table
Click on any filter arrow in the heading row.
Select All.

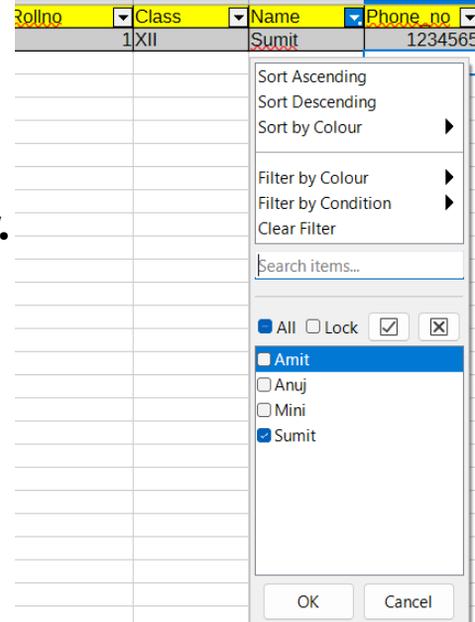
All records of the table will be displayed.

Filter 2: Display record of student whose Roll Number is 3
Click on the filter arrow of Rollno column.

Select 3 from the list.
Only the record of the student with roll number 3 will be displayed.

Filter 3: Display records of students who are in Class X
Click on the filter arrow of Class column.
Select X from the list.
Records of students studying in Class X will be displayed.

Filter 4: Display record of student named Sumit
Click on the filter arrow of Name column.
Select Sumit from the list.
The record of Sumit will be displayed.

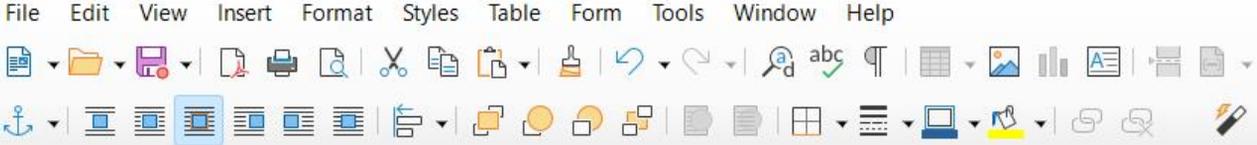


Q3 You are working on a project report about the solar system for your science class.

- Insert an image of the solar system, a constellation, and a spacecraft.
- Resize the solar system image to fit half of the page width.
- Crop the constellation image to remove unnecessary background.
- Align the solar system image to the centre of the page.
- Place the spacecraft image in the top-right corner of the page with text wrapping around it.

Q3.odt — LibreOffice Writer

File Edit View Insert Format Styles Table Form Tools Window Help

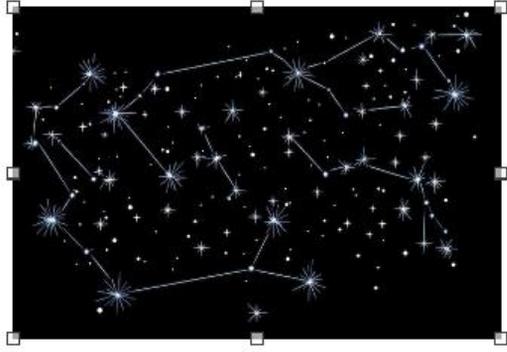
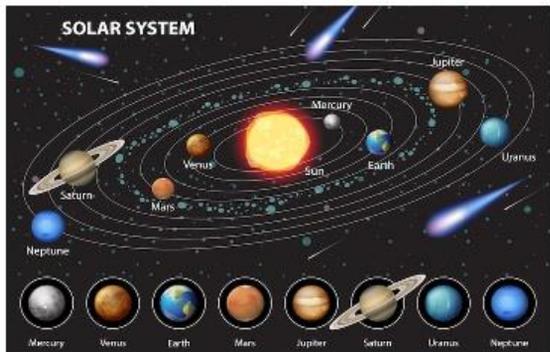


1 5 | 4 | 3 | 2 | 1 | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |

Project Report on Solar System

The solar system consists of the Sun, eight planets, their satellites, asteroids and other celestial bodies. It helps us understand the structure of space and the movement of planets.

In this project, images of the solar system, a constellation and a spacecraft are inserted to explain the topic visually. Various formatting tools such as resize, crop, alignment and text wrapping are used to present the information clearly.



Procedure (Step by Step)

Step 1: Open **LibreOffice Writer** from the Start Menu.

Step 2: Type the heading **Project Report on Solar System**, make it **Bold** and **Centre aligned**.

Step 3: Type a short report describing the solar system below the heading.

Step 4: Click on **Insert → Image**, insert the **solar system image**, resize it to **half page width** and **centre align** it.

Step 5: Click on **Insert → Image**, insert the **constellation image**, use the **Crop** tool to remove unnecessary background and resize it.

Step 6: Click on **Insert → Image**, insert the **spacecraft image**, place it in the **top-right corner** and apply **text wrapping**.

Step 7: Adjust all images properly so that the content fits on **one page**.

Step 8: Click on **File → Save As**, give a file name and save the document.

Result

A project report on the solar system is created using images and formatting tools in LibreOffice Writer.

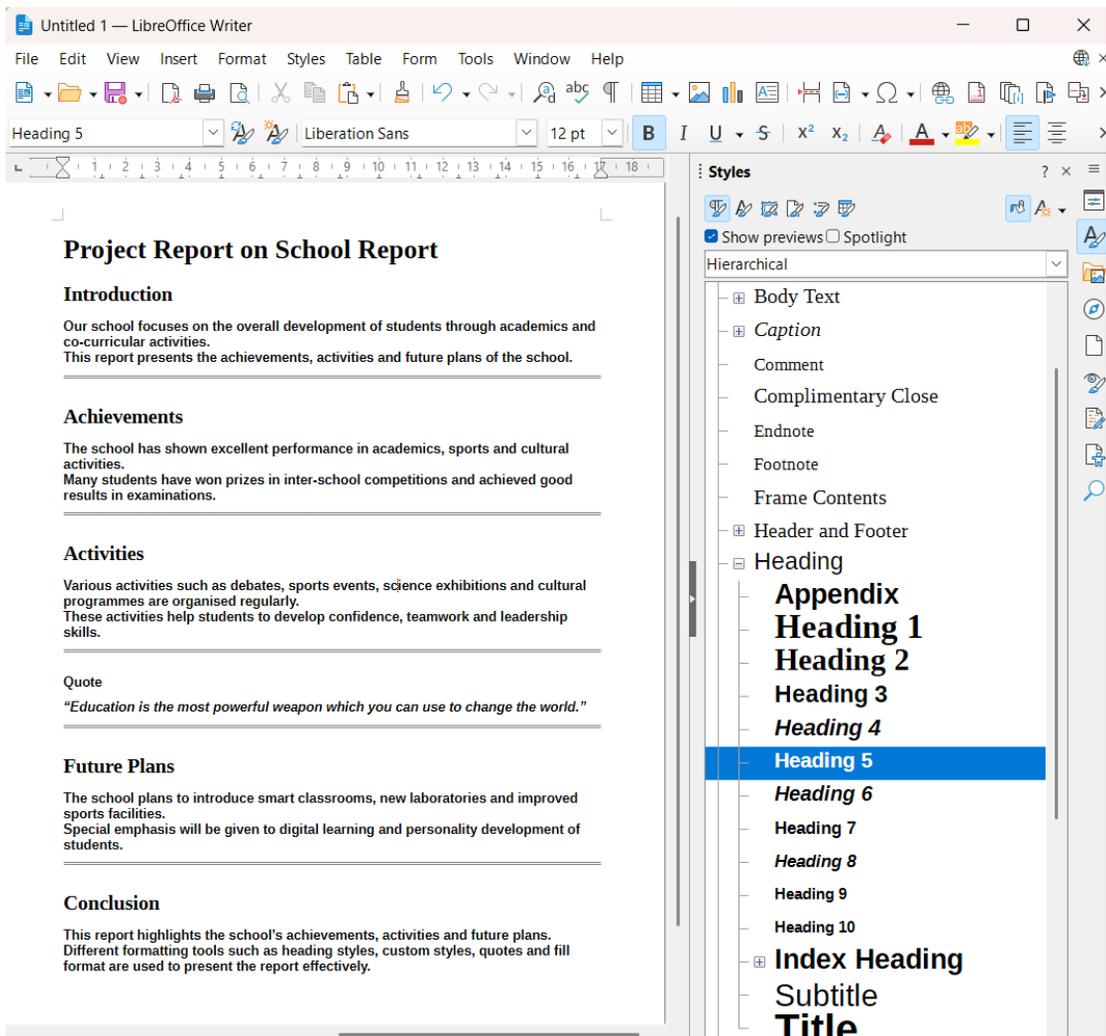
Conclusion

LibreOffice Writer provides easy options to insert, resize, crop, align images and apply text wrapping.



Q4 Mr. Mohit, school teacher, assigned a task to Jatin to create a school report containing sections such as Introduction, Achievements, Activities and Future Plans.

- Apply **Heading styles** for main section titles.
- Apply a **custom style to quotes** within the report.
- Use **Fill Format** to apply a text style to all paragraphs.



- Step 1:**
LibreOffice Writer open karo aur **File → New → Text Document** par click karo.
- Step 2:**
Report ka Title likho "*Project Report on School Report*" aur use select karke **Styles → Title** apply karo.
- Step 3:**
Main sections likho: **Introduction, Achievements, Activities, Future Plans, Conclusion.**



Step 4:

Har section heading select karke **Styles** → **Heading** (Heading 2 / Heading 5) apply karo.

Step 5:

Har heading ke niche paragraph likho aur text select karke **Styles** → **Body Text** apply karo.

Step 6:**Quote likho:**

“Education is the most powerful weapon which you can use to change the world.”

Step 7:

Quote select karke **Bold + Italic** apply karo aur **Styles** → **New Style from Selection** se **Custom Quote Style** banao.

Step 8:

Styles panel me **Fill Format Mode** (paint bucket icon) par click karo aur same style sabhi required paragraphs par apply karo.

Step 9:

Fill Format Mode band karne ke liye **Esc** key dabao.

Step 10:

Document ko **File** → **Save As** se save karo.

Q5 Manish is designing the cover page of your school magazine.

- **Insert an image of the school building at the centre of the cover page.**
- **Crop and resize the image proportionately.**
- **Add a drop shadow to the image for emphasis.**

Q5 – Steps

Step 1: Open **LibreOffice Draw**.

Step 2: Create a new drawing and type the title of the school magazine at the top of the page.

Step 3: Click on **Insert** → **Image** and insert the image of the **school building**.



Step 4: Select the image and place it at the **centre of the page**.

Step 5: Resize the image **proportionately** using the **corner handles**.

Step 6: Right-click on the image, select **Crop**, and remove unnecessary parts.

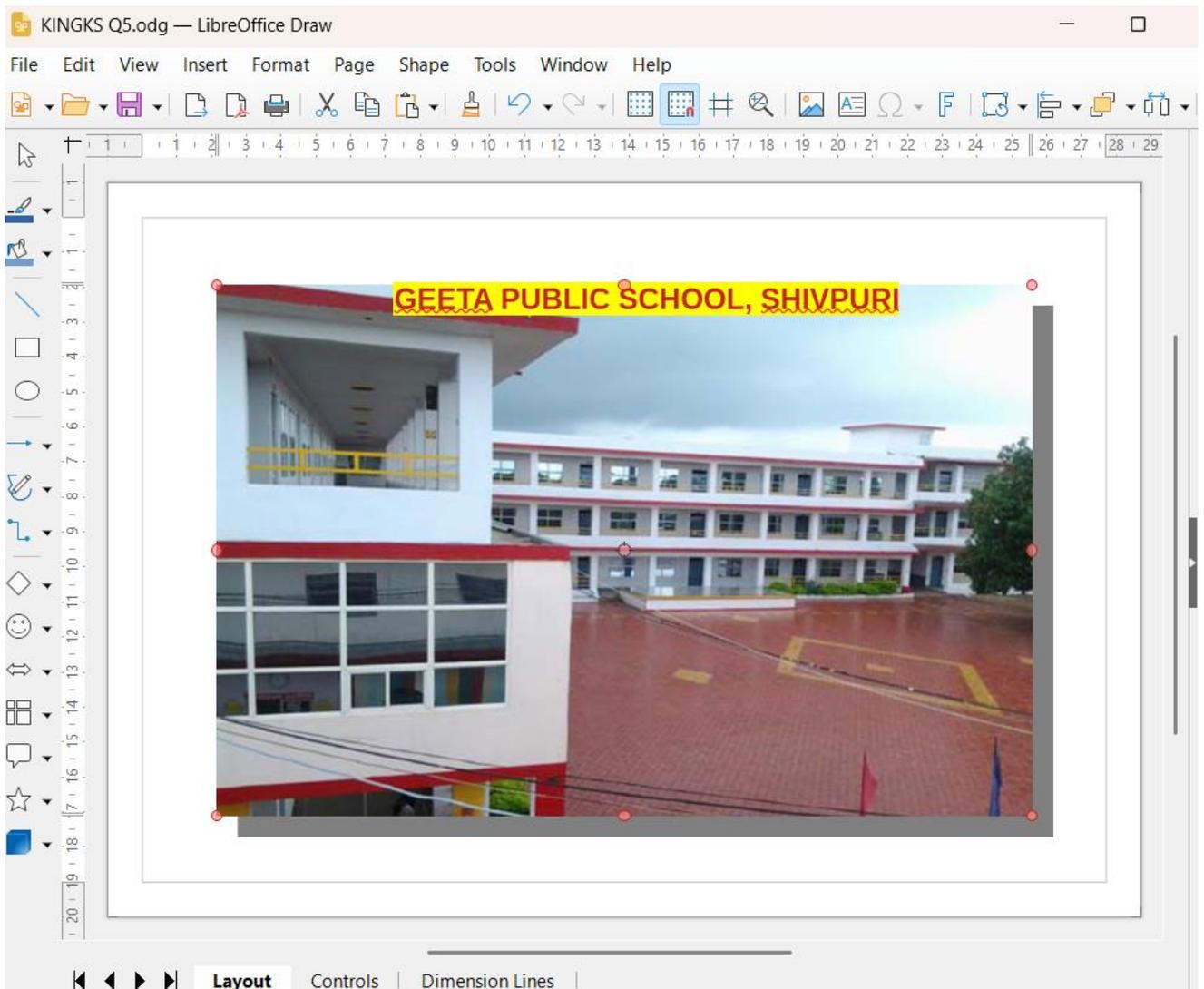
Step 7: Select the image, open **Properties**, and enable the **Drop Shadow** option.

Step 8: Adjust the image position and size to make the cover page attractive.

Step 9: Click on **File → Save As** and save the drawing.

Result

The cover page of the school magazine is created with a centred image, proportional resizing, cropping and drop shadow effect.



Q6 You are compiling the final report for your school, which includes multiple chapters.

- **Create a Table of Contents (TOC) that reflects the report with main topics and subtopics.**
- **Apply character styles to the TOC to differentiate between main topics (**bold**) and subtopics (*italic*).**
- **Update the TOC as new sections are added to the report.**

The screenshot shows the LibreOffice Writer interface with a document titled "KINGKS Q6.odt". The document content is as follows:

Q6 – Final School Report

Introduction

This report presents an overview of the school activities and achievements. It is organised into different chapters and sub-chapters for easy understanding.

Achievements

The school has achieved excellent results in academics, sports and cultural activities.

Q6 – Final School Report.....	1
Introduction.....	1
Achievements.....	1
Academic Achievements.....	1
Sports Achievements.....	1
Activities.....	1
Future Plans.....	1
Table of Contents.....	2
Conclusion.....	2

Academic Achievements

Students have performed well in examinations and academic competitions.

Sports Achievements

Students have won prizes in various inter-school sports competitions.

Activities

Various academic, cultural and sports activities are organised regularly in the school.

Future Plans

The school plans to introduce smart classrooms, advanced laboratories and improved sports facilities.
More emphasis will be given to digital learning.

Page 1 of 2 | 194 words, 1,302 characters | Default Page Style

Q6 – Steps (LibreOffice Writer)

Step 1: Open **LibreOffice Writer** and open the school report document.

Step 2: Apply **Heading styles** to the report:

- Main title → **Heading 1**
- Main sections → **Heading 2**
- Sub-sections → **Heading 3**

Step 3: Place the cursor at the position where the **Table of Contents (TOC)** is required.

Step 4: Click on **Insert → Table of Contents and Index → Table of Contents** and click **OK**.

Step 5: Select the **main topics** in the TOC and apply **Bold** character style.

Step 6: Select the **sub-topics** in the TOC and apply *Italic* character style.

Step 7: Add a new section in the report using the appropriate **Heading style**.

Step 8: Right-click on the TOC and select **Update Index** to update the TOC.

Result

The Table of Contents is created and updated with main topics in **bold** and sub-topics in *italic*.



Q7 Devash creates a template for the monthly school newsletter.

- **Design a template with predefined sections for the Principal's message, events, achievements and student spotlights.**
- **Include a background colour and custom fonts.**
- **Save the template and use it to format the current month's newsletter.**

Step 1: Open **LibreOffice Writer** and create a new document.

Step 2: Type the title **Geeta Public School – Monthly Newsletter** and apply **Heading 1**.

Step 3: Create fixed sections **Principal's Message, Events, Achievements and Student Spotlight** and apply **Heading 2** to all section headings.

Step 4: Apply a **background colour** using **Format → Page Style → Area**.

Step 5: Change the **font style and size** of the title, headings and text to use **custom fonts**.

Step 6: Adjust spacing and alignment to make the layout uniform.

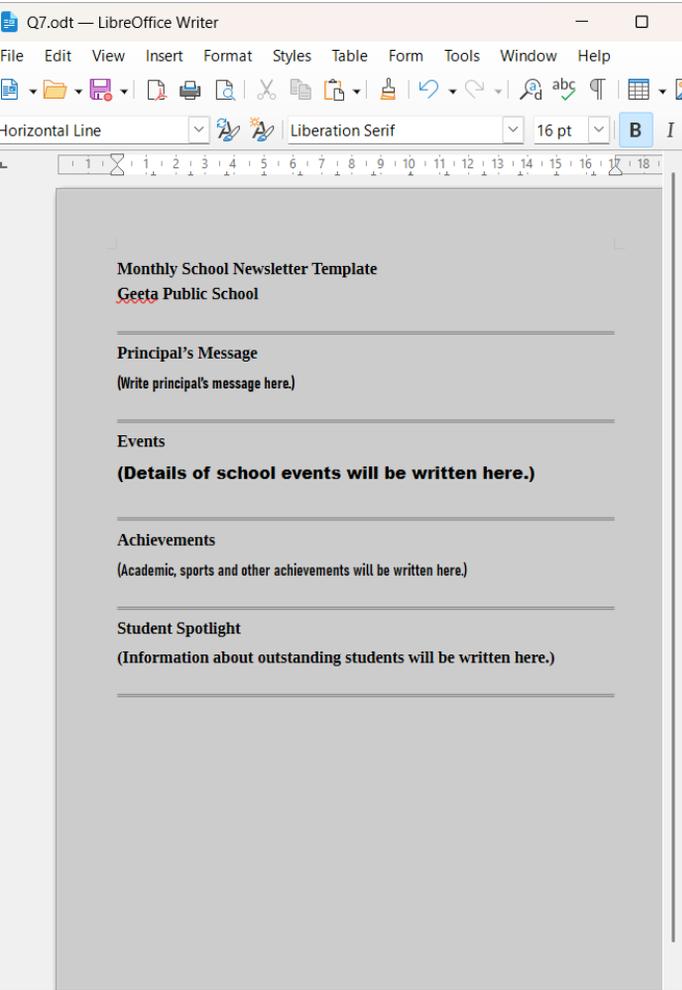
Step 7: Save the document as a template using **File → Templates → Save as Template** and name it **School Newsletter Template**.

Step 8: Create a new file using the saved template by clicking

File → New → Templates and selecting the saved template.

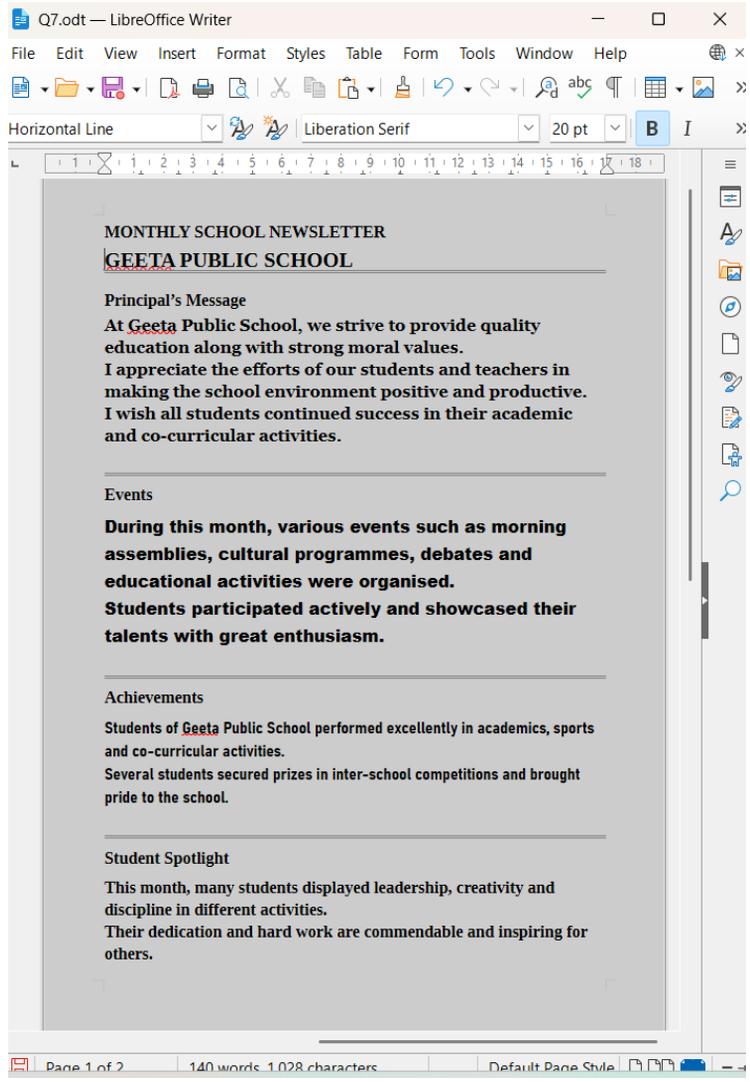
Step 9: Enter the current month's newsletter content in the predefined sections.

Step 10: Save the newsletter file.



<<<<< **TEMPLATE**
Q7

USED TEMPLATE
Q7 >>>>>



Q8 Create the following sheet and calculate the profit based on different units sold.

B4		\sum	=	=(B2-B1)*B3	
	A	B	C	D	E
1	Cost Price	200		Unit Sold	Profit
2	Units Sold	280		200	
3	Sales Price	500		300	
4	Profit	40000		400	
5				500	
6				600	
7				700	
8				800	
9					
10					
11					

Step 1: Open **LibreOffice Calc** and create a new spreadsheet.

Step 2: Enter the headings as shown below:

- **A1:** Cost Price
- **A2:** Selling Price
- **A3:** Unit Sold
- **A4:** Profit

Step 3: Enter the given values:

- **B1:** 200
- **B2:** 280
- **B3:** 500

Step 4: Click on cell **B4** and type the formula to calculate profit:

$$=(B2-B1)*B3$$

Press **Enter**.

(Profit will be displayed, e.g. **40000**)

Step 5: Now create the second table for different units sold.

Enter headings:

- **D1:** Unit Sold
- **E1:** Profit

Step 6: Enter unit sold values in column D:

- D2 → 200
- D3 → 300
- D4 → 400
- D5 → 500
- D6 → 600
- D7 → 700
- D8 → 800

Step 7: Click on cell E2 and enter the formula:

$=(B2-B1)*D2$

Press **Enter**.

Step 8: Use the **Fill Handle** to copy the formula from **E2 to E8** to calculate profit for all units sold.

Step 9: Adjust column width and save the file.

Result

Profit is calculated successfully for different units sold using formulas in LibreOffice Calc.

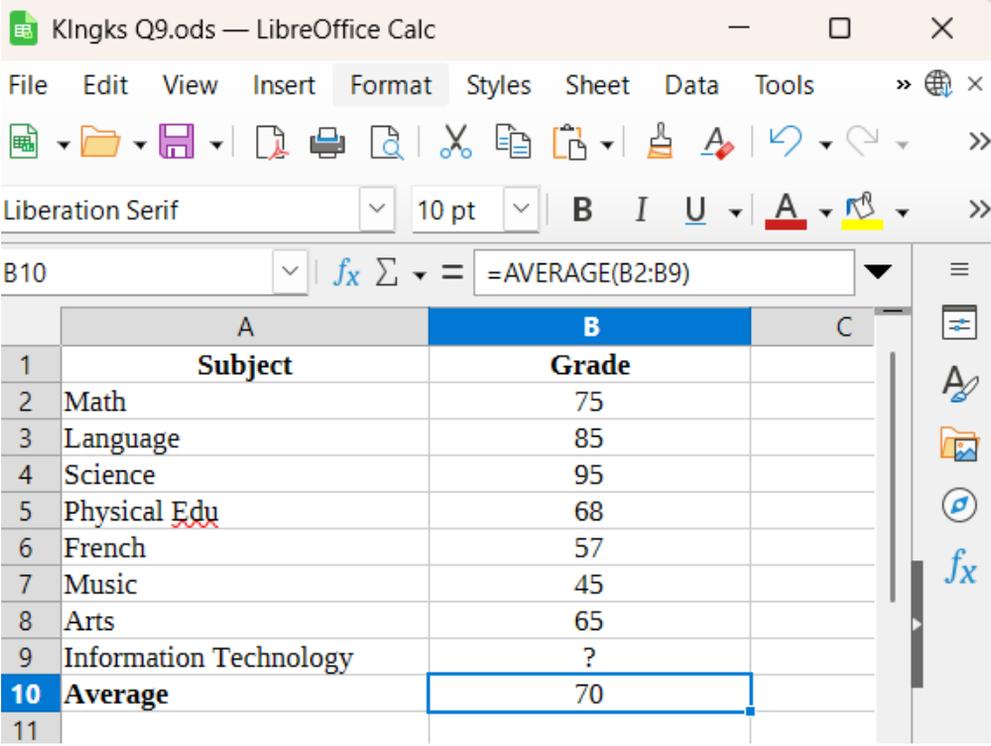
Conclusion

LibreOffice Calc helps in calculating profit easily by using formulas and fill handle feature.

	A	B	C	D	E	F
1	Cost Price	200		Unit Sold	Profit	
2	Units Sold	280		200	16000	
3	Sales Price	500		300	24000	
4	Profit	$=(B2-B1)*B3$		400	32000	
5				500	40000	
6				600	48000	
7				700	56000	
8				800	64000	
9						
10						
11						
12						

$$\text{Profit} = (\text{Selling Price} - \text{Cost Price}) \times \text{Units Sold}$$

Q9 Consider the following marksheet and use the Goal Seek feature to find the marks in place of question mark (?).



The screenshot shows the LibreOffice Calc interface. The spreadsheet has the following data:

	A	B	C
1	Subject	Grade	
2	Math	75	
3	Language	85	
4	Science	95	
5	Physical Edu	68	
6	French	57	
7	Music	45	
8	Arts	65	
9	Information Technology	?	
10	Average	70	
11			

The formula bar shows the formula: `=AVERAGE(B2:B9)`

Q9 – Steps (LibreOffice Calc – Goal Seek)

Step 1: Open **LibreOffice Calc** and create a new spreadsheet.

Step 2: Create the marksheet table by entering **Subject** in column A and **Grade** in column B.

Step 3: Enter the given marks for all subjects and **leave the cell with question mark (?) blank.**

Step 4: In the **Average** cell, enter the formula to calculate average, for example:

`=AVERAGE (B2 : B9)`

Step 5: Select the cell containing the **Average** value.

Step 6: Click on **Tools** → **Goal Seek.**

Step 7: In the Goal Seek dialog box, fill the following:

- **Formula cell:** Average cell
- **Target value:** 70
- **Variable cell:** Cell containing the missing marks (?)

Step 8: Click **OK** and then click **Yes** to accept the result.

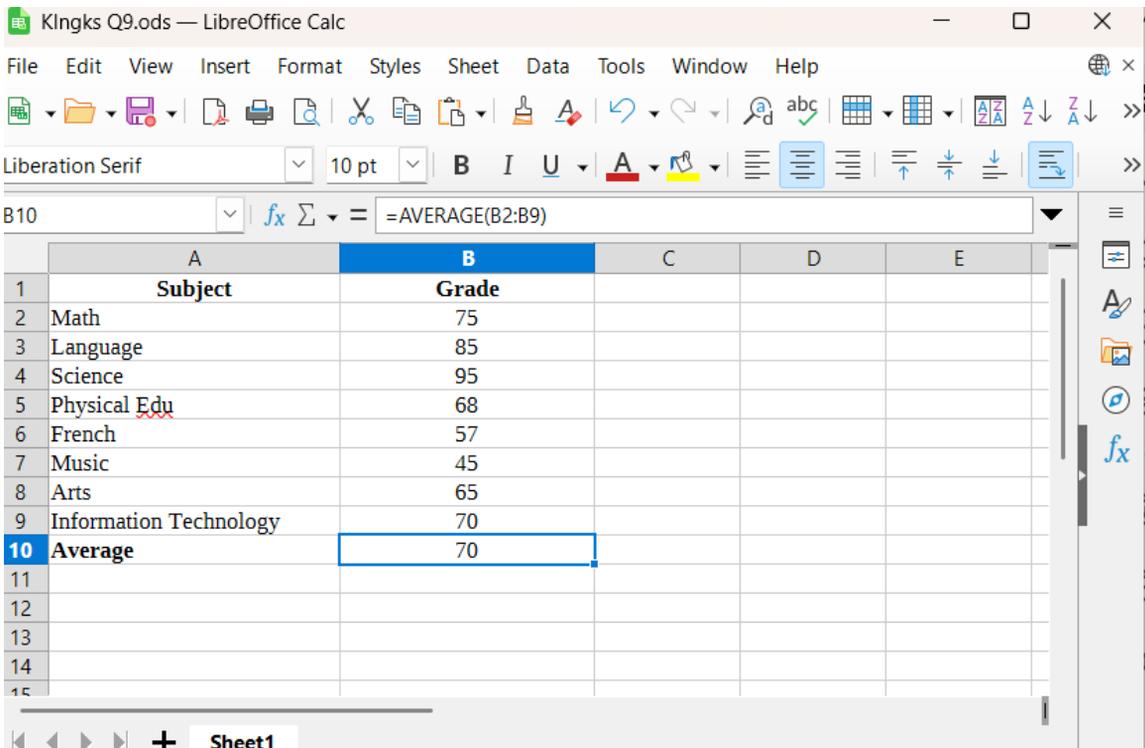
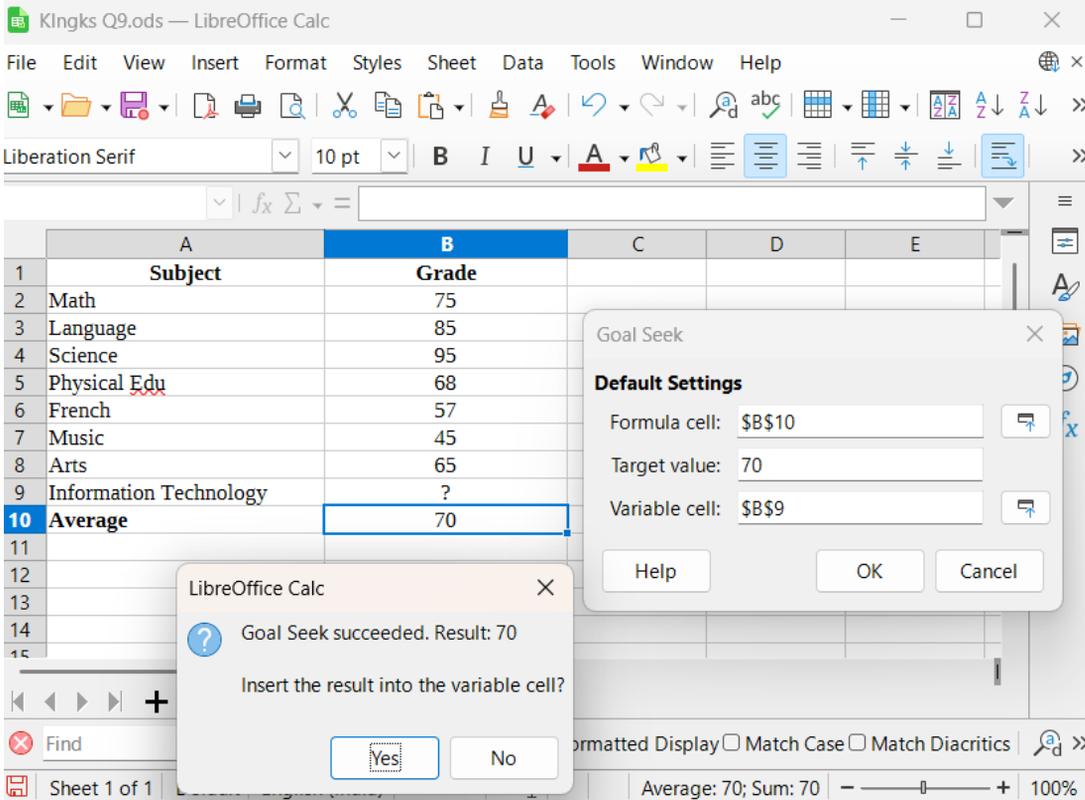
Step 9: The required marks in place of **(?)** are displayed automatically.

Result

The missing marks are calculated successfully using the Goal Seek feature.

Conclusion

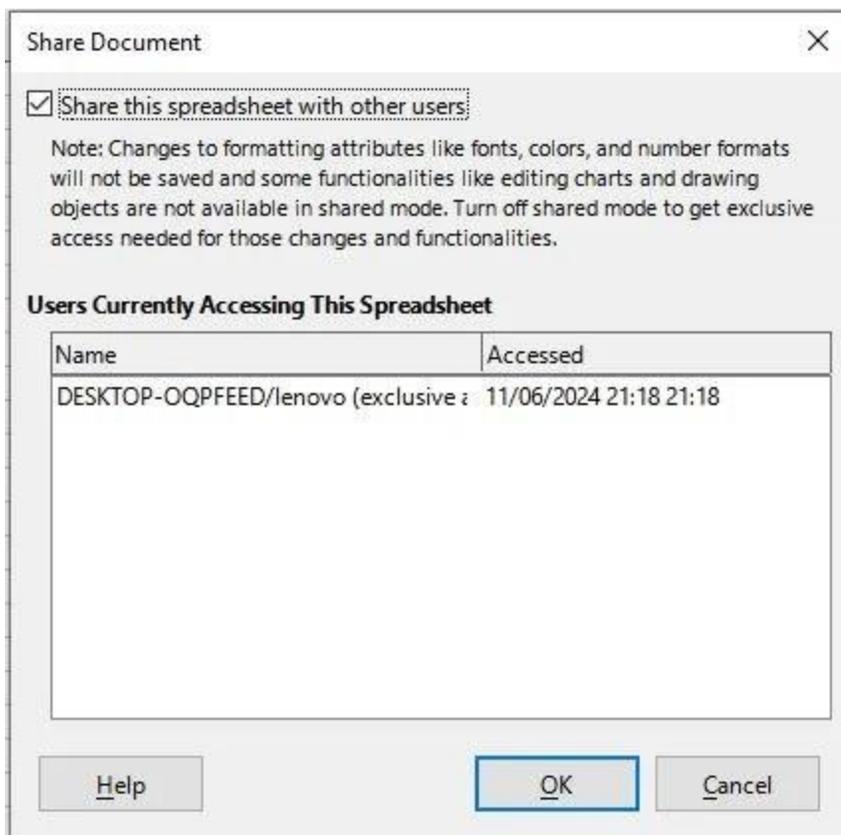
Goal Seek in LibreOffice Calc is used to find an unknown value when the desired result is known.



Q10 You and your classmates are working on a group project using a shared spreadsheet.

Perform the following tasks:

- **Set up a shared spreadsheet for collaboration among classmates.**
- **Practice entering, editing and formatting comments to enhance communication and clarity within the spreadsheet.**
- **Learn to manage changes effectively by reviewing and either accepting or rejecting them.**



Step 1: Open **LibreOffice Calc** and create a new spreadsheet.

Step 2: Save the file so that it can be shared with classmates.

Step 3: Enable sharing by clicking **Tools** → **Share Spreadsheet** and click **OK**.

Step 4: Enter data in the spreadsheet collaboratively with classmates.

Step 5: To add a comment, right-click on a cell and select **Insert Comment**.

Step 6: Type the comment and format it if required for clarity.

Step 7: Enable change tracking by clicking **Edit → TrackChanges → Record**.

Step 8: Make changes in the spreadsheet (edit data).

Step 9: Review changes by clicking **Edit → Track Changes → Manage**.

Step 10: Accept or reject changes as required and save the spreadsheet.

Result

A shared spreadsheet is created where comments are added and changes are tracked, reviewed and managed successfully.

Conclusion

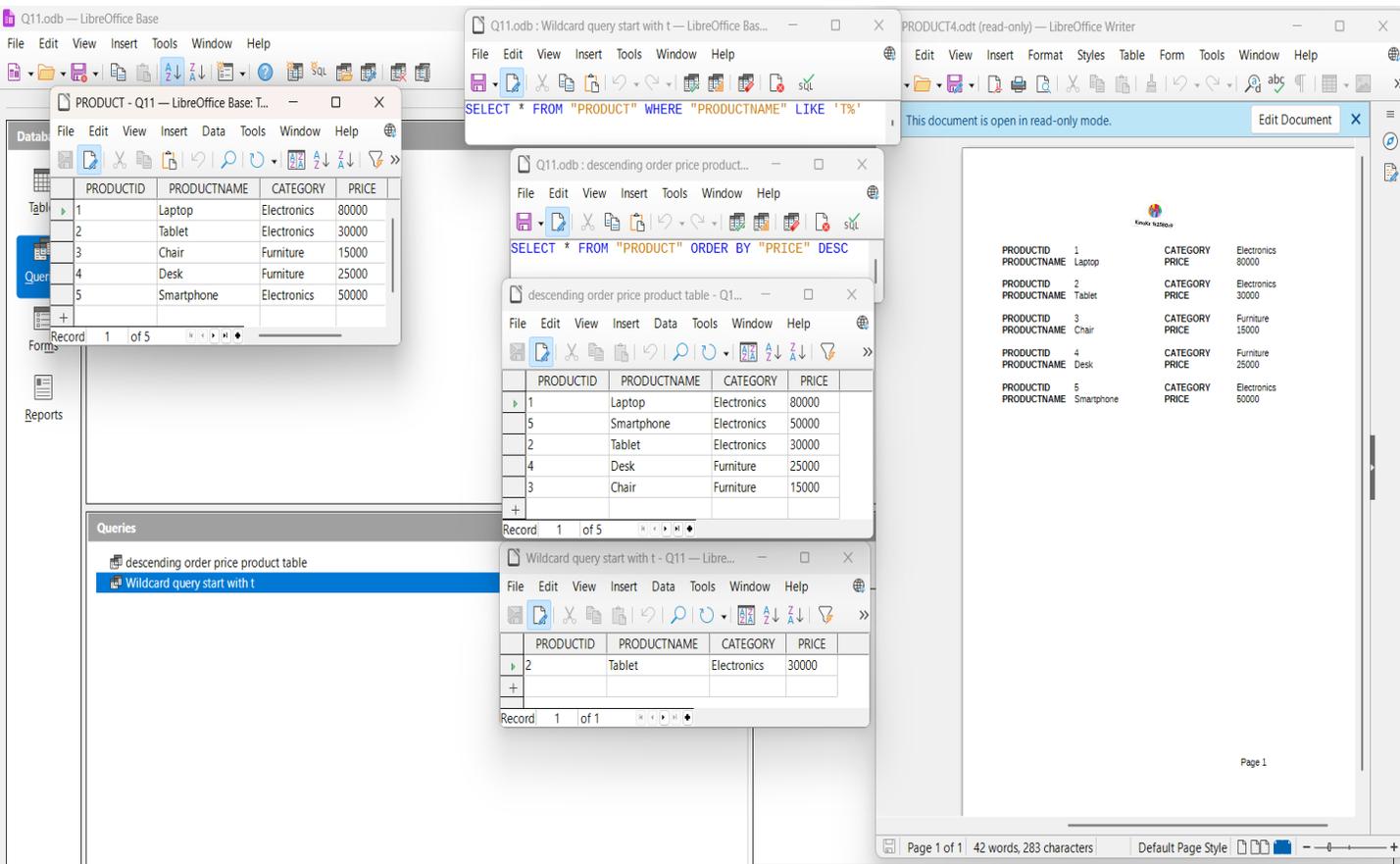
LibreOffice Calc provides tools to collaborate using shared spreadsheets, comments and track changes for effective group work.

In this task, a shared spreadsheet is created using **LibreOffice Calc**, which is a spreadsheet application based on **open-source software**. LibreOffice Calc works on a **cell-based spreadsheet system**, where data is organised in rows and columns. The spreadsheet is shared to allow multiple classmates to collaborate on the same file. Comments are added to cells to improve communication and clarity while working on the project. The **Track Changes** feature of LibreOffice Calc is used to record all edits made by different users. These changes are reviewed and either accepted or rejected as required. This ensures proper collaboration, transparency, and effective management of group work.



Q11 Consider the following Product table and perform these tasks:

- Sort the Product table by Price in descending order.
- Use a wildcard to create a query that retrieves all products where the ProductName starts with letter 'T'.
- Generate a report based on the Product table.
- Format the report to show products in ascending order of ProductID.
- Add the page number in the footer of the report.



Software: LibreOffice Base

Base: Open-source database management system

A. Create Database

Step 1: Open **LibreOffice Base**.

Step 2: Select **Create a new database** and click **Next**.

Step 3: Choose **Yes**, register the database and click **Finish**.

Step 4: Save the database with a suitable name (e.g. **Q11.odb**).

B. Create Table using Design View

Step 5: Click on **Tables** → **Create Table in Design View**.

Step 6: Create the following fields:

Field Name	Data Type
PRODUCTID	Integer
PRODUCTNAME	Text (VARCHAR)
CATEGORY	Text (VARCHAR)
PRICE	Integer

Step 7: Set **PRODUCTID** as the **Primary Key**.

Step 8: Save the table as **PRODUCT**.

C. Enter Records

Step 9: Open the **PRODUCT** table in **Data View**.

Step 10: Enter the given product records and save the table.

D. Query (Sort by Price – Descending) – Design View

Step 11: Click **Queries** → **Create Query in Design View**.

Step 12: Add the **PRODUCT** table and click **Close**.

Step 13: Drag all fields to the query grid.

Step 14: In the **Sort** row of **PRICE**, select **Descending**.

Step 15: Click **Run Query** and save it as **descending order price product table**.

E. Query (Wildcard – ProductName starts with T)

Step 16: Create a new query in **Design View**.

Step 17: Add the **PRODUCT** table.

Step 18: Drag required fields to the grid.

Step 19: In the **Criteria** row under **PRODUCTNAME**, type:
LIKE 'T%'

Step 20: Run and save the query as **Wildcard query start with T.**

F. Create Report using Wizard

Step 21: Click **Reports** → **Use Wizard to Create Report.**

Step 22: Select **PRODUCT** table and move all fields to the report.

Step 23: Click **Next** and select **PRODUCTID** for sorting in **Ascending order.**

Step 24: Choose a report layout and click **Next.**

Step 25: Give a suitable report name and click **Finish.**

G. Add Page Number in Footer

Step 26: Open the report in **Edit Mode.**

Step 27: Click **Insert** → **Page Number.**

Step 28: Place the page number in the **Footer** section.

Step 29: Save and close the report.

Result

The PRODUCT table is created, sorted, queried using wildcard, and a report is generated with ProductID in ascending order and page number in the footer.

Conclusion

LibreOffice Base helps in creating tables, queries and reports efficiently using Design View and Wizard tools.



Q12 You are tasked with organising student marks data in a spreadsheet and automating the calculation of total marks using a macro named “TotalMarks”.

- **Set up a spreadsheet with three columns: Student Name in column A, Subject in column B, and Marks Obtained in column C. Ensure data starts from row 2.**
- **Develop a macro named “TotalMarks” that iterates through each row of the spreadsheet (starting from row 2) to calculate the total marks obtained by each student.**
- **For each student, sum up the marks obtained from all subjects listed.**
- **Display the total marks for each student in column D, starting from row 2.**
- **Execute the macro to automatically calculate and populate the total marks for each student based on the data provided in the spreadsheet.**

The screenshot shows two windows from LibreOffice. The left window is 'Untitled 12.ods - LibreOffice Calc' displaying a spreadsheet with the following data:

	A	B	C	D	E
1	Student Name	Subject	Marks Obtained	Total Marks	
2	Ram	Maths	65	65	
3	Ram	Science	72	137	
4	Ram	English	58	195	
5	Ram	Grammar	80	275	
6	Ram	Social Science	75	350	
7	Ram	IT	68	418	
8	Ram	AI	70	488	
9	Ram	Lab Manual	66	554	
10	Ram	Internal	74	628	

The right window is 'My Macros & Dialogs.Standard - LibreOffice Basic' showing the following VBA code for the 'TotalMarks' macro:

```
1 Sub TotalMarks
2   Dim oSheet As Object
3   Dim i As Integer
4   Dim total As Integer
5
6   oSheet = ThisComponent.CurrentController.ActiveSheet
7   total = 0
8
9   For i = 1 To 9 ' Row 2 to Row 10 (0-based index)
10      total = total + oSheet.getCellByPosition(2, i).Value
11      oSheet.getCellByPosition(3, i).Value = total
12   Next i
13 End Sub
```

Software: LibreOffice Calc

Step 1: Open **LibreOffice Calc** and create a new spreadsheet.

Step 2: In Row 1, enter the headings:

- **A1:** Student Name
- **B1:** Subject
- **C1:** Marks Obtained
- **D1:** Total Marks

Step 4: Click on **Tools** → **Macros** → **Organize Macros** →

LibreOffice Basic.

Step 5: Create a new module and write a macro named

TotalMarks.

Step 6: Program the macro to read marks from **Column C** and calculate the cumulative total of all subjects.

Step 7: Display the calculated total marks in **Column D** for each row.

Step 8: Save and close the Macro editor.

Step 9: Run the macro using **Tools** → **Macros** → **Run Macro** and select **TotalMarks.**

Result

The total marks of the **single student** across all subjects are calculated automatically.

Conclusion

The macro TotalMarks is used to calculate the total marks of one student studying multiple subjects.

Q13 You have received two versions of a financial report from different teams. You need to identify discrepancies and reconcile the differences between the two sheets. How would you do this task in LibreOffice Calc?

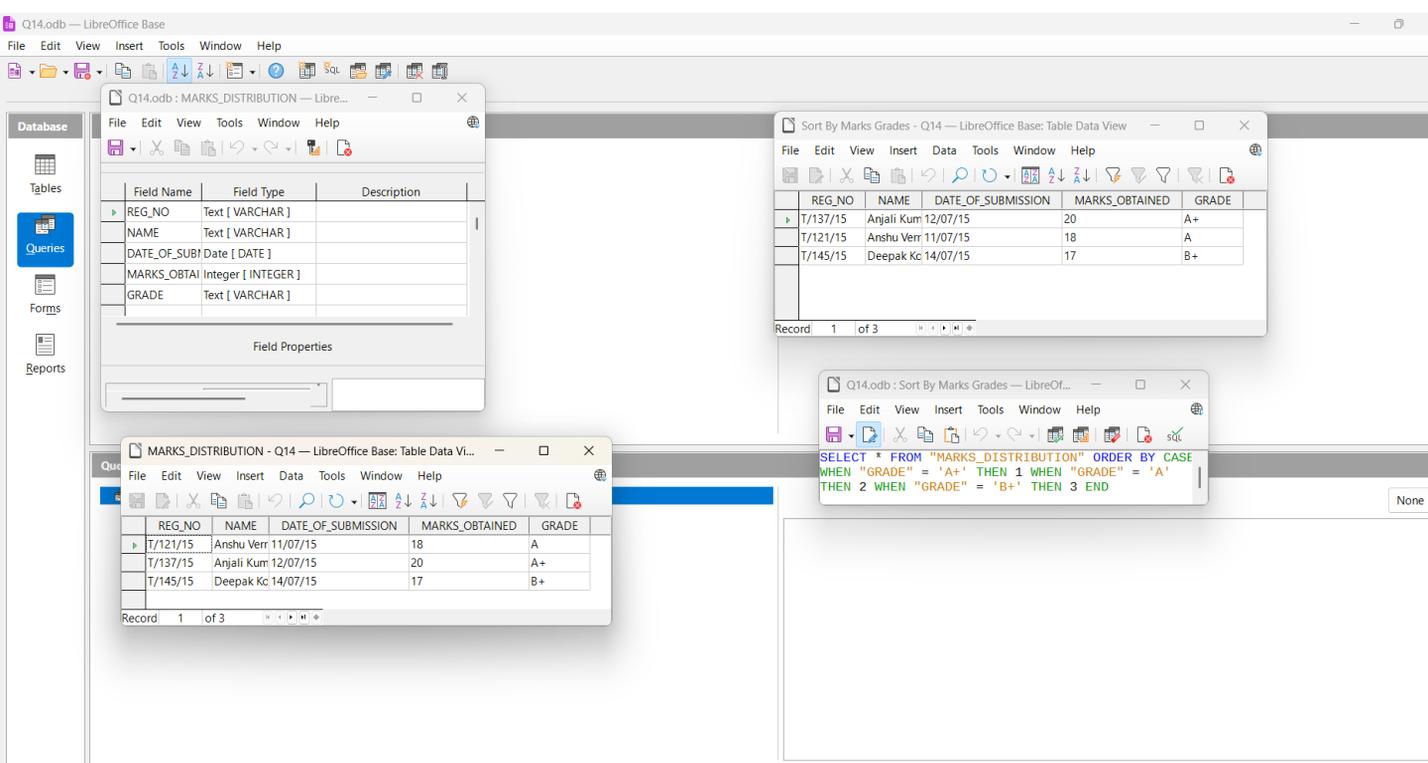
Answer

To identify and reconcile discrepancies between two versions of a financial report, both sheets are opened in **LibreOffice Calc**. The **Compare Document** feature is used to compare the two reports and highlight differences. The **Track Changes** option helps in reviewing all modifications made in the sheets. Required changes are accepted and incorrect entries are rejected to reconcile the data. Finally, the corrected spreadsheet is saved to ensure accuracy and consistency in the financial report.

Q14 Create the following table using the Design View in LibreOffice Base:

REG_NO	NAME	DATE_OF_SUBMISSION	MARKS_OBTAINED	GRADE
T/121/15	Anshu Verr	11/07/15	18	A
T/137/15	Anjali Kum	12/07/15	20	A+
T/145/15	Deepak Kc	14/07/15	17	B+

- Name the table as “**MARKS_DISTRIBUTION**”.
- Enter at least **three records** in the table.
- **Sort the data on the basis of grades.**



Step 1:

Open **LibreOffice Base**.

Step 2:

Click **File** → **New** → **Database**.

Step 3:

Select **Create a new database** → Click **Next** → Click **Finish**.
Save the database (e.g., Q14 .odb).

Steps to Create Table Using Design View

Step 4:

Click on **Tables** in the left panel.

Step 5:

Click **Create Table in Design View**.

Step 6:

Enter the following fields and data types:

- REG_NO – Text [VARCHAR]
- NAME – Text [VARCHAR]
- DATE_OF_SUBMISSION – Date [DATE]
- MARKS_OBTAINED – Integer [INTEGER]
- GRADE – Text [VARCHAR]

Step 7:

Set **REG_NO** as the **Primary Key**.

Step 8:

Save the table with the name **MARKS_DISTRIBUTION**.

Step 9:

Open the table and enter the given records.

Steps to Create Query Using SQL View

Step 10:

Click on **Queries** in the left panel.

Step 11:

Click **Create Query in SQL View**.

Step 12:

Type the following SQL command:

```
SELECT *  
FROM "MARKS_DISTRIBUTION"  
ORDER BY  
CASE  
  WHEN "GRADE" = 'A+' THEN 1  
  WHEN "GRADE" = 'A' THEN 2  
  WHEN "GRADE" = 'B+' THEN 3  
END;
```

Step 13: Click the **Run (▶)** button to execute the query.

Step 14: The records are displayed in sorted order based on grades.

Step 15: Save the query with a suitable name (e.g., **Sort_By_Grade**).

Final Result :- The data is sorted according to grade priority instead of alphabetical order.

Q15 Create the following table using the Design View.

ROLL_NO	NAME	DATE_OF_BIRTH	PHONE_NO	PLACE
A001	Manan Gupta	12/09/87	2354667667	Delhi
A002	Ryansi Sharma	09/05/86	3435555555	Chhattisgarh
A003	Rahul John	02/05/90	6565544324	Mumbai

Perform the following operations:

- **Sort the Place in decreasing order.**
- **Set Roll No. as the Primary Key.**
- **Delete the record A002.**
- **Add another record to the table.**

Step 1: Open LibreOffice Base

1. Open **LibreOffice**.
2. Click on **Base**.
3. Select **Create a new database** → Click **Finish**.
4. Save the database as **Q15.odb**.

Step 2: Create Table using Design View

1. Click on **Tables** from the left panel.
2. Click **Create Table in Design View**.
3. Enter the following fields:

Field Name	Field Type
ROLL_NO	Text (VARCHAR)
NAME	Text (VARCHAR)
DATE_OF_BIRTH	Date (DATE)
PHONE_NO	Text (VARCHAR)
PLACE	Text (VARCHAR)

Step 3: Set Primary Key

1. Right-click on **ROLL_NO**.
2. Click **Primary Key**.
3. A key icon will appear next to ROLL_NO.

Step 4: Save the Table

- Press **Ctrl + S**.
- Name the table as **STUDENTS**.
- Click **OK**.

Step 5: Enter Records

Enter the following records in **Table Data View**:

Roll No	Name	Date of Birth	Phone No	Place
A001	Manan Gupta	12/09/1987	2354667667	Delhi
A002	Ryansh Sharma	09/05/1986	3435555555	Chhattisgarh
A003	Rahul John	02/05/1990	6565544324	Mumbai

Step 6: Sort Place in Decreasing Order (SQL Query)

1. Click on **Queries** → **Create Query in SQL View**.
2. Write the following SQL command:
`SELECT * FROM "STUDENTS" ORDER BY "PLACE" DESC;`
3. Click **Run Query**.
4. Save the query as **place in decreasing order**.

Step 7: Delete Record with Roll No A002

1. Click **Tools** → **SQL**.
2. Enter the following command:
`DELETE FROM "STUDENTS" WHERE "ROLL_NO" = 'A002';`
3. Click **Execute**.
4. The record A002 will be deleted successfully.

Step 8: Add a New Record

1. Again open **Tools** → **SQL**.
2. Write the following SQL command:
 - `INSERT INTO "STUDENTS" ("ROLL_NO", "NAME", "DATE_OF_BIRTH", "PHONE_NO", "PLACE") VALUES ('A004', 'Toshi Dubey', '1992-03-18', '9876543210', 'Indore');`
3. Click **Execute**.

Final Result

The table now contains:

- Primary Key set on **ROLL_NO**
- Records sorted by **Place in decreasing order**
- Record **A002** deleted
- New record **A004** added

The screenshot shows the LibreOffice Base interface with the following components:

- Field Properties:** A table defining the fields for the STUDENTS table:

Field Name	Field Type	Description
ROLL_NO	Text [VARCHAR]	
NAME	Text [VARCHAR]	
DATE_OF_BIRT	Date [DATE]	
PHONE_NO	Text [VARCHAR]	
PLACE	Text [VARCHAR]	
- Table Data View (STUDENTS):** A table showing the current data:

ROLL_NO	NAME	DATE_OF_BIRTH	PHONE_NO	PLACE
A003	Rahul John	02/05/90	6565544324	Mumbai
A001	Manan Gupta	12/09/87	2354667667	Delhi
A002	Ryansi Sharma	09/05/86	3435555555	Chhattisgarh
A003	Rahul John	02/05/90	6565544324	Mumbai
- Table Data View (place in decreasing order):** A table showing the data sorted by Place in decreasing order:

ROLL_NO	NAME	DATE_OF_BIRTH	PHONE_NO	PLACE
A003	Rahul John	02/05/90	6565544324	Mumbai
A001	Manan Gupta	12/09/87	2354667667	Delhi
A002	Ryansi Sharma	09/05/86	3435555555	Chhattisgarh
- SQL Command Window:** Contains the following query:

```
SELECT *
FROM "STUDENTS"
ORDER BY "PLACE" DESC;
```

Execute SQL Statement

SQL Command

Command to execute:

```
DELETE FROM "STUDENTS"
WHERE "ROLL_NO" = 'A002';
```

Run SQL command directly
 Show output of "select" statements **Execute**

Previous commands:

Status

1: Command successfully executed.

Output

1 rows updated

Help **Close**

Execute SQL Statement

SQL Command

Command to execute:

```
INSERT INTO "STUDENTS"
("ROLL_NO", "NAME", "DATE_OF_BIRTH",
"PHONE_NO", "PLACE")
VALUES
('A004', 'Toshi Dubey', '1992-03-18',
'9876543210', 'Indore');
```

Run SQL command directly
 Show output of "select" statements **Execute**

Previous commands:

Status

1: Command successfully executed.
2: Command successfully executed.

Output

1 rows updated

Help **Close**

STUDENTS - Q15 — LibreOffice Base: Table Data View

File Edit View Insert Data Tools Window Help

ROLL_NO	NAME	DATE_OF_BIRTH	PHONE_NO	PLACE
A001	Manan Gupta	12/09/87	2354667667	Delhi
A003	Rahul John	02/05/90	6565544324	Mumbai
A004	Toshi Dubey	18/03/92	9876543210	Indore
+				

Record 1 of 3

Q16 You are a school database that includes two main tables: Students and Classes.

Students Table

StudentID	StudentName	DateOfBirth	ClassID
1	Priya Sharma	15/03/05	101
2	Rahul Kapoor	22/07/04	102
3	Meena Patel	10/11/05	101
4	Aisha Khan	25/02/04	103
5	Dev Rai	30/05/05	102

Classes Table

ClassID	ClassName	TeacherName
101	Mathematics	Mr. Prakash
102	Science	Ms. Yashika
103	History	Mrs. Seema

- **Set up a relationship between the Students table and the Classes table.**
- **Write a query to list all students along with the class they are in and the teacher of that class.**

Step 1: Create a New Database

1. Open **LibreOffice Base**.
2. Click **Create a new database** → **Next** → **Finish**.
3. Save the database as **Q16.odb**.

Step 2: Create CLASSES Table using Design View

1. Click **Tables** → **Create Table in Design View**.
2. Enter the following fields:

Field Name	Field Type
ClassID	Integer
ClassName	Text (VARCHAR)
TeacherName	Text (VARCHAR)

- Set **ClassID** as **Primary Key**.
- Save the table as **CLASSES**.

Step 3: Create STUDENTS Table using Design View

- Click **Tables** → **Create Table in Design View**.
- Enter the following fields:

Field Name	Field Type
StudentID	Integer
StudentName	Text (VARCHAR)
DateOfBirth	Date
ClassID	Integer

- Set **StudentID** as **Primary Key**.
- Save the table as **STUDENTS**.

Step 4: Enter Records in Both Tables

- Enter records in **CLASSES** table (Mathematics, Science, History with teachers).
- Enter records in **STUDENTS** table with proper **ClassID** values.

Step 5: Create Relationship between Tables

- Click **Tools** → **Relationships**.
- Add **CLASSES** and **STUDENTS** tables.
- Drag **ClassID** from **CLASSES** to **ClassID** in **STUDENTS**

1. A **one-to-many relationship (1:n)** is created.
2. Save and close the relationship window.

Step 6: Create Query using SQL View

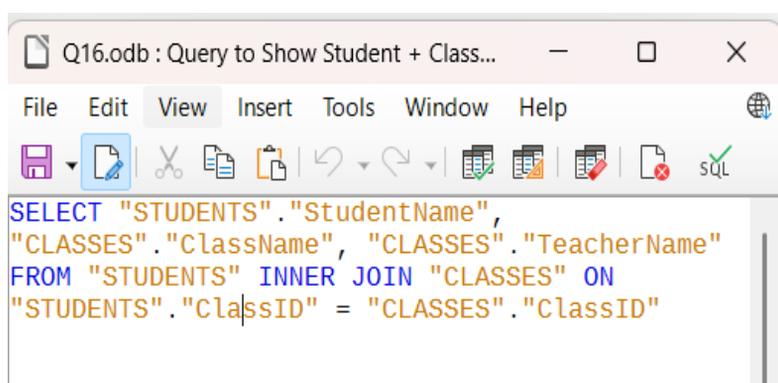
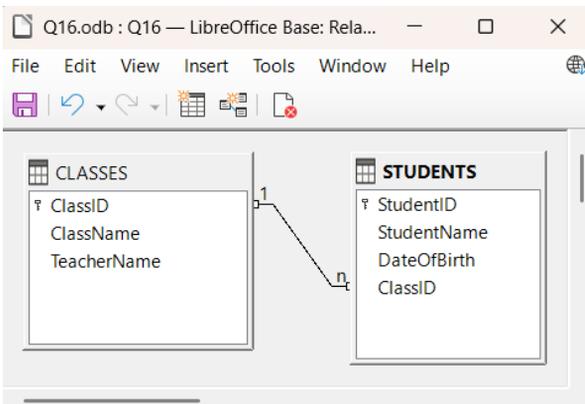
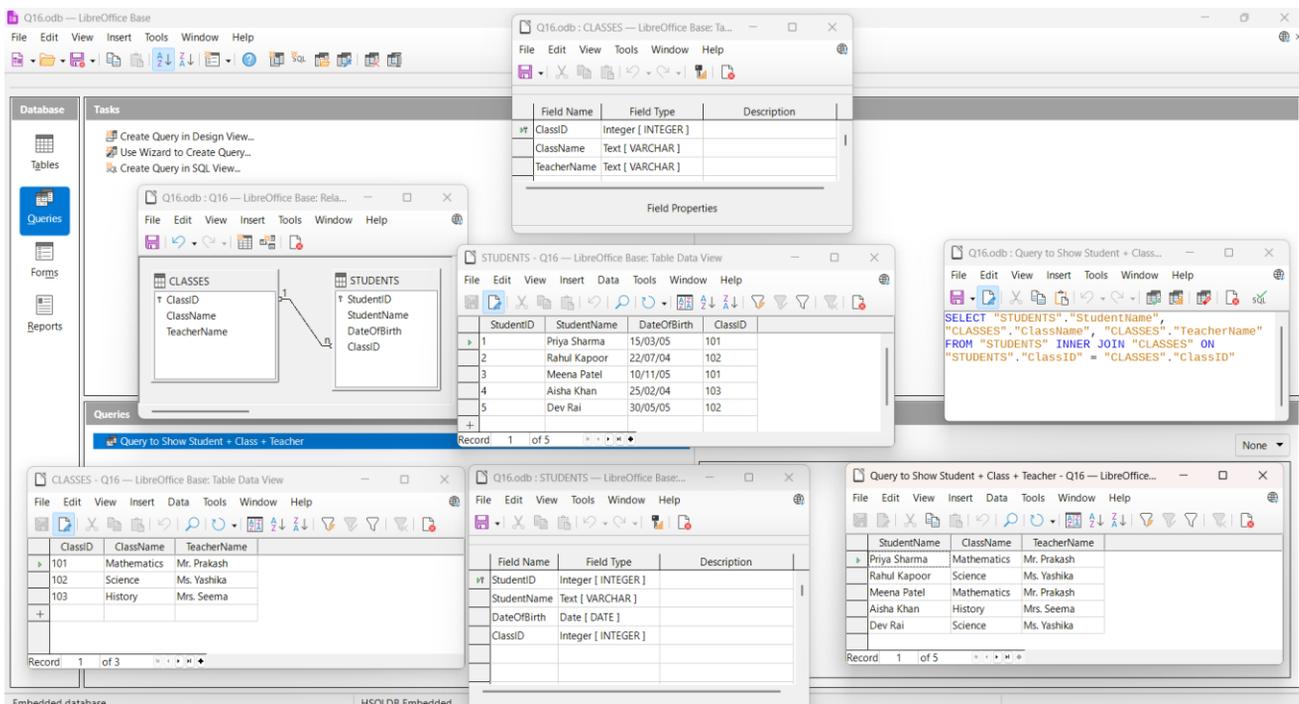
1. Click **Queries** → **Create Query in SQL View**.
2. Type the following SQL command:

```
SELECT "STUDENTS"."StudentName", "CLASSES"."ClassName",
"CLASSES"."TeacherName" FROM "STUDENTS" INNER JOIN
"CLASSES" ON "STUDENTS"."ClassID" = "CLASSES"."ClassID";
```

3. Click **Run Query (F5)**.

Step 7: Save the Query

1. Save the query as
Query to Show Student + Class + Teacher.



Q17 Create the following table for the Students' Details:

	BookID	StudentName	Class	Section
	1	Amit Kumar	7	C
	2	Ria Aggarwal	8	A
	3	Yash Yadav	6	D
	4	Ananya Gupta	8	B
	5	Praveen Soni	7	C

- Create a form for the *Students' Details* table.
- Enter the following records through the form.

BookID	StudentName	Class	Section
6	Gyan Shankar	8	B
7	Richa Sinha	6	A

Step 1: Create Database

- Open **LibreOffice Base**.
- Select **Create a new database** → Click **Next** → **Finish**.
- Save the database as **Q17.odb**.

Step 2: Create Table in Design View

- Click **Tables** → **Create Table in Design View**.
- Create the following fields:

Field Name	Field Type
BookID	Integer (AutoValue = Yes, Primary Key)
StudentName	Text (VARCHAR)
Class	Integer
Section	Text (VARCHAR)

- Set **BookID** as **Primary Key** and **AutoValue = Yes**.
- Save the table as **STUDENTS_DETAILS**.

Step 3: Insert Records in Table

Enter the following records (BookID will be auto-generated):

StudentName	Class	Section
Amit Kumar	7	C
Ria Aggarwal	8	A
Yash Yadav	6	D
Ananya Gupta	8	B
Praveen Soni	7	C

Step 4: Create Form

1. Click **Forms** → **Use Wizard to Create Form**.
2. Select table **STUDENTS_DETAILS**.
3. Move all fields to selected fields.
4. Click **Next** → **Next** and choose **Columnar Layout**.
5. Click **Finish** and save the form as **STUDENTS_FORM**.

Step 5: Enter Records Through Form

Using **STUDENTS_FORM**, add the following records:

StudentName	Class	Section
Toshi Dubey	10	A
Sanskar Parihar	10	A

(BookID is generated automatically starting from 1)

Step 6: Result

All records are successfully entered through the form and BookID is auto-generated.

Conclusion

The Students' Details table was created using Design View, a form was designed using Form Wizard, and records were entered successfully through the form.

Q17.odb : STUDENTS_DETAILS — LibreOffice Base: Database Form

File Edit View Insert Format Styles Table Form Tools Window Help

StudentName
Toshi Dubey

Class
10

Section
A

Record 6 of 7

Q17.odb : STUDENTS_DETAILS — LibreOffice Base: Database Form

File Edit View Insert Format Styles Table Form Tools Window Help

StudentName
Sanskar Parihar

Class
10

Section
A

Record 7 of 7

STUDENTS_DETAILS - Q17 — Libr...

File Edit View Insert Data Tools Window Help

	BookID	StudentName	Class	Section
▶	1	Amit Kumar	7	C
	2	Ria Aggarwal	8	A
	3	Yash Yadav	6	D
	4	Ananya Gupta	8	B
	5	Praveen Soni	7	C
	6	Toshi Dubey	10	A
	7	Sanskar Parihar	10	A
+	<AutoField:			

Record 1 of 7

Q18. Consider the following spreadsheet:

Product	Units Sold
A	69
B	73
C	88
D	82

Average Unit Sold is calculated in **cell B8**.

1. Calculate the **average units sold** in cell **B8**.

2. **Maximise the average number of units sold to 95** using **Goal Seek**.

3. The criteria for maximising the average number of units sold is as follows:

1. **B2 ≤ 88**

2. **B3 ≥ 82**

3. **B4 ≤ 90**

4. **B5 ≥ 60**

Step 1: Enter the data

1. Open **LibreOffice Calc**.

2. Enter Units Sold values in column B:

1. **B2 = 69**

2. **B3 = 73**

3. **B4 = 88**

4. **B5 = 82**

Step 2: Calculate the Average

3. In cell **B8**, enter:

=AVERAGE(B2:B5)

4. Press **Enter**.

5. Initial average = **78**.

Step 3: Apply Goal Seek

6. Go to **Tools → Goal Seek**.

7. Fill the dialog box:

6. **Formula cell:** **\$B\$8**

7. **Target value:** **95**

8.Variable cell: B2

1. Click **OK**.

👉 LibreOffice Calc changes the value of **B2**.

👉 **Average becomes 95.**

Step 4: Verify the Conditions

Now check the given conditions:

- **B2 ≤ 88** ❌ (B2 becomes greater than 88)
- **B3 ≥ 82** ❌ (B3 is still 73)
- **B4 ≤ 90** ✔️
- **B5 ≥ 60** ✔️

Important Observation

✔️ The **average value 95 is achieved**

❌ **But all the given conditions are NOT satisfied**

Goal Seek changes **only one cell**, ignoring the remaining constraints.

PROOF (Why conditions are violated)

Reason 1: Single-variable limitation

- Goal Seek works on **only one variable cell at a time**
- It does **not enforce multiple conditions simultaneously**

Reason 2: Mathematical contradiction

To maintain all conditions, maximum possible values are:

$$B2 = 88$$

$$B3 = 82$$

$$B4 = 90$$

$$B5 = 60$$

Maximum possible average:

$$(88 + 82 + 90 + 60) \div 4 = 80$$

But achieved average is: 95

Therefore, reaching **95 automatically breaks at least one condition.**

Q19 Create a table named *Sales* with the columns: *product_id*, *product_name*, *unit_price*, and *units_sold*.

Insert some sample data into the *Sales* table. You can include at least 5 different products with varying prices and unit sales.

Display all products and their total sales amount ($\text{unit_price} \times \text{units_sold}$).

Find the product with the highest total sales amount.

Show all products with a unit price greater than a specific value (user input).

Calculate the average unit price across all products.

The screenshot displays the LibreOffice Base interface for a database named 'Q19'. The main window shows the 'SALES' table in Table Data View with the following data:

PRODUCT_ID	PRODUCT_NAME	UNIT_PRICE	UNITS_SOLD
1	Laptop	50000	10
2	Mobile	20000	25
3	Tablet	15000	18
4	Printer	12000	12
5	Headphones	3000	40

Below the table, the 'Queries' pane lists several queries, with 'Find the product with the highest total sales' selected. Several SQL query windows are open, showing the following queries:

- `SELECT * FROM "SALES" WHERE "UNIT_PRICE" > 15000 // user defined this value 15000`
- `SELECT "PRODUCT_NAME", ("UNIT_PRICE" * "UNITS_SOLD") AS "TOTAL_SALES" FROM "SALES" ORDER BY "TOTAL_SALES" DESC LIMIT 1`
- `SELECT AVG("UNIT_PRICE") AS "AVERAGE_UNIT_PRICE" FROM "SALES"`
- `SELECT "PRODUCT_ID", "PRODUCT_NAME", "UNIT_PRICE", "UNITS_SOLD", ("UNIT_PRICE" * "UNITS_SOLD") AS "TOTAL_SALES" FROM "SALES"`

The 'Display all products with Total Sales Amount' query window shows the following results:

PRODUCT_ID	PRODUCT_NAME	UNIT_PRICE	UNITS_SOLD	TOTAL_SALES
1	Laptop	50000	10	500000
2	Mobile	20000	25	500000
3	Tablet	15000	18	270000
4	Printer	12000	12	144000
5	Headphones	3000	40	120000

Step 1: Create a New Database

- Open **LibreOffice** → **Base**
- Select **Create a new database**
- Click **Finish**
- Save the file as **Q19.odb**

Step 2: Create Table in Design View

- Click **Tables** → **Create Table in Design View**
- Enter the following fields:

Field Name	Field Type
PRODUCT_ID	Integer
PRODUCT_NAME	Text (VARCHAR)
UNIT_PRICE	Integer
UNITS_SOLD	Integer

- Set **PRODUCT_ID** as **Primary Key**
- Save table as **SALES**

Step 3: Insert Sample Data

Enter at least **5 records**, for example:

PRODUCT_ID	PRODUCT_NAME	UNIT_PRICE	UNITS_SOLD
1	Laptop	50000	10
2	Mobile	20000	25
3	Tablet	15000	18
4	Printer	12000	12
5	Headphones	3000	40

Queries (SQL View)

Step 4: Display All Products with Total Sales

1. Click **Queries** → **Create Query in SQL View**

2. Write:

```
SELECT "PRODUCT_ID", "PRODUCT_NAME", "UNIT_PRICE",
"UNITS_SOLD", ("UNIT_PRICE" * "UNITS_SOLD") AS
"TOTAL_SALES" FROM "SALES";
```

3. Click **Run**

4. Save as **Display Total Sales**

Step 5: Find Product with Highest Total Sales

1. Create new SQL Query

2. Write:

```
SELECT "PRODUCT_NAME", ("UNIT_PRICE" * "UNITS_SOLD")
AS"TOTAL_SALES" FROM "SALES" ORDER BY "TOTAL_SALES"
DESCLIMIT 1;
```

3. Run and save as **Highest Total Sales**

Step 6: Show Products with Unit Price Greater Than a Value

(Example: greater than 15000)

```
SELECT * FROM "SALES" WHERE "UNIT_PRICE" > 15000;
```

Save as **Unit Price Greater Than Value**

Step 7: Calculate Average Unit Price

1. Create new SQL Query

2. Write:

```
SELECT AVG("UNIT_PRICE") AS "AVERAGE_UNIT_PRICE"
FROM"SALES";
```

3. Run and save as **Average Unit Price**

Q20 Consider the following Employee table.

	EmployeeID	FirstName	LastName	Department	Salary
▶	1	Aarav	Patel	IT	60000
	2	Riya	Sharma	HR	55000
	3	Kabir	Gupta	Sales	62000
	4	Ananya	Reddy	IT	58000
	5	Advik	Joshi	Sales	60000
+					

Create a report using the Employee table.

Format the report according to the requirement.

Add the page number in the footer of the report.

Step 1: Open LibreOffice Base

- Open **LibreOffice Base**.
- Open your database file **Q20.odt**.

Step 2: Create the EMPLOYEE Table

1. Click on **Tables** → **Create Table in Design View**.
2. Create the following fields:

Field Name	Data Type
EmployeeID	Integer (Primary Key)
FirstName	Text (VARCHAR)
LastName	Text (VARCHAR)
Department	Text (VARCHAR)
Salary	Integer

- Set **EmployeeID** as **Primary Key**.
- Save the table as **EMPLOYEE**.

Step 3: Insert Records into EMPLOYEE Table

Open the table and insert the following records:

EmployeeID	FirstName	LastName	Department	Salary
1	Aarav	Patel	IT	60000
2	Riya	Sharma	HR	55000
3	Kabir	Gupta	Sales	62000
4	Ananya	Reddy	IT	58000
5	Advik	Joshi	Sales	60000

Save and close the table.

Step 4: Create Report Using Wizard

- Click on **Reports** → **Use Wizard to Create Report**.
- Select table **EMPLOYEE**.
- Move all fields to the right:
 - EmployeeID
 - FirstName
 - LastName
 - Department
 - Salary
- Click **Next**.
- No grouping → **Next**.

- Sorting not required → **Next**.
- Select layout **Tabular**.
- Orientation **Portrait**.
- Click **Finish**.
- Save report as **EMPLOYEE_REPORT**.

Step 5: Open Report in Design View

- Right-click on **EMPLOYEE_REPORT**.
- Select **Edit**.

Step 6: Format the Report

- Increase font size of field labels.
- Make field names **Bold**.
- Align text properly in **Detail section**.
- Insert logo image in **Page Header** (as shown).

Step 7: Add Page Number in Footer

- Click on **Page Footer** section.
- Insert a **Label Field**.
- Type the following expression:

= "Page " & PageNum

- Align it to the center.

Step 8: Save and View Report

- Save the report.
- Click **View Report**.
- Verify:
 - All employee records are shown.
 - Formatting is correct.
 - Page number appears at the bottom.

Final Output

- A properly formatted **Employee Report**
- Uses **EMPLOYEE** table
- Includes **Page Number**
- Matches the given question exactly

