4 | SPREADSHEETS AND DATABASES

Introduction

In the healthcare, the management and analysis of health data are critical for improving patient care, conducting research, and informing policy decisions. Two types of software, in particular, have become indispensable in this regard. They are spreadsheet software and database software. Both offer unique capabilities for storing, organizing, and analyzing health data, but they also have distinct characteristics that make them suitable for different purposes.

Spreadsheet software, such as Microsoft Excel or Google Sheets, provides a user-friendly interface for data entry and manipulation. It allows users to organize data in rows and columns, perform calculations, create charts, and apply various data analysis tools. For small datasets and simple analyses, spreadsheet software can be a quick and intuitive solution. However, it may not be ideal for managing large or complex datasets due to limitations in data capacity, performance, and data integrity controls.

On the other hand, database software, such as MySQL, Oracle, or Microsoft SQL Server, is designed to handle larger and more complex datasets. Databases can store vast amounts of data, maintain data integrity, and provide powerful tools for data querying and reporting. They allow for the creation of relationships between different data elements, which can be crucial in healthcare settings where patient data may be spread across multiple tables or databases.

Both spreadsheet and database software play vital roles in health data science. The choice between them depends on the size and complexity of the dataset, the required tasks, and the user's technical expertise. Understanding their capabilities and limitations is key to leveraging these tools effectively in the ever-evolving landscape of health data.

Spreadsheets

This section provides a short introduction to the use of spreadsheet software for health data management.

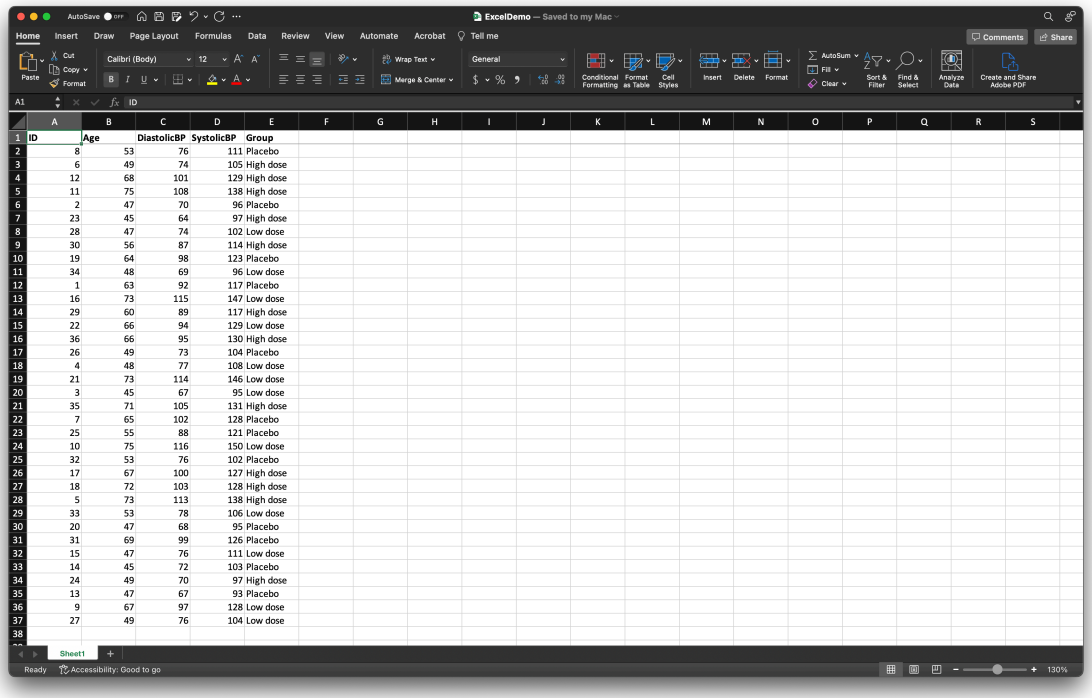
Spreadsheet Basics

A spreadsheet is a computer program that allows users to organize data in rows and columns. It is often used for storing, manipulating, and analyzing data in tabular form.

The most popular spreadsheet software is Microsoft Excel, but there are many other options available, such as Google Sheets, LibreOffice Calc, and Apple Numbers.

A spreadsheet consists of a grid of cells arranged in rows and columns. Each cell can contain a value, a formula, or a function. Values can be numbers, text, or dates. Formulas are mathematical expressions that perform calculations on values in other cells. Functions are predefined formulas that perform specific tasks, such as calculating the sum of a range of cells or finding the average of a set of values.

The following figure shows an example of a spreadsheet in Microsoft Excel. It contains data on the age and diastolic and systolic blood pressure of participants on three interventions.



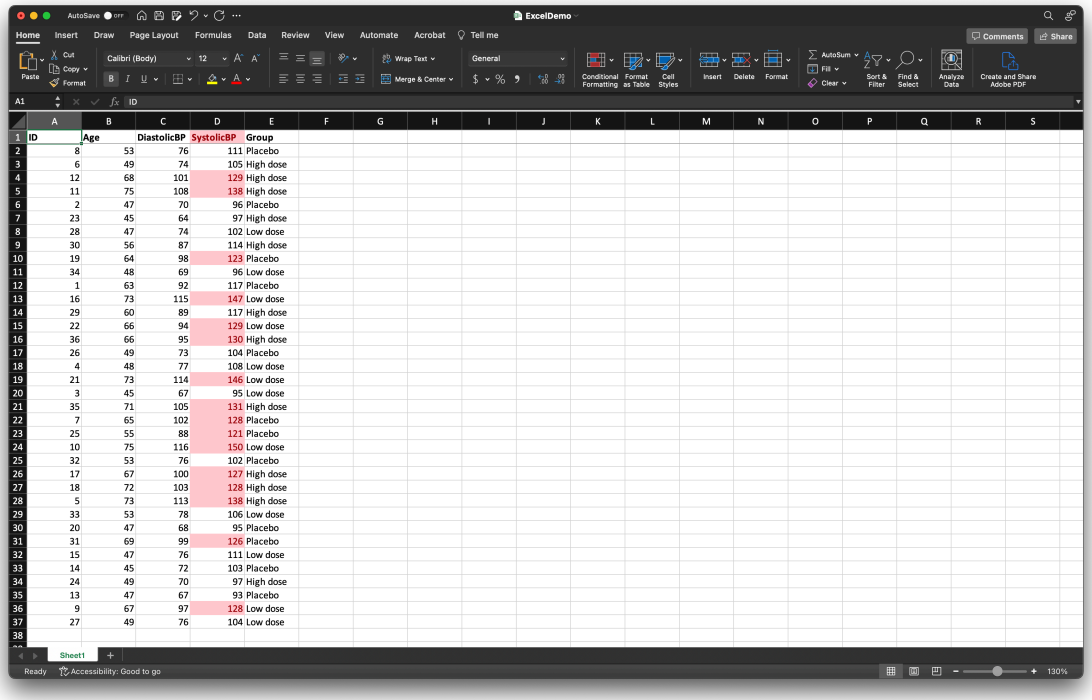
ID	Age	DiastolicBP	SystolicBP	Group
8	53	76	111	Placebo
6	49	74	105	High dose
12	68	101	129	High dose
11	75	108	138	High dose
2	47	70	96	Placebo
23	45	64	97	High dose
28	47	74	102	Low dose
30	56	87	114	High dose
19	64	98	123	Placebo
34	48	69	96	Low dose
1	63	92	117	Placebo
16	73	115	147	Low dose
29	60	89	117	High dose
22	66	94	129	Low dose
36	66	95	130	High dose
26	49	73	104	Placebo
4	48	77	108	Low dose
21	73	114	146	Low dose
3	45	67	95	Low dose
35	71	105	131	High dose
7	65	102	128	Placebo
25	55	88	121	Placebo
10	75	116	150	Low dose
32	53	76	102	Placebo
17	67	100	127	High dose
18	72	103	128	High dose
5	73	113	138	High dose
33	53	78	106	Low dose
20	47	68	95	Placebo
31	69	99	126	Placebo
15	47	76	111	Low dose
14	45	72	103	Placebo
24	49	70	97	High dose
13	47	67	93	Placebo
9	67	97	128	Low dose
27	49	76	104	Low dose

Figure 4.1: A spreadsheet in Microsoft Excel

Data Entry

One of the main uses of spreadsheets is data entry. They provide a user-friendly interface for entering data into a table format. Users can enter data manually or import it from other sources, such as text files or databases. Spreadsheets also allow users to edit and format data in various ways, such as changing the font size or color, adding borders around cells, or applying conditional formatting rules.

The following figure shows an example of data entry in Microsoft Excel. The data is entered manually into a table format. The entries for the systolic blood pressure values that are in excess of 120 are shaded red.



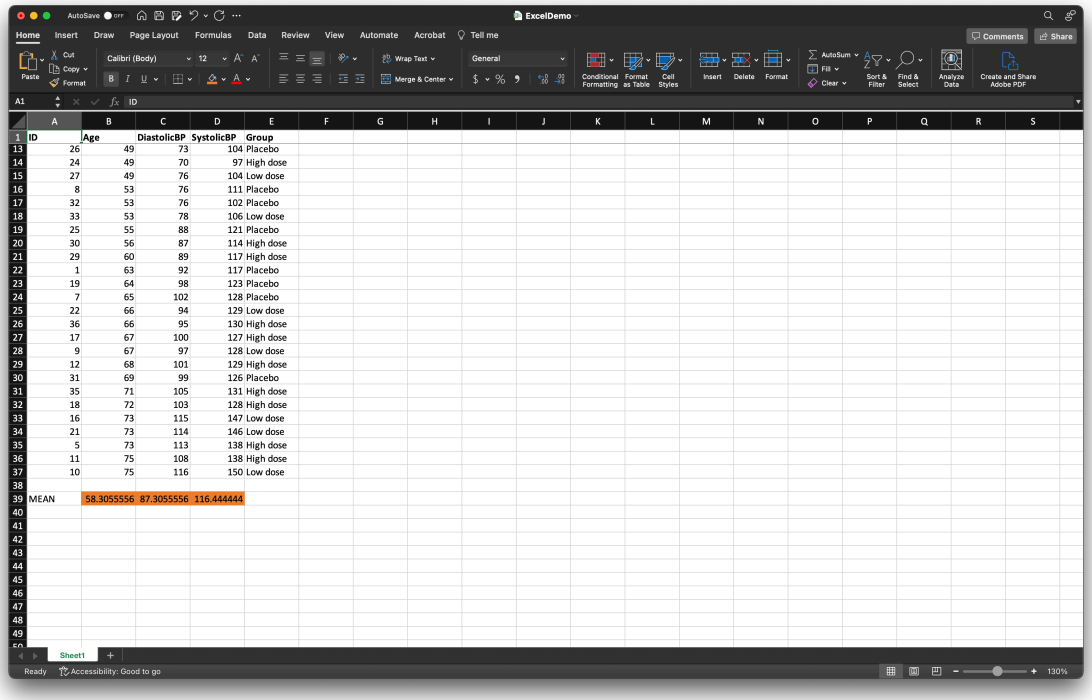
ID	Age	DiastolicBP	SystolicBP	Group
8	53	76	111	Placebo
6	49	74	105	High dose
12	68	101	129	High dose
11	75	108	138	High dose
2	47	70	96	Placebo
23	45	64	97	High dose
28	47	74	102	Low dose
30	56	87	114	High dose
19	64	98	123	Placebo
34	48	69	96	Low dose
1	63	92	117	Placebo
16	73	115	147	Low dose
29	60	89	117	High dose
22	66	94	129	Low dose
36	66	95	130	High dose
26	49	73	104	Placebo
4	48	77	108	Low dose
21	73	114	146	Low dose
3	45	67	95	Low dose
35	71	105	131	High dose
7	65	102	128	Placebo
25	55	88	121	Placebo
10	75	116	150	Low dose
32	53	76	102	Placebo
17	67	100	127	High dose
18	72	103	128	High dose
5	73	113	138	High dose
33	53	78	106	Low dose
20	47	68	95	Placebo
31	69	99	126	Placebo
15	47	76	111	Low dose
14	45	72	103	Placebo
24	49	70	97	High dose
13	47	67	93	Placebo
9	67	97	128	Low dose
27	49	76	104	Low dose

Figure 4.2: Data entry in Microsoft Excel

Data Manipulation

Another common use of spreadsheets is data manipulation. They provide tools for sorting, filtering, and summarizing data in various ways. Users can sort data by one or more columns, filter data based on specific criteria, or summarize data using formulas such as SUM, AVERAGE, or COUNT.

The following figure shows an example of data manipulation in Microsoft Excel. The data is sorted by the age of the particpants in ascending order, , and summarized using the AVERAGE function to calculate the mean of each variable.



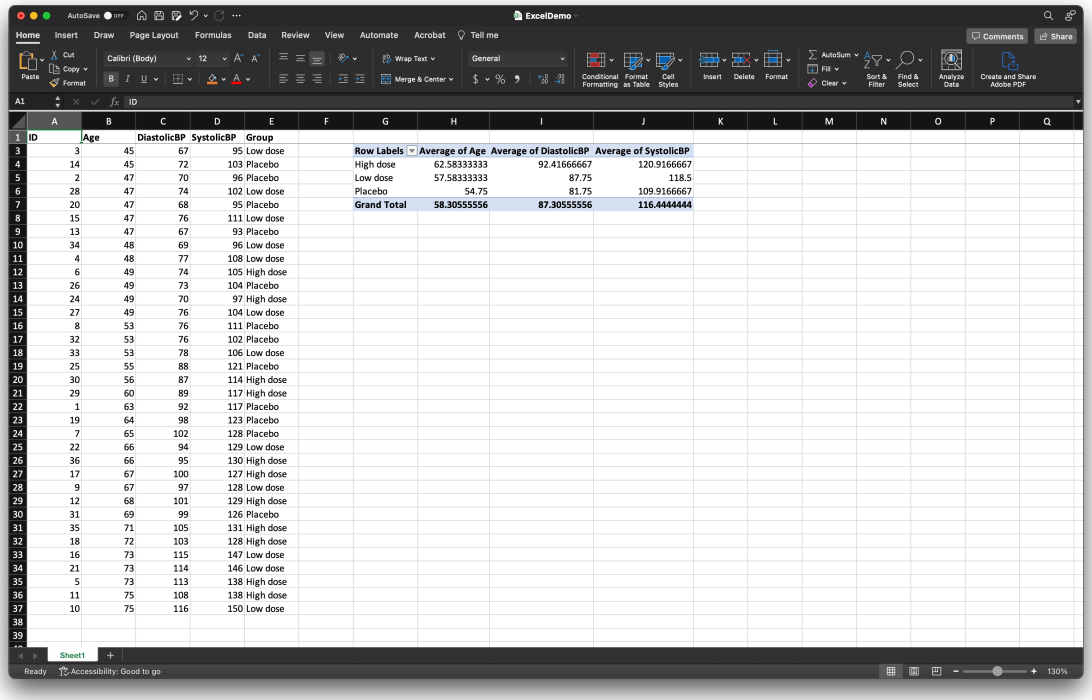
ID	Age	DiastolicBP	SystolicBP	Group
26	49	73	104	Placebo
24	49	70	97	High dose
27	49	76	104	Low dose
8	53	76	111	Placebo
32	53	76	102	Placebo
33	53	78	106	Low dose
25	55	88	121	Placebo
30	56	87	114	High dose
29	60	89	117	High dose
1	63	92	117	Placebo
19	64	98	123	Placebo
7	65	102	128	Placebo
22	66	94	129	Low dose
36	66	95	130	High dose
17	67	100	127	High dose
9	67	97	128	Low dose
12	68	101	129	High dose
31	69	99	126	Placebo
35	71	105	131	High dose
18	72	103	128	High dose
16	73	115	147	Low dose
21	73	114	146	Low dose
5	73	113	138	High dose
11	75	108	138	High dose
10	75	116	150	Low dose
MEAN	58.3055556	87.3055556	116.444444	

Figure 4.3: Data manipulation in Microsoft Excel

Data Analysis

Spreadsheets also provide tools for analyzing data in various ways. Users can perform calculations on data using formulas or functions, create charts to visualize data, or apply various data analysis tools such as pivot tables or conditional formatting rules.

The following figure shows an example of data analysis in Microsoft Excel. The data is analyzed using a pivot table to summarize the numerical variable by the classes of the Group (categorical) variable.



ID	Age	DiastolicBP	SystolicBP	Group
3	45	67	95	Low dose
14	45	72	103	Placebo
2	47	70	96	Placebo
28	47	74	102	Low dose
20	47	68	95	Placebo
15	47	76	111	Low dose
13	47	67	93	Placebo
34	48	69	96	Low dose
4	48	77	108	Low dose
6	49	74	105	High dose
26	49	73	104	Placebo
24	49	70	97	High dose
27	49	76	104	Low dose
8	53	76	111	Placebo
32	53	76	102	Placebo
33	53	78	106	Low dose
25	55	88	121	Placebo
30	56	87	114	High dose
29	60	89	117	High dose
1	63	92	117	Placebo
19	64	98	123	Placebo
7	65	102	128	Placebo
22	66	94	129	Low dose
36	66	95	130	High dose
17	67	100	127	High dose
9	67	97	128	Low dose
12	68	101	129	High dose
31	69	99	126	Placebo
35	71	105	131	High dose
18	72	103	128	High dose
16	73	115	147	Low dose
21	73	114	146	Low dose
5	73	113	138	High dose
11	75	108	138	High dose
10	75	116	150	Low dose

Row Labels	Average of Age	Average of DiastolicBP	Average of SystolicBP
High dose	62.58333333	92.41666667	120.9166667
Low dose	57.58333333	87.75	118.5
Placebo	54.75	81.75	109.9166667
Grand Total	**58.30555556**	**87.30555556**	**116.4444444**

Figure 4.4: Data analysis in Microsoft Excel

Data Visualization

Finally, spreadsheets provide tools for visualizing data in various ways. Users can create charts to visualize data in a graphical format, such as bar charts or pie charts. They can also apply conditional formatting rules to highlight specific data points or trends in the data.

The following figure shows an example of data visualization in Microsoft Excel. A chart is created to visualize the correlation between the age and diastolic blood pressure values.

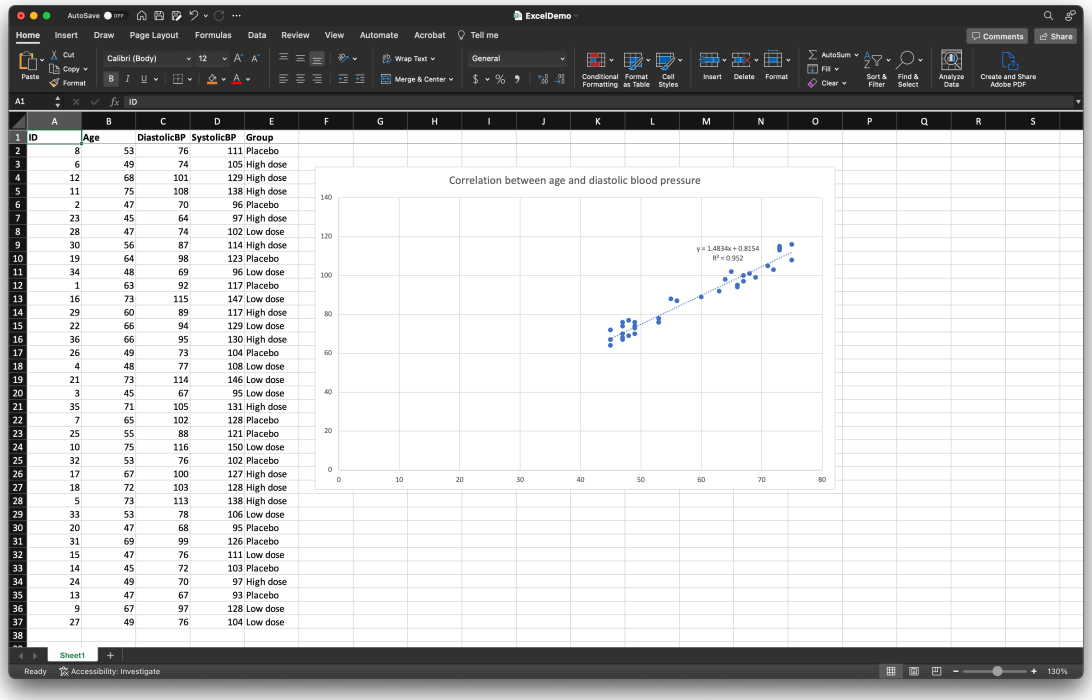


Figure 4.5: Data visualization in Microsoft Excel

While data entry and manipulation can be fairly straight forward using spreadsheet software, consideration must be given to the *layout of the data*. The key concept here is that of tidy data.

Tidy data

Data is at the heart of any analytical task, be it simple data exploration or building complex predictive models. However, raw data is often messy and not suitable for analysis directly. It is estimated that data scientists spend about 80% of their time cleaning and preprocessing data, which makes it a crucial step in the data analysis process. One key aspect of data preprocessing is transforming the data into a 'tidy' format.

The concept of tidy data was introduced by statistician Hadley Wickham in the paper Tidy Data, with the aim of providing a standard way to organize data values within a dataset. According to Wickham, tidy data is a standard way of mapping the meaning of a dataset to its structure. A dataset is tidy when each variable forms a column, each observation forms a row, and each type of observational unit forms a table.

The tidy data framework offers several benefits. For one, it makes data cleaning and preprocessing more systematic and less error-prone. Moreover, tidy data structure is optimized for vectorized programming languages like R or Python, allowing for efficient code that is easier to write, read, and debug. Lastly, because it is a standard, tidy data allows for the development of reusable tools that can work with many different datasets.

Tidy data principles are widely applied in statistical modeling and visualization. Many R packages, for instance, assume data is in a tidy format. Packages like `ggplot2` for visualization, `dplyr` for data manipulation, and 'modelr' for modeling are designed to work with tidy data. Therefore, ensuring your data is tidy can make your analysis more efficient and your code more reusable.

Tidying data often involves several steps. First, the reshape needs to be reshaped so that each variable is a column, each observation is a row, and each type of observational unit is a table. This can involve merging multiple datasets, splitting a single dataset into multiple ones, or transposing the dataset.

Missing values and outliers may need to be considered, as these can affect the quality of your analysis. How this is handled will depend on the specific context and the nature of the data.

Finally, the variables need to be transformed to make them easier to work with. This can involve scaling numeric variables, converting categorical variables into dummy variables, or creating new variables by combining existing ones.

Tidy data is a powerful concept that simplifies the initial stages of data analysis. By providing a standard way to organize data, it makes data cleaning and preprocessing more efficient and less error-prone. The principles of tidy data are now deeply integrated into many tools for data analysis and provide a solid foundation for any data analytical task. Despite the time and effort required to tidy data, the benefits are worth it, yielding more robust and reliable analytical results. It is best to spend the time when designing a spreadsheet to ensure that the data is tidy, rather than trying to tidy it later.

File formats

Spreadsheet software packages save files in proprietary and unique file formats. These formats add additional features to the spreadsheet, such as macros or formulas, which are not supported by other spreadsheet software packages. For example, Microsoft Excel saves files in the .xlsx format, while Google Sheets saves files in the .gsheet format. The additional data stored in these files formats can cause compatibility issues when importing files into data analysis tools such as R or Python.

All spreadsheet software programs can export spreadsheet as comma-separated values (CSV) files. These files contain only the data in the spreadsheet and do not include any additional features such as macros or formulas. They can be imported into data analysis tools such as R or Python without any compatibility issues.

A CSV file is a simple file format. It has several advantages, particularly when compared to other formats such as XLSX or ODS. Four such advantages are listed below.

1. **Universality and Compatibility**: CSV is a simple, plain text format that can be read and written by many programs, including most spreadsheet software (like Excel, Google Sheets, or LibreOffice Calc), many database management systems, and programming languages. This makes it ideal for data exchange between applications, even ones that run on different platforms.

2. **Simplicity**: Because it is a text-based file format, you can open, read, and edit CSV files using a plain text editor if needed. Each line of the file corresponds to a row in the table, and commas separate the values (or fields) within each row.

3. **Size**: CSV files are generally smaller in size compared to other formats like XLSX, as they contain no formatting, formulas, macros, or other extra features. This makes them more efficient for storing and transferring large volumes of data.

4. **Import and Export**: CSV is often used to import and export data from web and mobile applications, databases, and data analysis tools. It is one of the most commonly supported formats for data import in various software tools.

However, keep in mind that while CSV files have these advantages, they do not support features such as cell formatting, formulas, charts, or images that other file formats like XLSX support. Therefore, the choice of format should depend on your specific needs. Also note that other so-called delimited file formats are available such as tab-delimited files (TSV) or pipe-delimited files (PSV). These formats are similar to CSV but use different delimiters (e.g., tabs or pipes) instead of commas to separate values within each row.

Databases

A health database is a structured collection of health-related data, often managed and stored in an organized manner for easy retrieval and analysis. It can include various types of health information, such as electronic health records (EHRs), insurance claims, pharmaceutical research data, patient demographics, clinical trials data, laboratory results, imaging data, and public health statistics. This data can be both structured (e.g., numerical values, categorical variables) and unstructured (e.g., clinical notes, radiology reports).

Health databases are crucial for a wide range of purposes, including clinical decision support, research, public health reporting, patient care management, and health policy development. They can facilitate interoperability, enhance the quality and safety of healthcare delivery, and provide the foundation for predictive analytics and precision medicine. Databases can help healthcare organizations improve patient care, reduce costs, and increase efficiency. It can also provide valuable insights into the health of populations and inform policy decisions. Moreover, databases are crucial tools for the collection of data for research purposes.

The creation of a database takes some planning and effort, but it can be a worthwhile investment in the long run. The key components of a databse are tables and the relationships between them. Tables are used to store data in a structured format, and relationships are used to link data across tables.

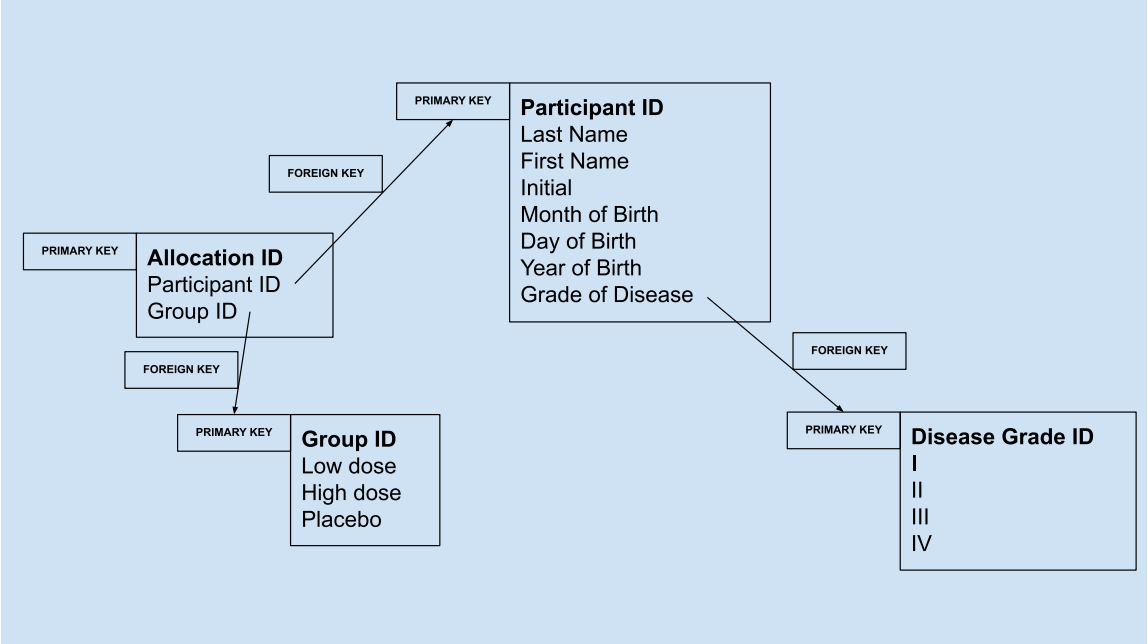


Figure 4.: Table and Relationships

Another key concept in database design is normalization.

Database normalization

Database normalization is a systematic process used in relational database design to organize data to reduce redundancy and improve data integrity. The primary objective of normalization is to eliminate anomalies (insertion, update, and deletion anomalies) that can occur when data is added, modified, or deleted, which can lead to loss of data consistency.

The process of normalization involves dividing larger tables into smaller, less redundant tables and defining relationships between them. The relationships between these tables are established based on a set of rules or **normal forms**. Each rule corresponds to a normal form. There are six normal forms, each with a progressively stricter set of requirements: First Normal Form (1NF), Second Normal Form (2NF), Third Normal Form (3NF), Boyce-Codd Normal Form (BCNF), Fourth Normal Form (4NF), and Fifth Normal Form (5NF, also known as Project-Join Normal Form (PJNF)).

While database normalization helps to keep the database design clean, flexible, and efficient, it's important to note that it's not always necessary to achieve the highest level

of normalization. Sometimes, denormalization (intentionally adding redundancy to data) can be beneficial for performance in read-heavy applications. It's a matter of balance between performance, complexity, and data integrity.

The first three normal forms are discussed below. The remaining three normal forms are beyond the scope of this course.

First Normal Form (1NF) is the initial step towards normalization in database design. A table is said to be in 1NF if it follows the following three rules.

1. Each table should have a primary key: Unique identifier for each record in the table.
2. Each column in the table should contain atomic (indivisible) values, meaning each cell should contain a single value. There should be no repeating groups or arrays.
3. Values stored in a column should be of the same domain, meaning they should be of the same type (integer, string, date, etc.).

For example, the table below is in non-normal form.

PatientID	Name	Complaints
1	John Doe	Headache, runny nose
2	Jane Doe	Swollen ankles, Shortness of breath
3	Mary Jane	Fever, Runny nose

This table is not in 1NF because the "Purchased Items" column contains multiple values.

To bring this table into first normal form (1NF), it is modified so that each cell in the table contains only atomic (single) values.

PatientID	Name	Complaints
1	John Doe	Headache
1	John Doe	Runny nose
2	Jane Doe	Swollen ankles
2	Jane Doe	Shortness of breath
3	Mary Jane	Fever
3	Mary Jane	Runny nose

In the normalized version, each row represents a unique transaction and each cell contains a single value, making the table compliant with the 1NF.

The Second Normal Form (2NF) is a level of database normalization that extends the First Normal Form (1NF) by ensuring that all non-prime attributes (attributes not part of

the primary key) in the table are fully functionally dependent on the primary key.

A table is said to be in 2NF if it adheres to the following two rules.

1. The table is in 1NF.
2. All non-prime attributes in the table must depend on the whole of a candidate key, not just part of it.

For example, consider the following table:

StudentID	Subject	Lecturer
1	Math	Prof. Johnson
1	Science	Prof. Smith
2	Math	Prof. Johnson
3	Science	Prof. Smith

It is assumed that the combination of StudentID and Subject forms a composite primary key. The Lecturer column depends on the Subject column, but not on the whole primary key (it doesn't depend on StudentID). This violates the rules of 2NF.

To bring this table into 2NF, it is split into two tables.

Student_Subject Table:

StudentID	Subject
1	Math
1	Science
2	Math
3	Science

Subject_Lecturer Table:

Subject	Lecturer
Math	Prof. Johnson
Science	Prof. Smith

Now, in the Student_Subject table, each subject chosen by the student forms a unique record. In the Subject_Lecturer table, each subject has a unique lecturer assigned. Both tables are now in 2NF because every non-prime attribute (in this case, only Lecturer) is fully functionally dependent on the primary key.

The Third Normal Form (3NF) is a level of database normalization that extends the Second Normal Form (2NF) by ensuring that all non-prime attributes in the table are not

transitively dependent on the primary key. In other words, no non-prime attribute should depend on another non-prime attribute.

A table is in 3NF if it adheres to the following two rules.

1. The table is in 2NF.
2. Every non-prime attribute is non-transitively dependent on every candidate key in the table.

An example is shown in the table below.

Student Table:

StudentID	CourseID	CourseName
1	C1	Math
2	C2	Science
3	C3	History
4	C1	Math

Here, the primary key is a combination of StudentID and CourseID. However, the attribute CourseName is transitively dependent on the primary key through the CourseID. This means the table is not in 3NF.

To bring this table into 3NF, it is split into two tables.

Student_Course Table:

StudentID	CourseID
1	C1
2	C2
3	C3
4	C1

Course Table:

CourseID	CourseName
C1	Math
C2	Science
C3	History

In these tables, all non-prime attributes (in this case, only CourseName) are non-transitively dependent on the primary key. Hence, both tables are now in 3NF.

REDCap Database

Many institutions, including The George Washington University provides access to the RedCap database for data collection and academic research.

Task: Read the REDCap website of The George Washington University.

Task: Watch the three RECap videos on the website.

Differences between spreadsheets and databases

Below is a table that summarizes the differences between the two major types of software used for health data management: spreadsheets and databases.

Key	Spreadsheet	Database
Primary Use	For calculations and small datasets	To manage and manipulate large datasets
Data Structure	Flat or tabular structure	More complex structures (tables, relations)
Data Volume	Ideal for small volumes of data	Ideal for large volumes of data
Scalability	Limited	High, can manage large amounts of data
Data Relations	Limited	Relationships between data are integral
Data Integrity	Limited, no built-in measures	Built-in measures to ensure data integrity
Multi-user Accessibility	Limited	Multiple users can access and manipulate data simultaneously
Security	Basic security measures	Advanced security measures, user-level access
Complexity	Easy to learn and use	Requires knowledge of database design and SQL
Data Analysis Capabilities	Basic analysis and visualizations	Sophisticated querying, reporting, and analysis

Please note that both spreadsheets and databases have their specific strengths and are well-suited to different types of tasks. The choice between the two should be based on specific needs and requirements.

Links

Below are links to commonly used spreadsheet and database software.

1. Spreadsheet software
2. Database software
3. Microsoft Excel
4. Google Sheets
5. LibreOffice Calc
6. Apple Numbers
7. MySQL
8. Oracle
9. Microsoft SQL Server
10. PostgreSQL

Quiz questions

Questions

1. What is the difference between a spreadsheet and a database?
2. What are the advantages and disadvantages of using a spreadsheet for health data management?
3. What are the advantages and disadvantages of using a database for health data management?
4. What are some examples of tasks that can be performed using a spreadsheet?
5. What are some examples of tasks that can be performed using a database?
6. What are some examples of tasks that can be performed using both a spreadsheet and a database?
7. What are some examples of tasks that can be performed using neither a spreadsheet nor a database?
8. What are some examples of tasks that can be performed using a spreadsheet but not a database?
9. What are some examples of tasks that can be performed using a database but not a spreadsheet?
10. What are some examples of tasks that can be performed using both a spreadsheet and a database?