FIT5196 DATA WRANGLING

Week 11

Data Validation

By Jackie Rong

Faculty of Information Technology

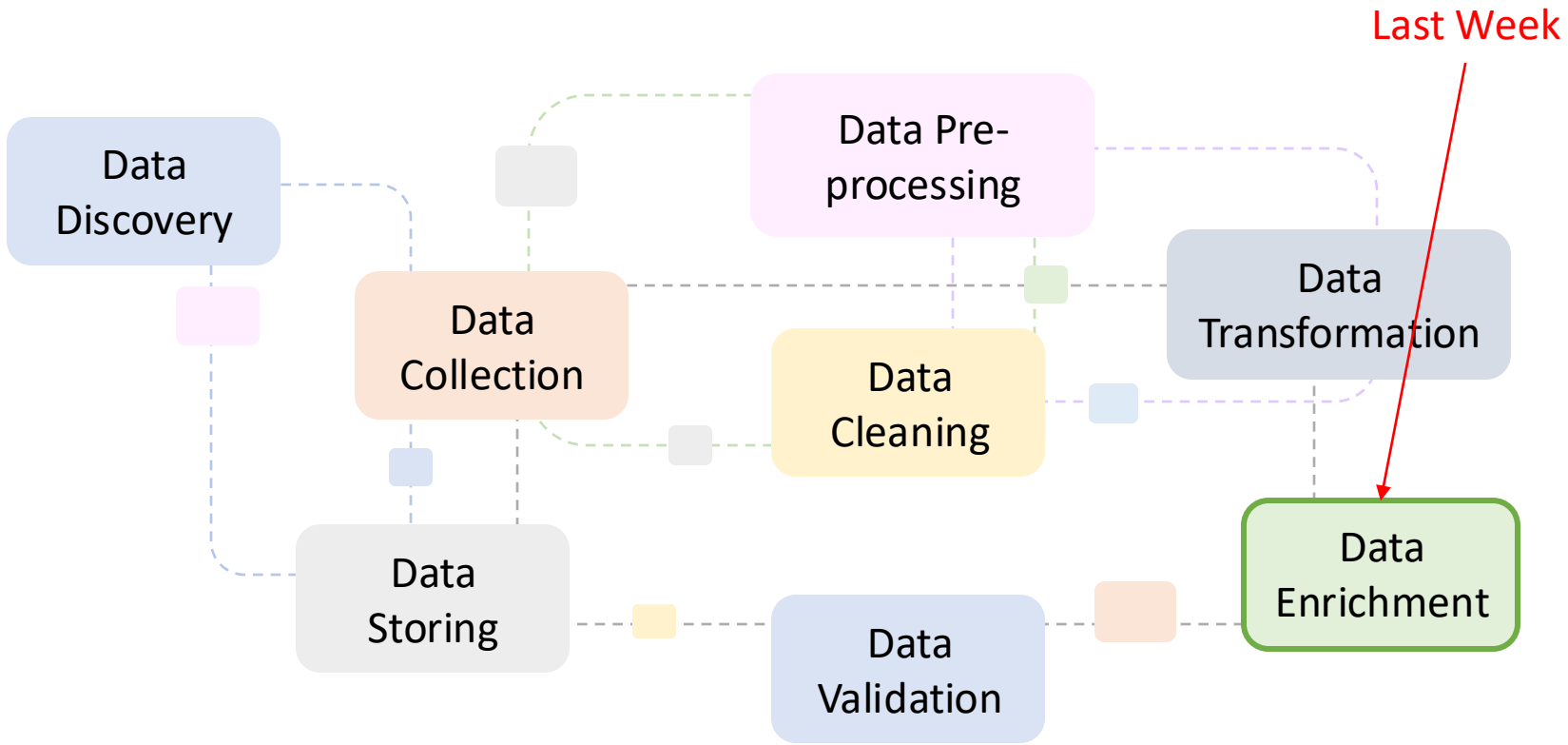
Monash University

Data Wrangling Tasks (Recap)

In the **Data Pre-processing** stage, preliminary data preparation tasks are performed to make raw data more suitable for analysis.

Data enrichment refers to the process of enhancing existing data by appending additional context or information from external sources.

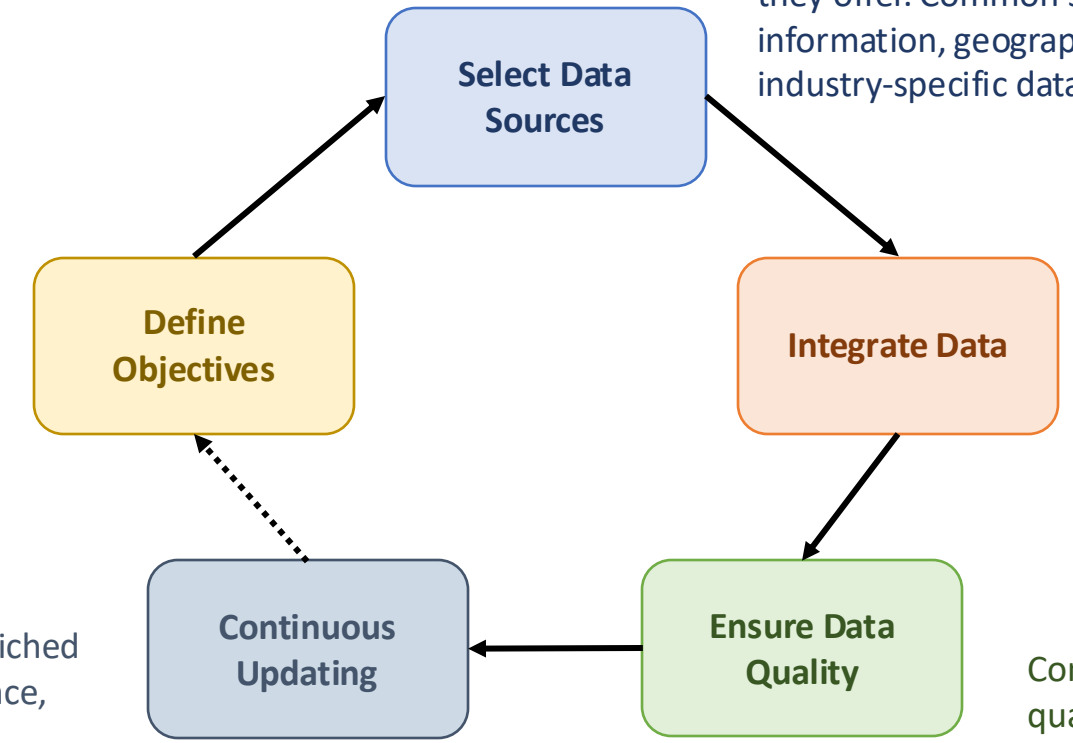
Data integration is a crucial component of the data wrangling process, which involves combining data from different sources to create a unified view.



Steps of Data Enrichment

Choose appropriate external sources based on the reliability, accuracy, and relevance of the data they offer. Common sources include demographic information, geographic data, social media data, industry-specific databases, and more.

Determine what specific information is missing from your current dataset and what you need to enhance its value for particular uses, such as targeted marketing, customer relationship management, or advanced analytics.

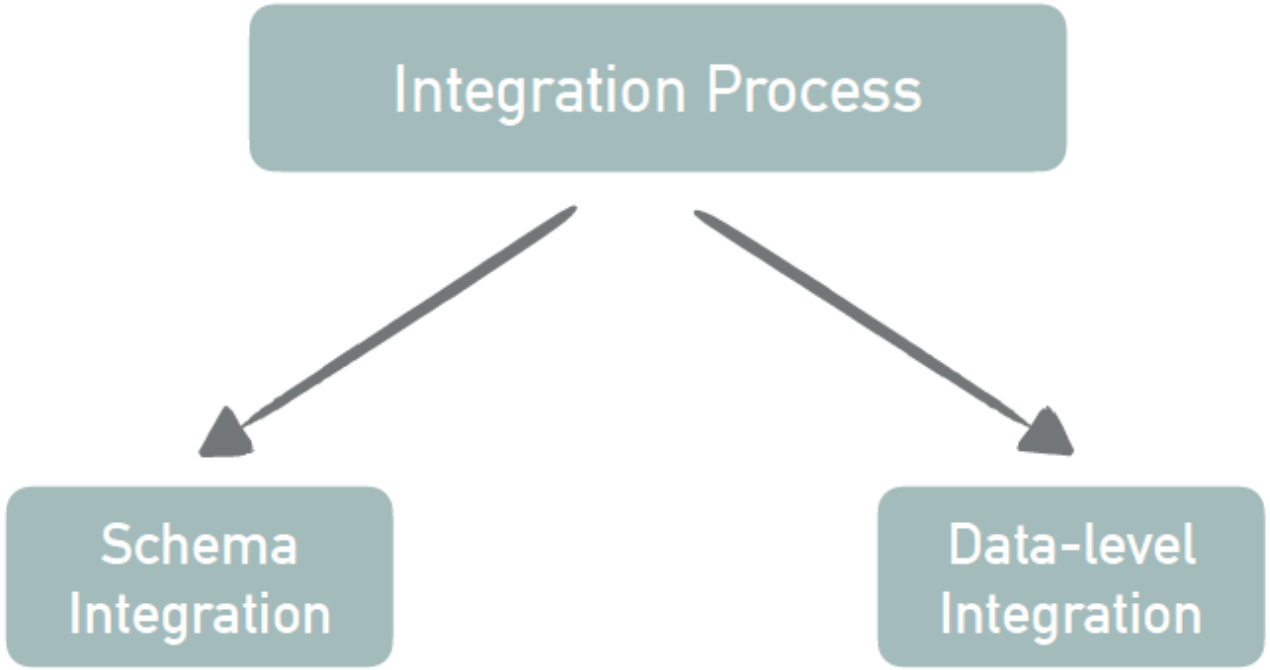


Merge the external data with your existing dataset. This might involve complex ETL (Extract, Transform, Load) processes, especially if the data structures differ significantly.

Periodically update the enriched data to maintain its relevance, especially for dynamically changing datasets like consumer behaviour or market trends.

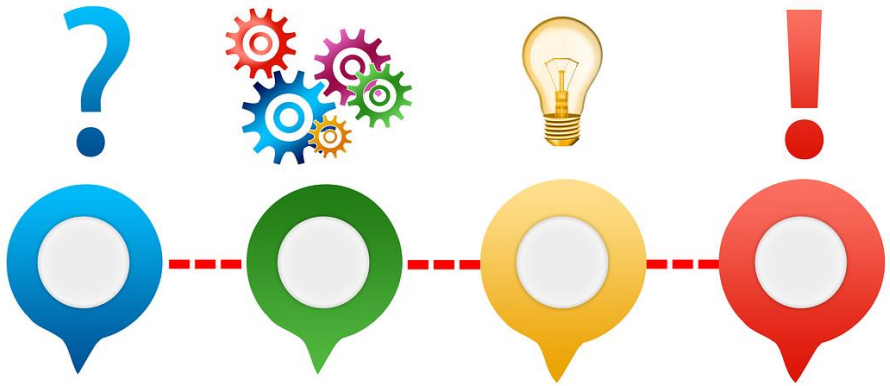
Conduct thorough checks to verify the quality of the enriched data. This includes validating the accuracy, completeness, and timeliness of the data.

Data Integration Category



Data Validation

- Definition of Data Validation
- Three types of Data Validation
  - Structural Validation
  - Content Validation
  - Logical Validation
- Error Handling



MONASH University

Data Validation

- **Data validation** is a critical process in data management and analysis that ensures the accuracy, completeness, and reliability of data before it is used for decision-making, analysis, or reporting.
- The **goal** of data validation is to check and verify that the data meets specific criteria and standards set for a particular purpose.
- Data validation is essential for:
  - **Maintaining data quality**: Ensuring data is clean, correct, and useful.
  - **Preventing errors**: Reducing the risks associated with data-driven decisions that could be based on faulty, incomplete, or inaccurate data.
  - **Improving decision-making**: High-quality data leads to more accurate and reliable business insights and decisions.

Types of Data Validation

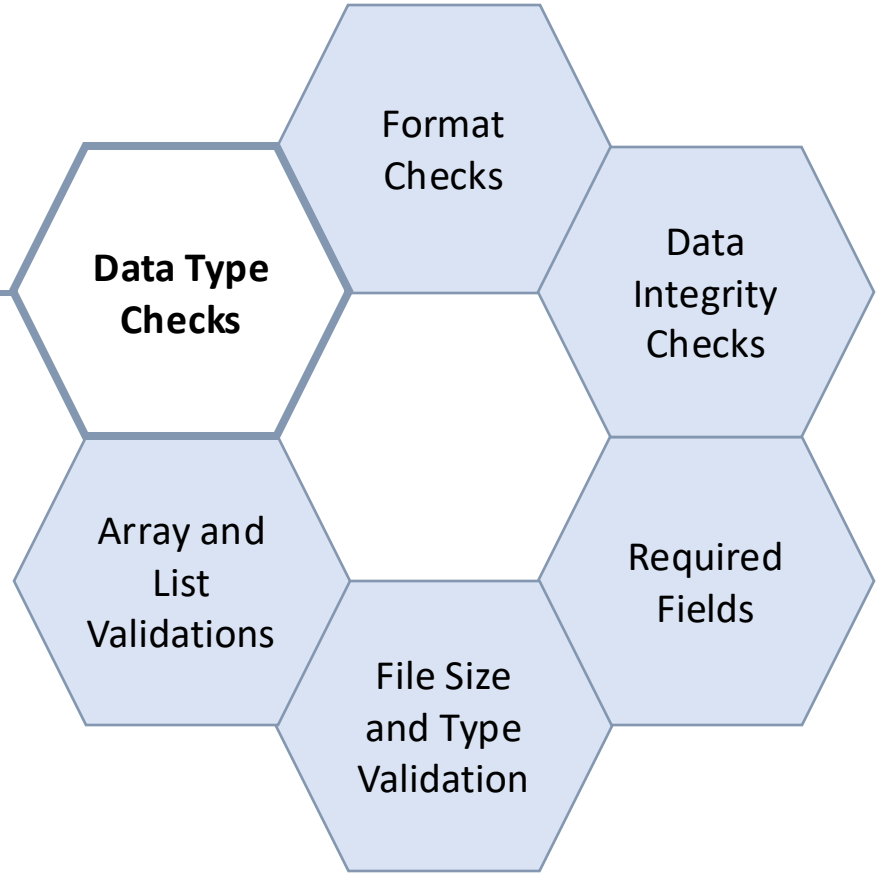
- Data validation can be categorized into various types, each addressing different aspects of data quality:
  - **Structural Validation**: Ensures the data adheres to the specified schema or model. This includes checks for data type, format, and size.
  - **Content Validation**: Focuses on the accuracy and relevance of the data content. This includes range checks, referential integrity checks, and cross-field validation.
  - **Logical Validation**: Involves checking the data against business rules and logic to ensure it makes sense in a given business context.

Structural Validation

- **Structural validation** checks that the data matches the specified structure, which can include the format, data types, and architecture of the data storage system.
- The **purpose** is to ensure that data is organized and formatted correctly, which is crucial for automated processing and reliable data integration.

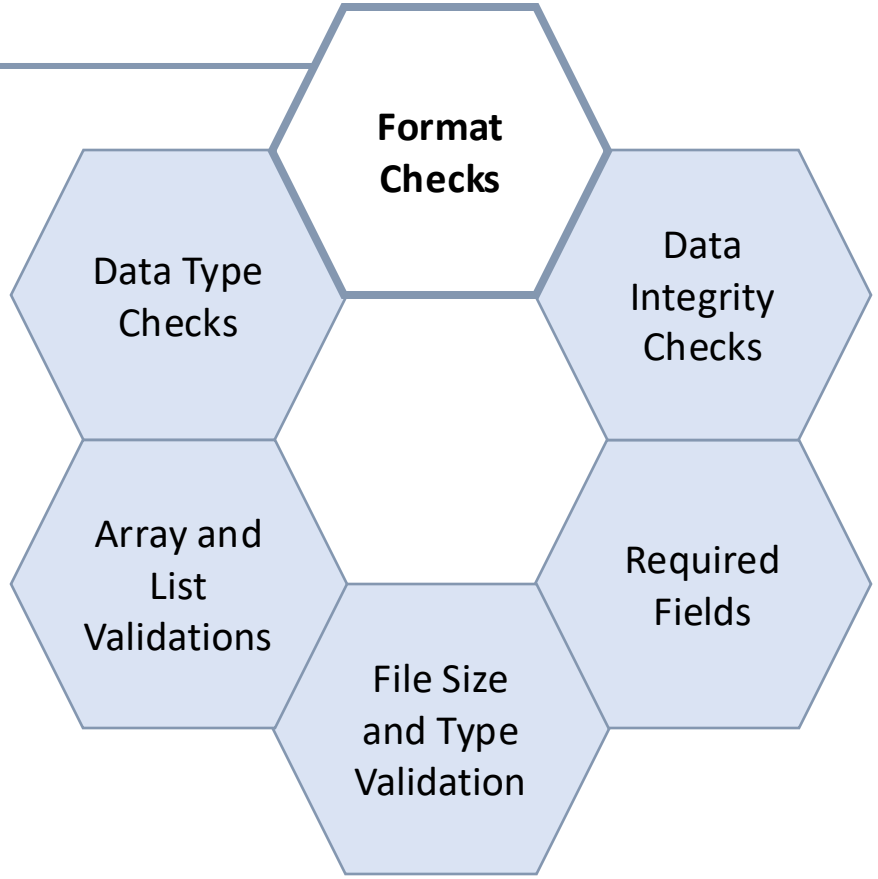
Tasks in Structural Validation

Ensuring that data fields contain the correct data types, such as integers, strings, dates, etc. For example, a date field should not accept alphabets or numbers that don't conform to date formats.

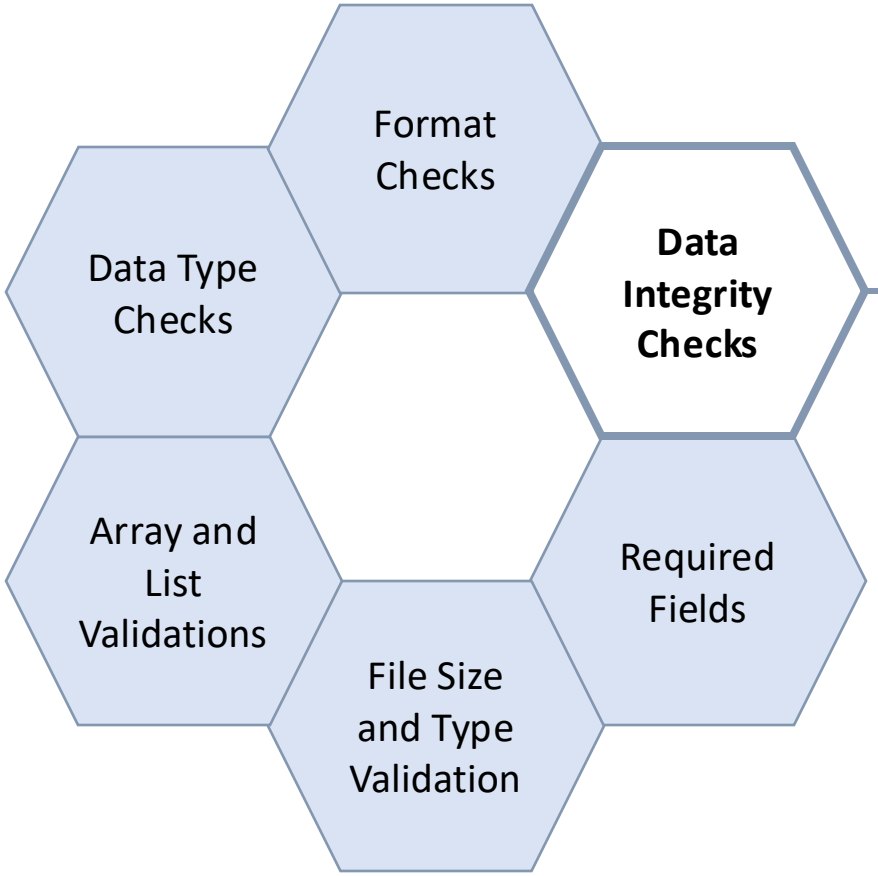


Tasks in Structural Validation

Verifying that data entries adhere to the specified format, such as telephone numbers, postal codes, email addresses, and other standardized formats.

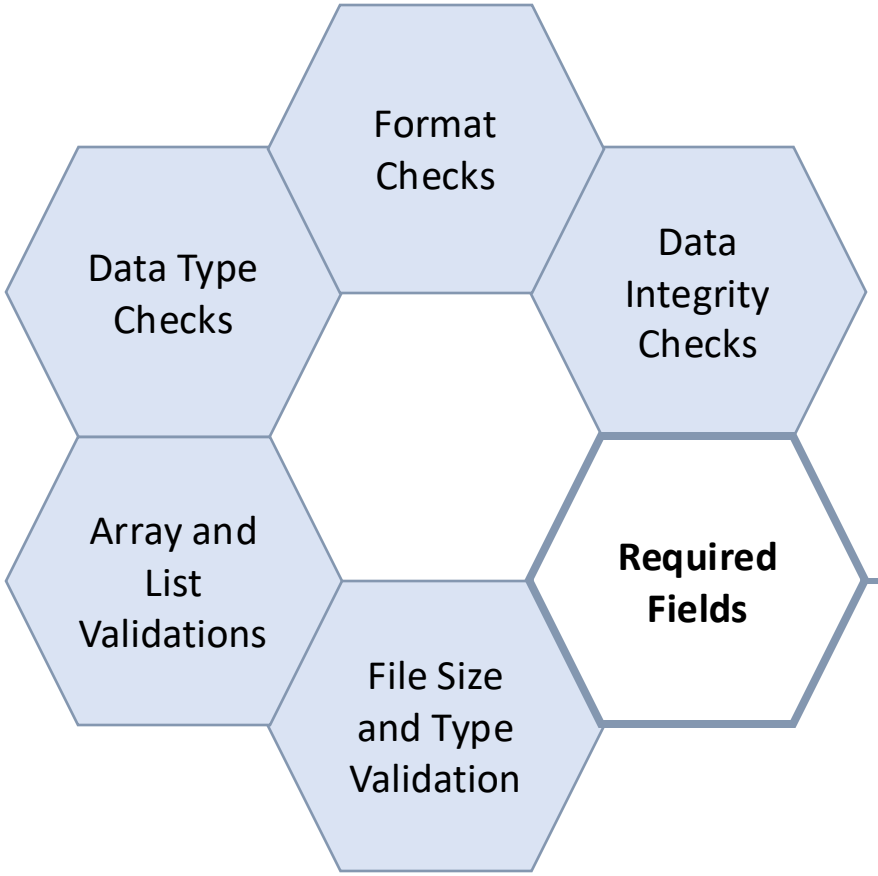


Tasks in Structural Validation



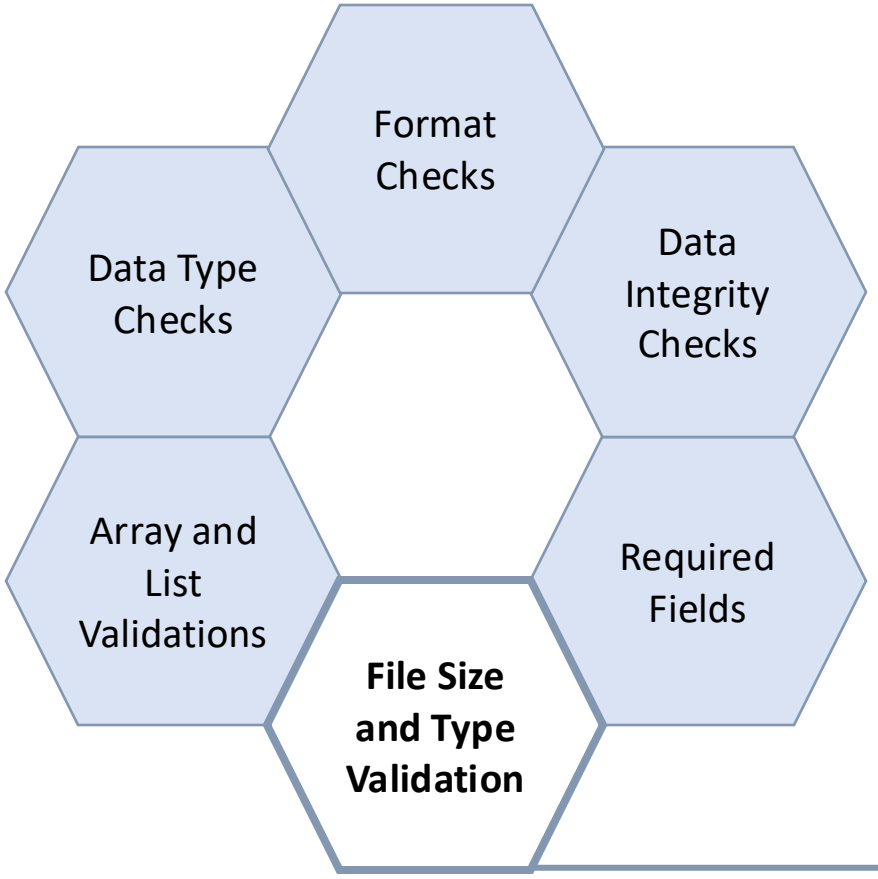
Ensuring that relationships among data fields are maintained, particularly in relational databases. This includes foreign key validations and checks for orphan records.

Tasks in Structural Validation



Checking for fields that must not be empty. This ensures that essential data is present for accurate processing.

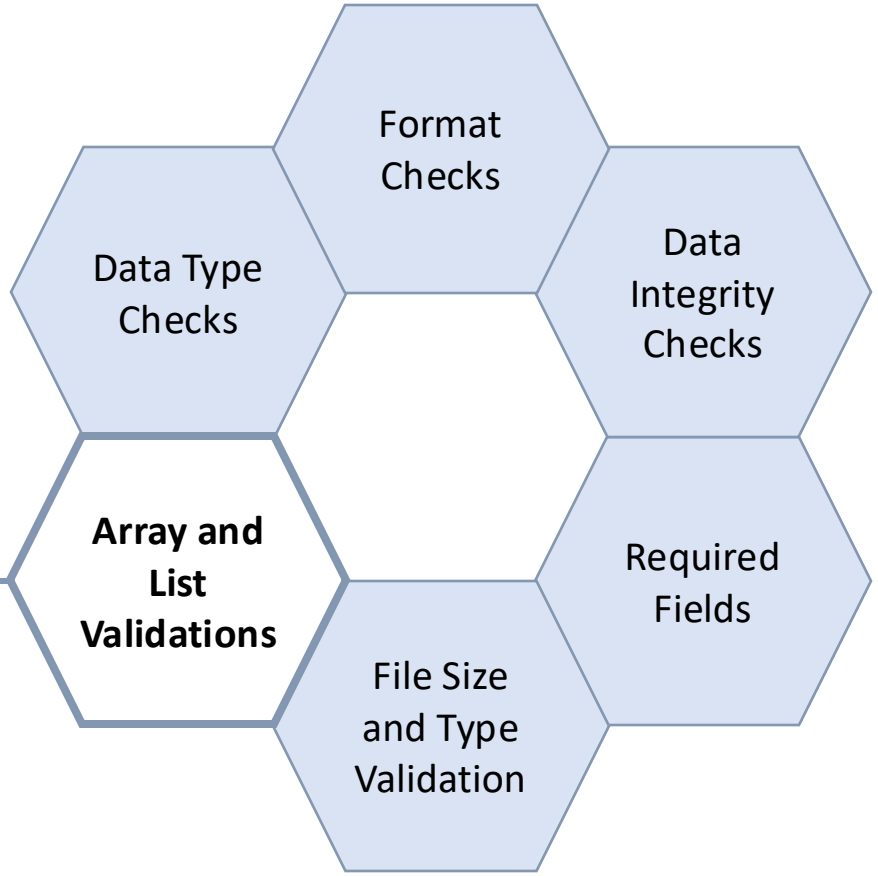
Tasks in Structural Validation



Ensuring that file uploads and data entries do not exceed the defined size limits and are of the expected file type, which is particularly relevant in data ingestion processes.

Tasks in Structural Validation

In data structures that involve arrays or lists, ensuring that the number of elements or the structure of these elements matches expected patterns.



Techniques and Tools for Structural Validation

- **Database Management Systems (DBMS)**
  - Most DBMSs inherently support structural validation through schema definitions, constraints, and data type specifications.
- **Data Validation Libraries**
  - Programming languages like Python and R offer libraries (e.g., Pandas, Pydantic in Python) that can perform structural checks on data as it is processed.
- **ETL (Extract, Transform, Load) Tools**
  - Many ETL tools include features to enforce data structure rules during the data transformation and loading phases.
- **Custom Scripts and Procedures**
  - In some cases, custom scripts are written to check for adherence to more complex or business-specific structures.

Challenges in Structural Validation

- **Scalability**
  - As data volumes grow, ensuring all data meets structural validation rules can become computationally intensive.
- **Evolving Schemas**
  - In dynamic environments where data schemas evolve, keeping validation rules updated can be challenging.
- **Integration of Diverse Data Sources**
  - Ensuring structural consistency across data from multiple sources requires robust integration and validation strategies.

Impact of Structural Validation

Effective structural validation is fundamental for:

- **Data Quality Assurance**
  - It prevents the propagation of structurally incorrect data through the analytics pipeline, ensuring that subsequent analyses are based on sound data.
- **Operational Efficiency**
  - By automating the early detection of data issues, it reduces the need for manual corrections, saving time and resources.
- **Enhanced Data Integration**
  - Structurally validated data integrates more seamlessly with existing systems and datasets, reducing errors related to data merging and transformation

An example – Structural Validation

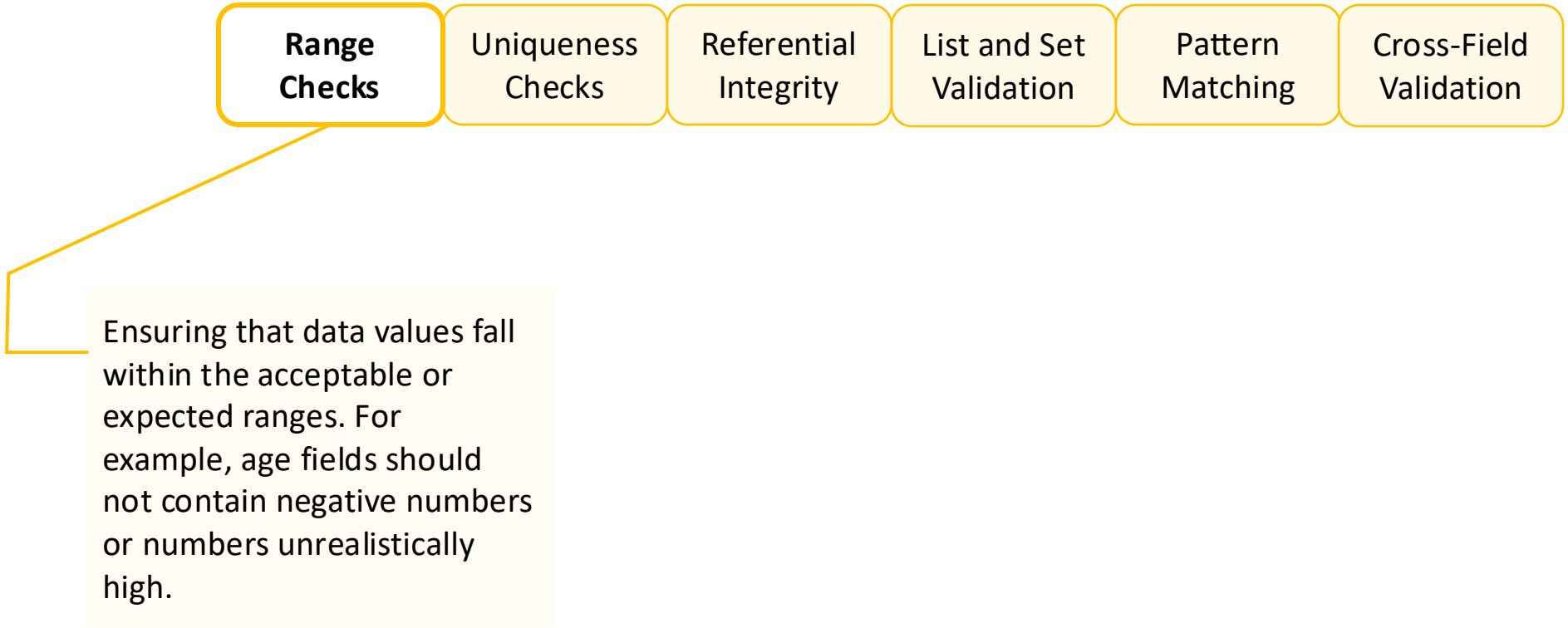
Suppose we have a dataset of online retail transactions from an e-commerce store. The dataset looks like this:

Customer_ID	Date_of_Purchase	Order_Amount	Product_ID	Customer_Email
123	2023-08-15	59.99	P001	john.doe@example.com
124	15-08-2023	120.50	P002	jane.smith@example.com
125	2023-08-16	-75.00	P003	adam_1@wrongformat
126	not available	99.99	NULL	eve.williams@example.com
126	2023-08-16	200.00	P004	eve.williams@example.com
	2023-08-17	49.50	P005	mark.adams@example.com

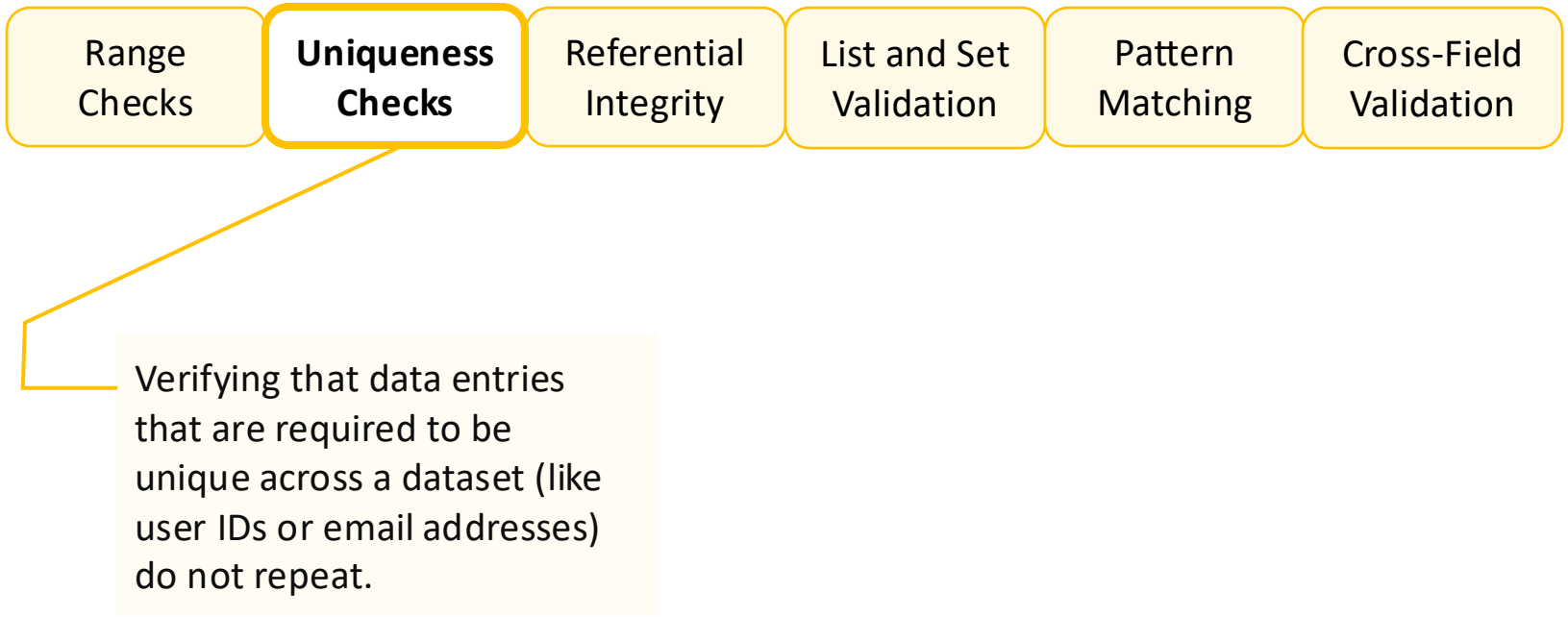
Content Validation

- **Content validation** focuses on verifying the accuracy, relevance, and quality of the data content itself.
- This process involves checking the actual data entries against predefined rules and criteria to ensure they are correct and appropriate for their intended use.
- Content validation ensures that data is not only structurally correct but also semantically accurate.
- It involves checks that data is logically correct and appropriate for the context in which it will be used.
- The main **purpose** is to avoid logical errors that could mislead data analysis or decision-making processes.

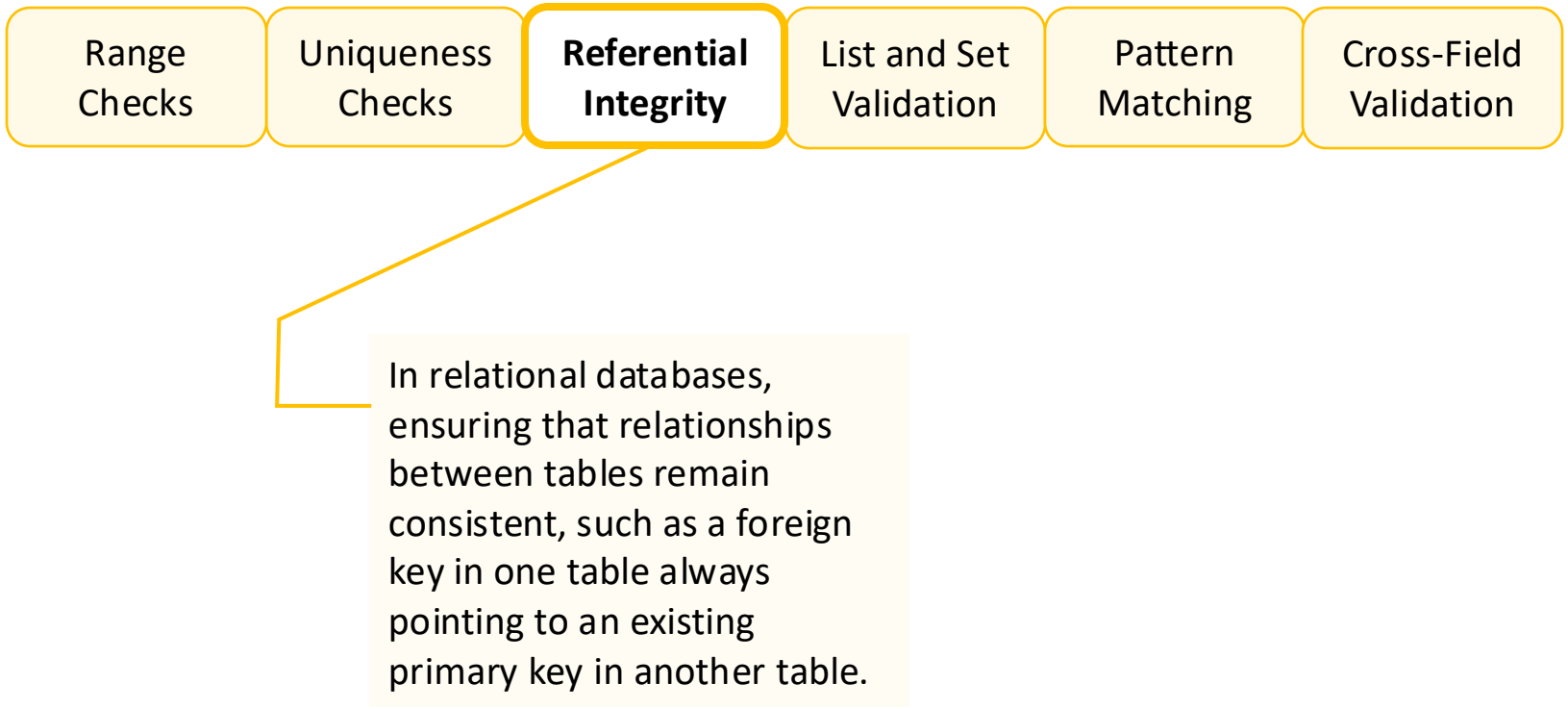
Tasks in Content Validation



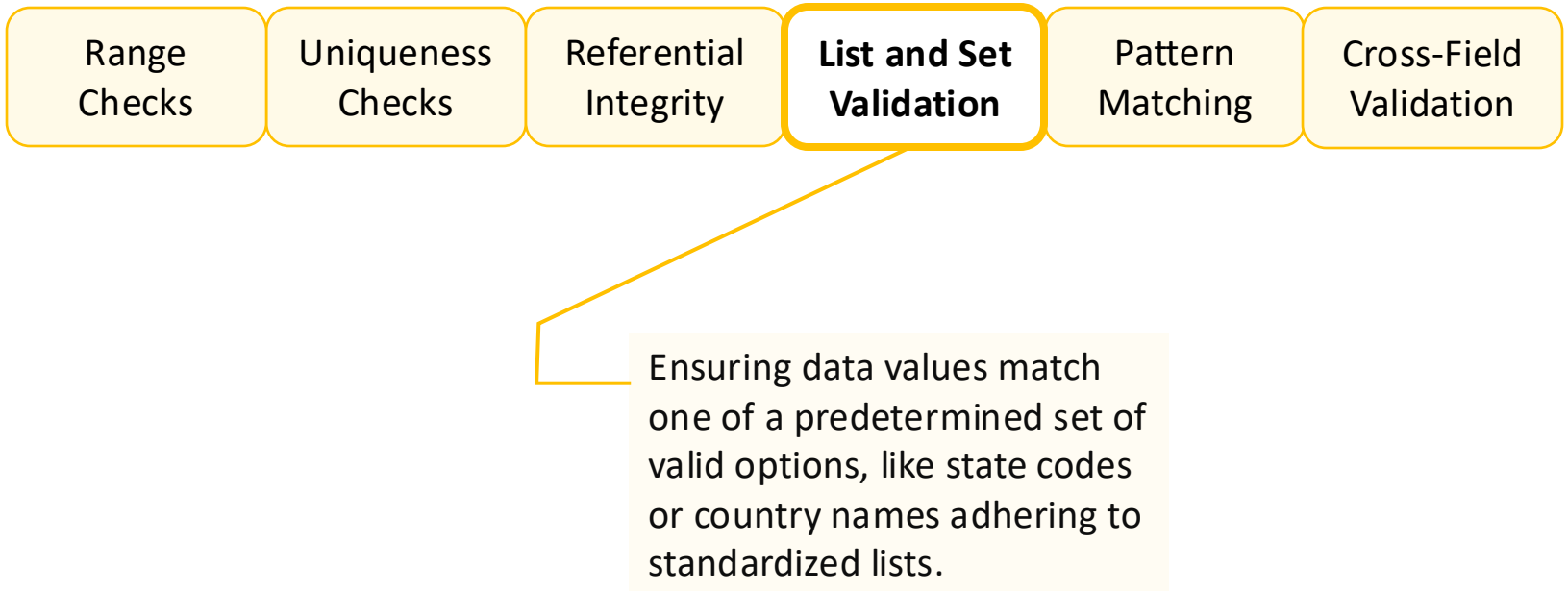
Tasks in Content Validation



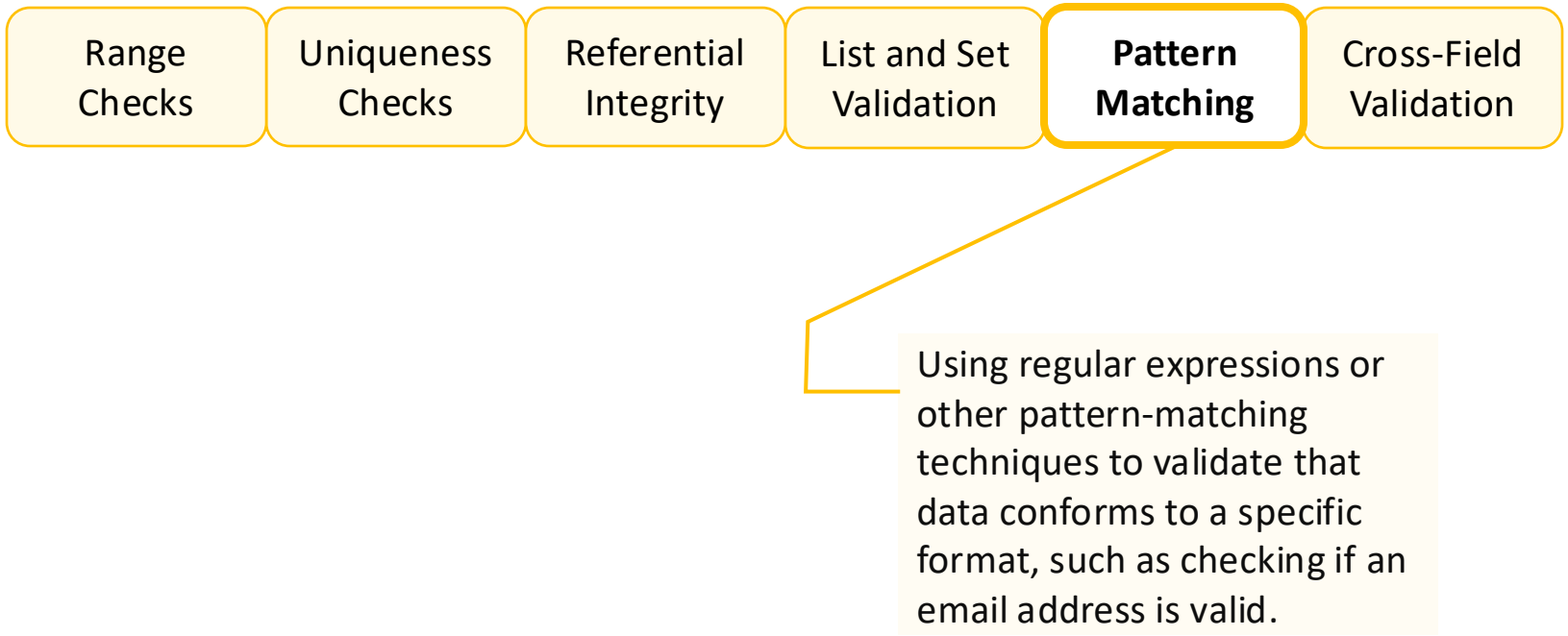
Tasks in Content Validation



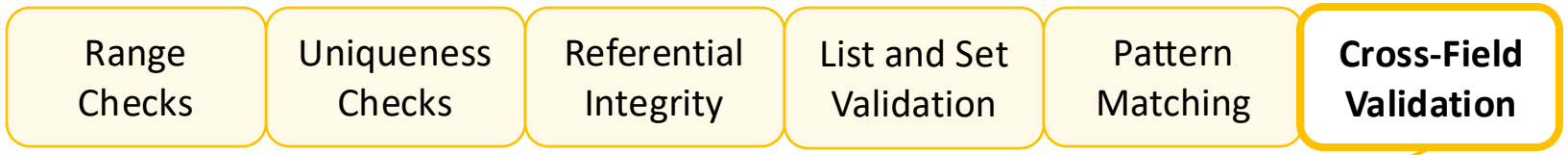
Tasks in Content Validation



Tasks in Content Validation



Tasks in Content Validation



Applying rules that depend on multiple fields to ensure consistency. For example, ensuring a delivery date is always later than the order date in a transaction record.

Techniques and Tools for Content Validation

- **Programming and Scripting**
  - Utilizing custom scripts in languages like Python, JavaScript, or SQL to automate content checks based on complex logic.
- **Data Validation Libraries**
  - Libraries such as Python's Pandas for data manipulation include functions that can facilitate content validation directly within data processing workflows.
- **ETL Tools**
  - Many ETL platforms come with built-in capabilities to apply content validation rules during data transformation and loading stages.
- **Data Quality Software**
  - Dedicated data quality platforms often feature comprehensive tools for validating content against a wide array of criteria and managing data quality across systems.

Challenges in Content Validation

- **Complexity of Business Rules**
  - Content validation can get complex when business rules that dictate data validity are intricate or change frequently.
- **High Volumes of Data**
  - Ensuring every data item is validated in large datasets can be time-consuming and resource-intensive.
- **Dynamic Data Sources**
  - When data comes from various sources that may change over time, maintaining consistent standards for content validation can be challenging.

Impact of Content Validation

Effective content validation is critical for:

- **Maintaining Data Integrity**: Ensures that data in the system is reliable and can be trusted for making business decisions.
- **Preventing Errors**: Reduces the risk of errors that could arise from incorrect data being used in operations, analysis, or customer interactions.
- **Enhancing User Trust**: Reliable data increases confidence among users and stakeholders, which is vital for operational transparency and business reputation.

Example (cont.) – Content Validation

Suppose we have a dataset of online retail transactions from an e-commerce store. The dataset looks like this:

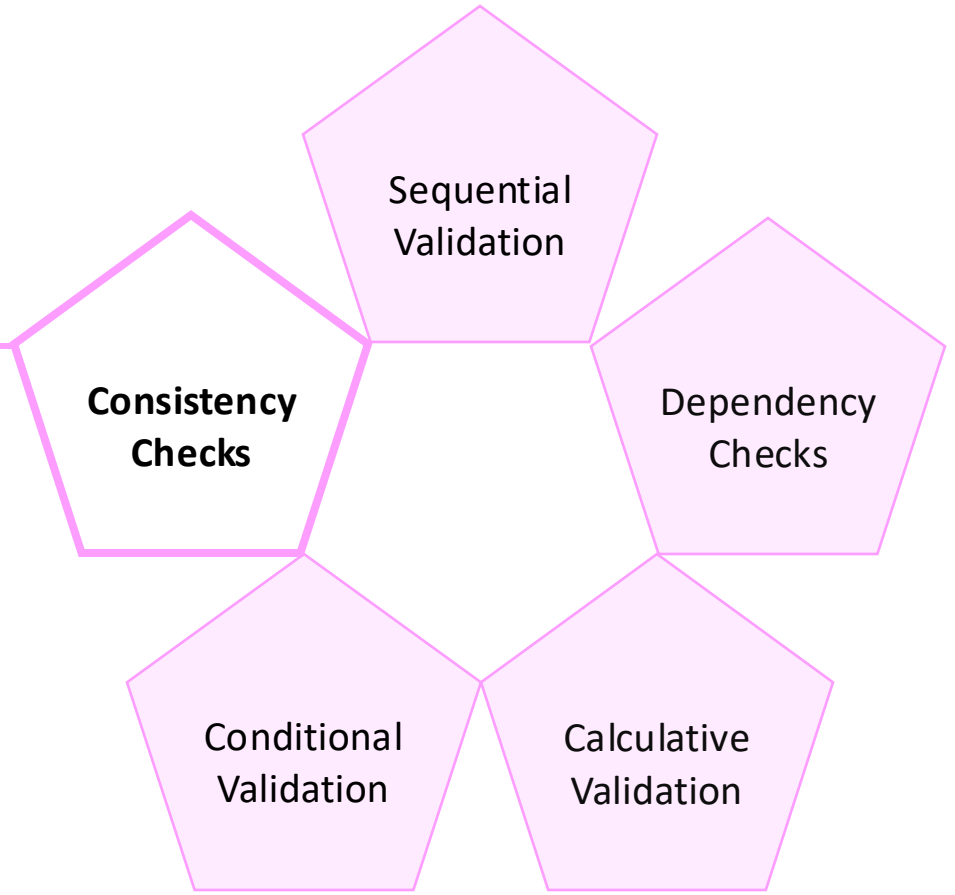
Customer_ID	Date_of_Purchase	Order_Amount	Product_ID	Customer_Email
123	2023-08-15	59.99	P001	john.doe@example.com
124	15-08-2023	120.50	P002	jane.smith@example.com
125	2023-08-16	-75.00	P003	adam_1@wrongformat
126	not available	99.99	NULL	eve.williams@example.com
126	2023-08-16	200.00	P004	eve.williams@example.com
	2023-08-17	49.50	P005	mark.adams@example.com

Logical Validation

- **Logical validation** is an advanced component of data validation that focuses on ensuring data adheres to defined business rules and logic.
- This process involves verifying the contextual correctness of data based on its relationship and relevance to other data within the dataset.
- Logical validation is crucial for maintaining the integrity and usefulness of data in complex business environments.

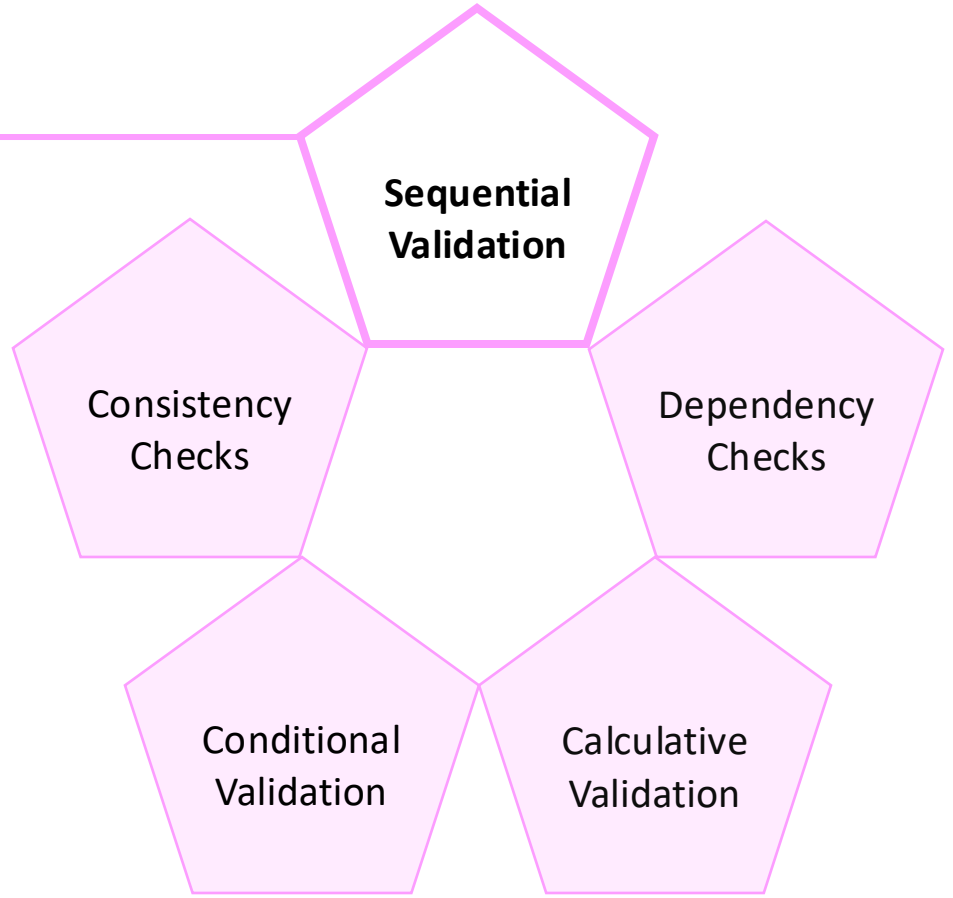
Tasks in Logical Validation

Verifying that data across different fields or records is consistent according to business rules. For instance, ensuring that a patient's medical treatment is appropriate for their diagnosed condition.

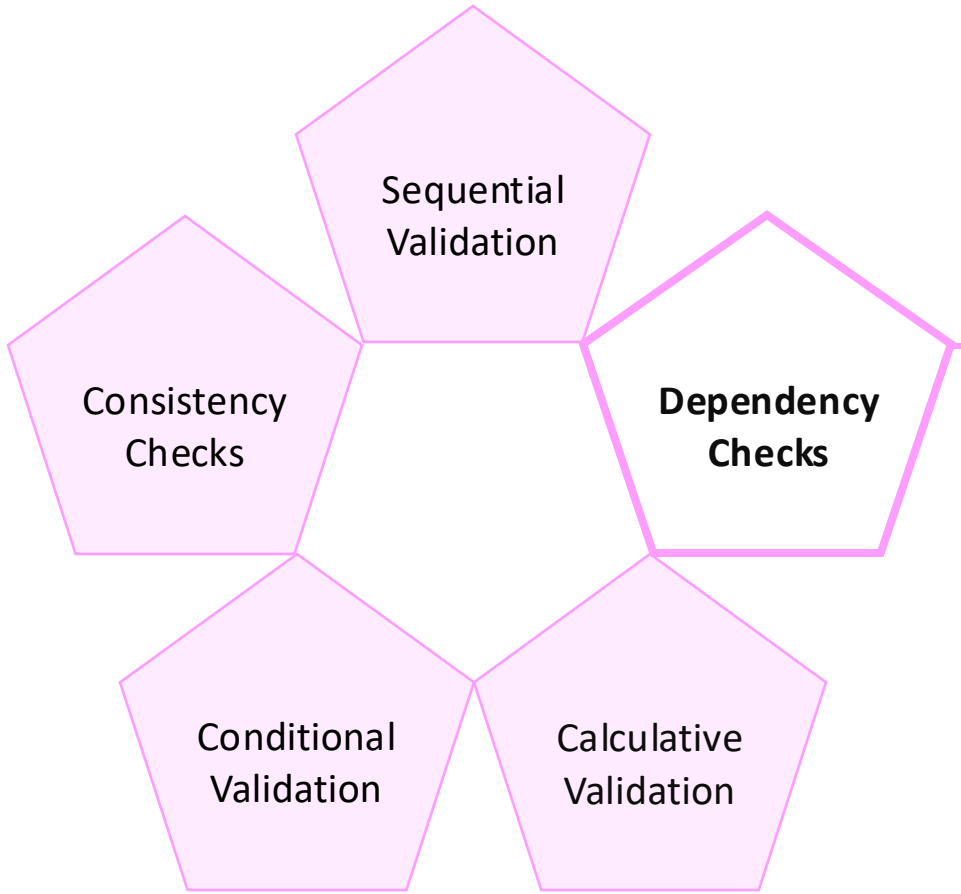


Tasks in Logical Validation

Ensuring data follows logical sequences or workflows, such as verifying that a product's shipping date follows its order date.

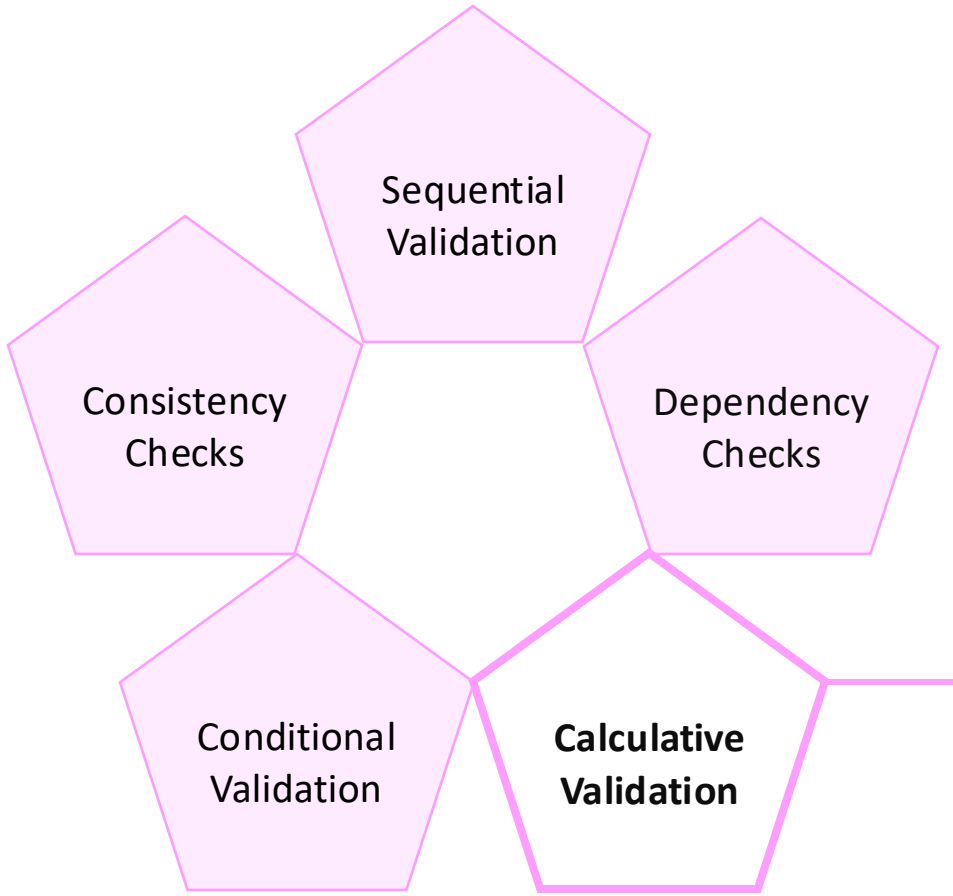


Tasks in Logical Validation



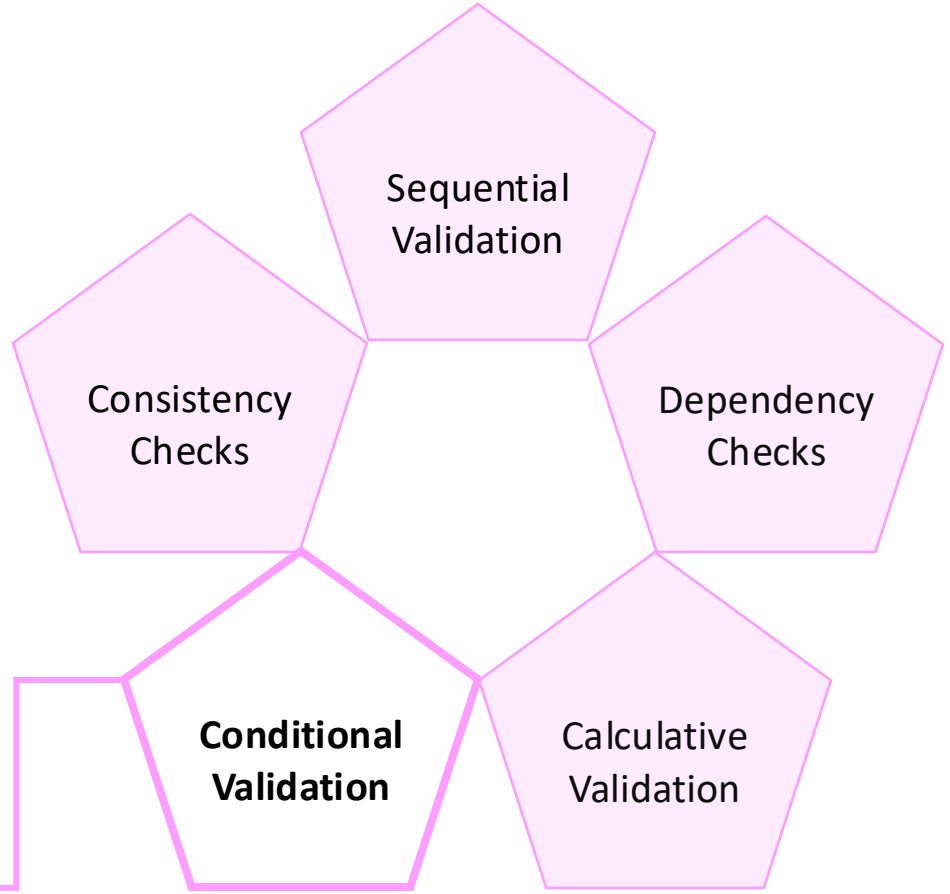
Validating that data relationships are maintained, such as ensuring a manager is always senior to the employees they supervise.

Tasks in Logical Validation



Checking that computed fields (e.g., total price, tax calculations) are correctly derived from their underlying data points.

Tasks in Logical Validation



Applying rules that are triggered only under specific conditions, such as special discounts that apply only if a purchase meets certain criteria.

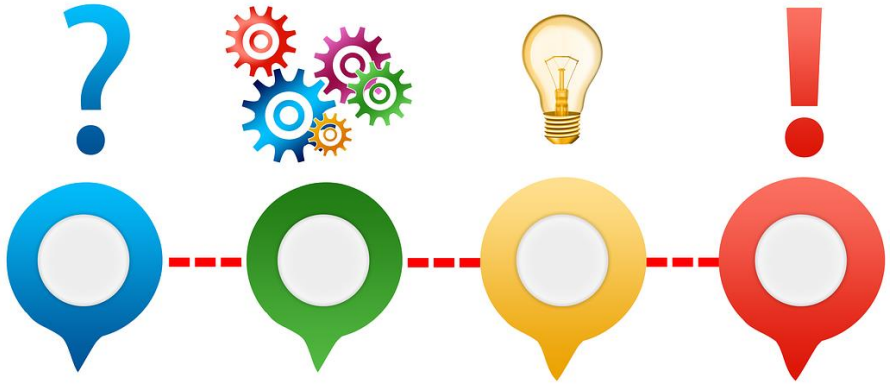
An Example – Logical Validation

Let's consider a dataset for **employee timesheets**. The dataset records employees' work hours per day, their hourly pay rate, and their department. The table looks like this:

Employee_ID	Date	Hours_Worked	Hourly_Rate	Department
1001	2023-09-01	8	25	HR
1002	2023-09-01	9	30	IT
1003	2023-09-01	15	22	Finance
1004	2023-09-01	24	28	Operations
1005	2023-09-01	7	45	HR
1006	2023-09-01	12	120	IT

Data Validation

- Definition of Data Validation
- Three types of Data Validation
  - Structural Validation
  - Content Validation
  - Logical Validation
- **Error Handling**

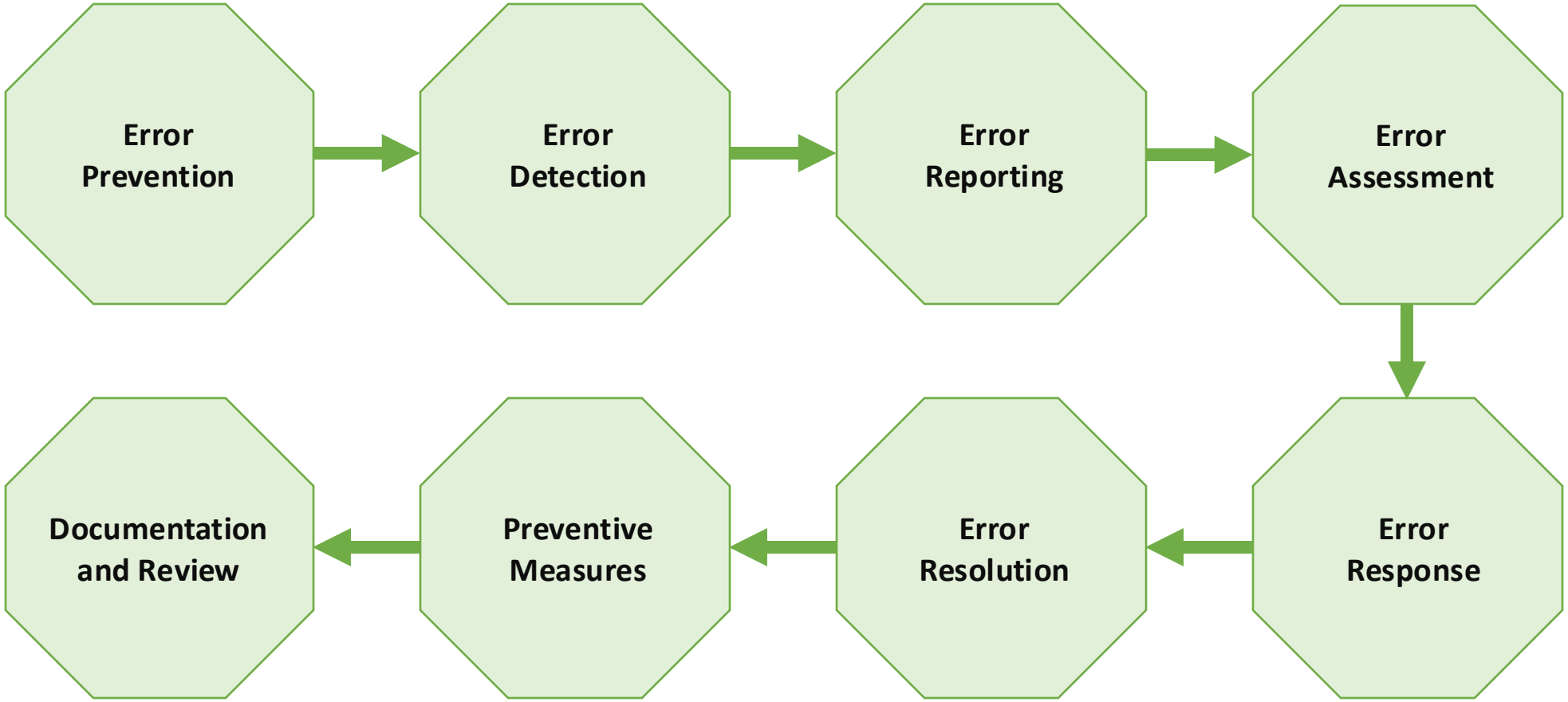


MONASH University

Error Handling

- **Error handling** involves strategies to manage and resolve errors that occur during data entry, processing, or analysis.
- Effective error handling not only addresses errors once they occur but also helps to prevent them from happening again.

General Procedure of Error Handling



Error Handling Strategies

- When implementing error handling strategies, it's important to consider various factors that ensure the processes are effective, efficient, and aligned with the overall goals of the system or application.
  - Comprehensive Error Detection
  - Clear Error Reporting
  - Prioritization of Errors
  - Consistent Error Handling
  - User Experience Considerations
  - Root Cause Analysis
  - Error Resolution
  - Continuous Improvement
  - Security Considerations
  - Compliance and Legal Requirements

Summary & To-do List

- Please download and read materials provided on Moodle.
- Review content learnt from Week 11.

- Assessments
  - Read the tasks in Assessment 2 and continue working on it.
  - Assessment 2 due on Monday, Week 12.

- Next week: Advanced Data Wrangling