

ATME COLLEGE OF ENGINEERING

13th KM Stone, Bannur Road, Mysore - 560 028



A T M E

College of Engineering

DEPARTMENT OF COMPUTER SCIENCE AND DESIGN



LECTURE NOTES

**COURSE: ROBOTIC PROCESS AUTOMATION DESIGN &
DEVELOPMENT (UiPath)**

COURSE CODE: BCG701

SEMESTER: VII

COURSE COORDINATOR: Mr. YOGESH N

(ACADEMIC YEAR 2026-27)

INSTITUTIONAL MISSION AND VISION

Objectives

- To provide quality education and groom top-notch professionals, entrepreneurs and leaders for different fields of engineering, technology and management.
- To open a Training-R & D-Design-Consultancy cell in each department, gradually introduce doctoral and postdoctoral programs, encourage basic & applied research in areas of social relevance, and develop the institute as a center of excellence.
- To develop academic, professional and financial alliances with the industry as well as the academia at national and transnational levels.
- To develop academic, professional and financial alliances with the industry as well as the academia at national and transnational levels.
- To cultivate strong community relationships and involve the students and the staff in local community service.
- To constantly enhance the value of the educational inputs with the participation of students, faculty, parents and industry.

Vision

Development of academically excellent, culturally vibrant, socially responsible and globally competent human resources.

Mission

- To keep pace with advancements in knowledge and make the students competitive and capable at the global level.
- To create an environment for the students to acquire the right physical, intellectual, emotional and moral foundations and shine as torch bearers of tomorrow's society.
- To strive to attain ever-higher benchmarks of educational excellence.

Department of Computer Science & Design

Vision of the Department

- To develop highly talented individuals in Computer Science and Engineering to deal with real world challenges in industry, education, research and society.

Mission of the Department

- To inculcate professional behaviour, strong ethical values, innovative research capabilities and leadership abilities in the young minds & to provide a teaching environment that emphasizes depth, originality and critical thinking.
- Motivate students to put their thoughts and ideas adoptable by industry or to pursue higher studies leading to research

PROGRAM EDUCATIONAL OBJECTIVES (PEO'S):

1. Empower students with a strong basis in the mathematical, scientific and engineering fundamentals to solve computational problems and to prepare them for employment, higher learning and R&D.
2. Gain technical knowledge, skills and awareness of current technologies of computer science engineering and to develop an ability to design and provide novel engineering solutions for software/hardware problems through entrepreneurial skills.
3. Exposure to emerging technologies and work in teams on interdisciplinary projects with effective communication skills and leadership qualities.
4. Ability to function ethically and responsibly in a rapidly changing environment by applying innovative ideas in the latest technology, to become effective professionals in Computer Science to bear a life-long career in related areas.

Program Specific Outcomes (PSOs)

PSO1: Ability to apply skills in the field of algorithms, database design, web design, cloud computing and data analytics.

PSO2: Apply knowledge in the field of computer networks for building network and internet based applications.

Module 4: Excel Automation

This module presents the core concepts, activities, configurations, and exercises related to Excel Automation using UiPath's StudioX platform. The course material centres on automating Excel workbooks through various activities to streamline data management and reporting, specifically within the context of an Employee Onboarding scenario.

Sample Overview: Employee Onboarding Scenario

- The automation exercises are based on managing employee onboarding data for a small firm's HR team, which has hired 20 new employees.
- **EmployeeOnboarding.xlsx:** The main workbook containing employee details (start dates, salaries, training requirements).
- **EmployeeOnboardingInput.xlsx:** Provides data for an additional five new employees that need to be appended to the main workbook.
- **EmployeeOnboardingFinal.xlsm:** The final formatted workbook with macros, ready for delivery after automation processes are applied.
- **Purpose:** Centralise employee data management and automate reporting for HR onboarding activities.

Excel Automation Activities Overview

The automation activities are grouped under the **Excel category** in UiPath StudioX. Each activity facilitates interaction with Excel workbooks and sheets to perform tasks such as:

- Creating or opening files
- Manipulating sheets (insert, rename, duplicate, delete)
- Handling rows and columns
- Managing data ranges and tables
- Formatting and refreshing Pivot Tables
- Exporting and saving files in Excel or PDF formats
- Running macros

Key Activities and Their Configurations

Use Excel File Activity

- Acts as a **parent container** for Excel automation activities.
- Allows opening or creating Excel workbooks to perform automation on them.

Configurations:

- Supports adding child activities such as sorting, copying, filtering, etc.

Sheet Manipulation Activities

These activities allow modification of sheets within the workbook.

Exercises and Practical Implementation

- Exercises accompany each configuration explanation, where students perform:
- Opening the EmployeeOnboarding workbook
- Inserting new sheets
- Renaming sheets for clarity
- Duplicating sheets to create backups or templates
- Deleting redundant or test sheets
- These exercises replicate real-world use cases to familiarise students with UiPath StudioX Excel activities.

For Each Excel Sheet

Purpose: To execute actions on each worksheet within an Excel workbook.

Usage: Requires configuration of at least one child activity to perform operations on individual sheets.

Key Configurations:

- For each: Defines the variable name representing the current sheet being iterated (default: \$CurrentSheet\$).
- In workbook: Identifies the workbook containing the sheets; auto-populated if within a parent Use Excel File activity.
- Example Outcome: Names of all sheets in a workbook (e.g., EmployeeOnboarding.xlsx) can be outputted sequentially to the console or output panel.

Insert Column

Purpose: Adds a new column within a specified range, sheet, or table.

Configurations:

- Exercise Focus: Practicing the configuration and execution of inserting columns with appropriate headers and formats.

Text To Columns

Purpose: Splits text from a single cell, range, or table into multiple columns based on delimiters.

Configurations:

- Example: Splitting a cell with contents like "notebooks,, clipboards" into separate columns by treating ",," as a single delimiter.

Delete Column

Purpose: Deletes specific column(s) from a sheet, range, or table.

Configurations:

- Selection method: Column name can be selected by Excel header or dynamically specified through Text Builder.

Insert Rows

Purpose: Inserts one or more rows into a defined range, sheet, or table.

Configurations:

- Example Usage: Inserting rows after headers or at specific indexes as required in workflow processes.

Delete Rows

Purpose: Deletes specified rows from a sheet, range, or table.

Configurations:

- Example Scenario: Deleting all hidden rows in a filtered table.

Find First/Last Data Row

Purpose: Identify the first and last row numbers containing data in a specified sheet, range, or table.

Key Configurations:

- Source: Required; sheet, range, or table to search.
- Has headers: Optional; excludes header row from count (default checked).
- Column name: Specifies the column to search.
- Save first/last row number: Stores the found row indices for later use.
- Blank rows to skip: Number of consecutive blank rows before stopping (default 1).
- First/Last row offset: Adjusts row indices by adding/subtracting rows for custom range.

For Each Excel Row

Purpose: Perform iterative actions on each row within a sheet, range, or table.

Key Configurations:

- For each: Reference name for the current row during iteration (default `CurrentRow`).
- In range: Required; data source to iterate through.
- Has headers: Indicates presence of header row for easier column referencing.
- Save after each row: Option to save Excel file post each row operation.
- EmptyRowBehavior: Defines handling of empty rows – *Stop* (default), *Skip*, or *Process*.

Write Cell

Purpose: Write static or dynamic values/formulas into specific Excel cells.

Key Configurations:

- What to write: Required; text or formula to input.
- Where to write: Required; cell address.
- Auto increment row: If checked, increments row number for each iteration to write multiple cells.

Create Pivot Table

Purpose: Automate creation of pivot tables for data summarization and analysis.

Key Configurations:

- Source: Required; data range or named table for pivot source.
- New table name: Required; name for the created pivot table.
- Destination: Required; location where pivot table is created.
- Child Activity - Add Pivot Field:
 - Field: Column to add to pivot.
 - Is a: Defines field type (row, column, filter, value).
 - Function: For value fields, specifies aggregation function (sum, count, etc.).

Format as Table

Purpose: Convert a specified cell range into a formatted Excel table.

Key Configurations:

- Destination: Range to be formatted.
- Table name: Optional; manual naming or auto-assigned.
- Save new table name as: Optional; stores the assigned table name for reuse.
- Replace existing: Option to overwrite existing table with the same name (default enabled).

Change Pivot Data Source

Purpose: Modify the source range of an existing pivot table to update its data.

Key Configurations:

- Pivot table: Identify which pivot table to update.
- New source: Define the new data range or table to replace the old source.

Core Excel Automation Activities

Activity	Description	Key Configurations/Details
Save Excel File	Saves changes to the referenced Excel workbook after automation steps.	- File : Workbook to save (auto-filled from "Use Excel File" activity).
Save Excel File As	Saves the workbook as a different Excel file format (.xlsx, .xlsb, .xlsm, .xls).	- Workbook : Workbook to be saved. - Save as type : Choose new file format. - Save as file : New file name. - Replace existing : Optional overwrite.
Save Excel File As PDF	Converts and saves the Excel workbook or project notebook as a PDF file.	- Workbook : Source workbook. - Save as file : PDF file path. - Replace existing : Optional overwrite.
VLookup	Implements Excel's VLookup function to find specific data in a range/table/sheet.	- Value to lookup : Target search value. - In range : Data range. - Column Index : Column to return data from. - Exact match : Optional (default true). - Save to : Output cell for result.
Filter	Filters data within a column based on specified criteria, using Basic or Advanced filters.	- Source : Range/sheet/table to filter. - Column name : Header of the column to filter. - Configure Filter : Values or conditions to apply. - Clear existing filter : Optional.
Run Spreadsheet Macro	Executes a macro embedded within a macro-enabled Excel workbook (.xlsm).	- Source workbook : Workbook containing the macro. - Macro name : Macro to execute. - Output to : Optional output location. - Add Macro Argument : Pass arguments to macro.

Detailed Configuration Highlights

- Save Excel File requires specifying the workbook, usually inherited from the parent activity.
- Save Excel File As allows saving in multiple Excel formats; the option to overwrite existing files is enabled by default.
- Save Excel File As PDF generates a PDF version of the workbook with overwrite options.
- VLookup requires precise inputs: the search value, data range, column index for the return value, and exact match toggling.
- Filter Activity supports both simple exact value filters and complex conditional filters (e.g., greater than specific values), with options to clear filters.
- Run Macro Activity mandates a macro-enabled workbook and allows passing multiple input arguments and capturing outputs.

Practical Examples

- VLookup Use Case: Find an employee named “John Smith” in a table and retrieve his start date by specifying the lookup value, the data range, and the column index containing the start date.
- Filter Use Case: Filter a transaction table to show only rows with "Completed" status or apply advanced filters to show quantities greater than 100.
- Macro Execution: Run a specific macro within an Excel workbook, optionally passing arguments and capturing returned values.

Key Insights and Conclusions

- **Integrating Excel Automation in RPA is critical** for automating administrative and data-heavy processes in business workflows.
- **Use Excel File activity** is foundational, enabling smooth integration of UiPath StudioX with Excel.
- Sheet-level activities (insert, rename, duplicate, delete) allow dynamic manipulation of workbooks, which helps maintain clean, organised data structures.
- Using **reference names** for files and sheets enhances maintainability and scalability of automation workflows.
- Optional configurations such as **password protection handling** and **read-only mode** increase the flexibility and security of automation scripts.
- The Employee Onboarding scenario provides a **practical business context** to apply Excel automation, reinforcing learning with relevant use cases.
- Understanding these elementary activities is essential to building more **complex automations involving rows, columns, formulas, pivot tables, and exporting data**.