LECTURE 04. COLLECTIONS. PART II

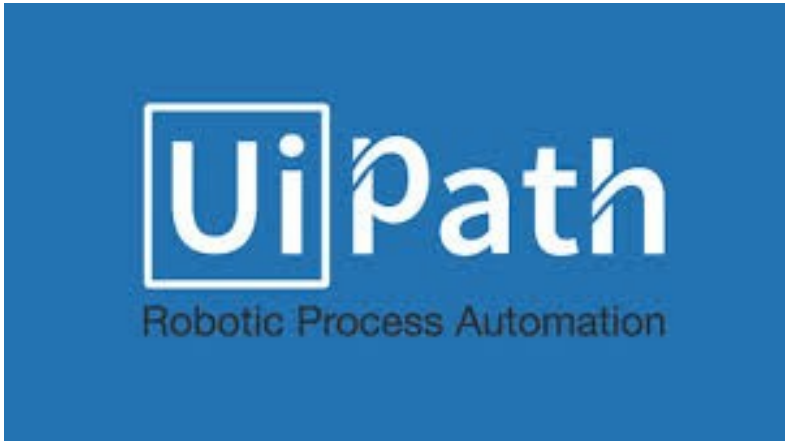
Robotic Process Automation
[24 October 2022]

Elective Course, 2022-2023, Fall Semester

Camelia Chisăliţă-Creţu, Lecturer PhD
Babeş-Bolyai University

Acknowledgements

This course is presented to our Faculty with the support of UiPath Romania.



Contents

Dictionary. Details

- **Dictionary** characteristics in UiPath:
  - **a dictionary has a flexible length;**
  - it implements the **IEnumerable** interface ==> can be iterated by using a **For Each** activity over the ***keys set***;
  - it is used to store:
    - multiple related pairs (key, value) that are passed as a **single argument** between workflows;
    - data in **Orchestrator queues**.

see **Demo6 - Dictionaries**

UiPath™

Dictionary. Declaration. Instantiation. Initialization

- ways to **declare/instantiate/initialize** a dictionary:
  - **Variables Panel:**
    - **Name:** bookDictionary; **Type:** Dictionary<String, String>;
    - **Default:** new Dictionary (of String, String) from{{"title", "Poems"}, {"author","M.Eminescu"}, {"publisher", "Litera"}}*//Count=3*
    - **Name:** gradeDictionary; **Type:** Dictionary<String, List<String>>;
    - **Default***:* new Dictionary(of Int32, List(of String)) from {{10, new List (of String) from {"Ana", "Anca"}}, {3, new List(of String) from {"me", "you", "her"}}} *//Count=2*
  - **Assign** activity:
    - monthDictionary = new Dictionary (of Int32, List(of String)) *//Count=0*
    - sDictionary = new Dictionary(of String, String) *//Count=0, pairs are added later*

see **Demo6 - Dictionaries**

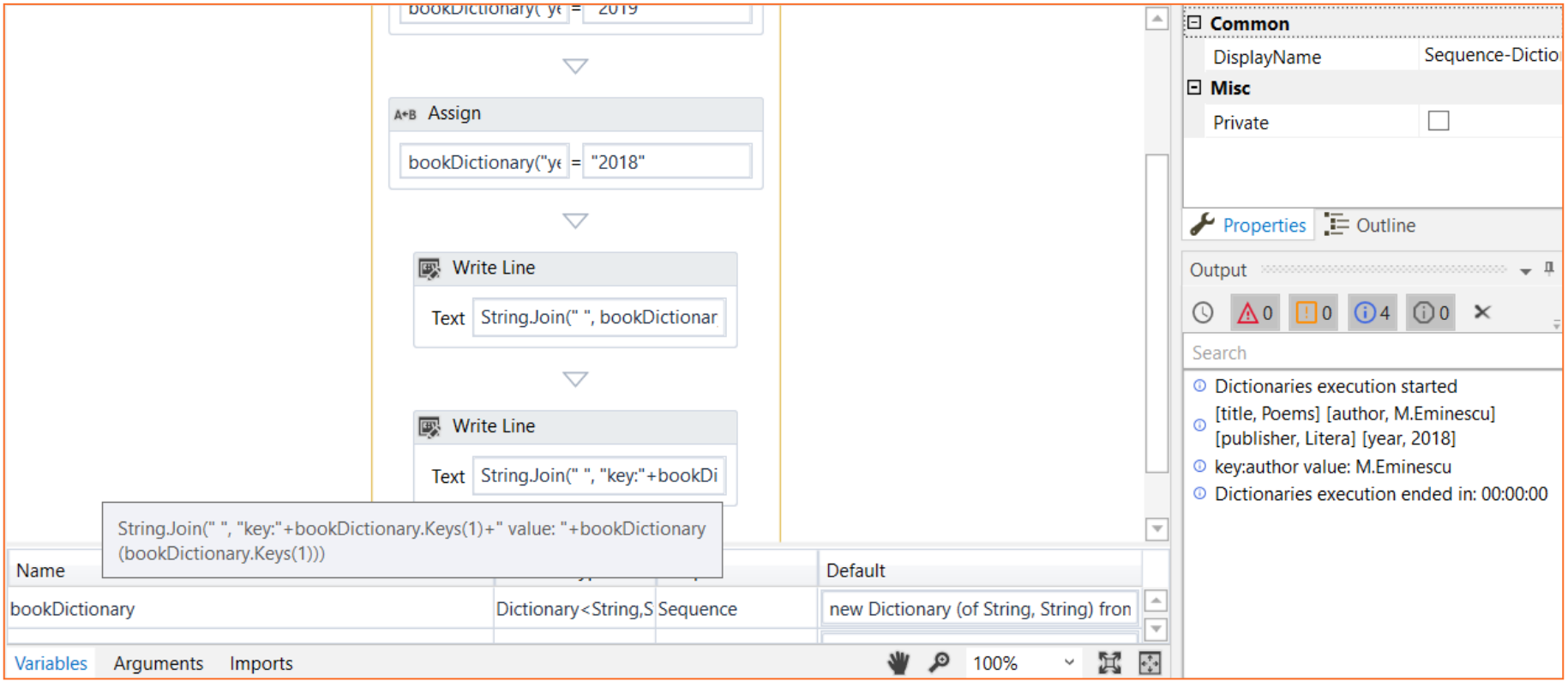
UiPath™

Dictionary. Operations

- ways to add pairs in a dictionary:
  - **Assign** activity:
    - bookDictionary("year") = "2019"// overrides the value on key “year” or adds a new pair {"year", "2019"}
    - monthDictionary(30)= new List(of String) from {"April", "June", "September"}
  - **Add To Collection** activity:
    - for monthDictionary = new Dictionary (of Int32, List(of String)) the properties that are set:
      - Collection = monthDictionary(31);
      - Item = "March";

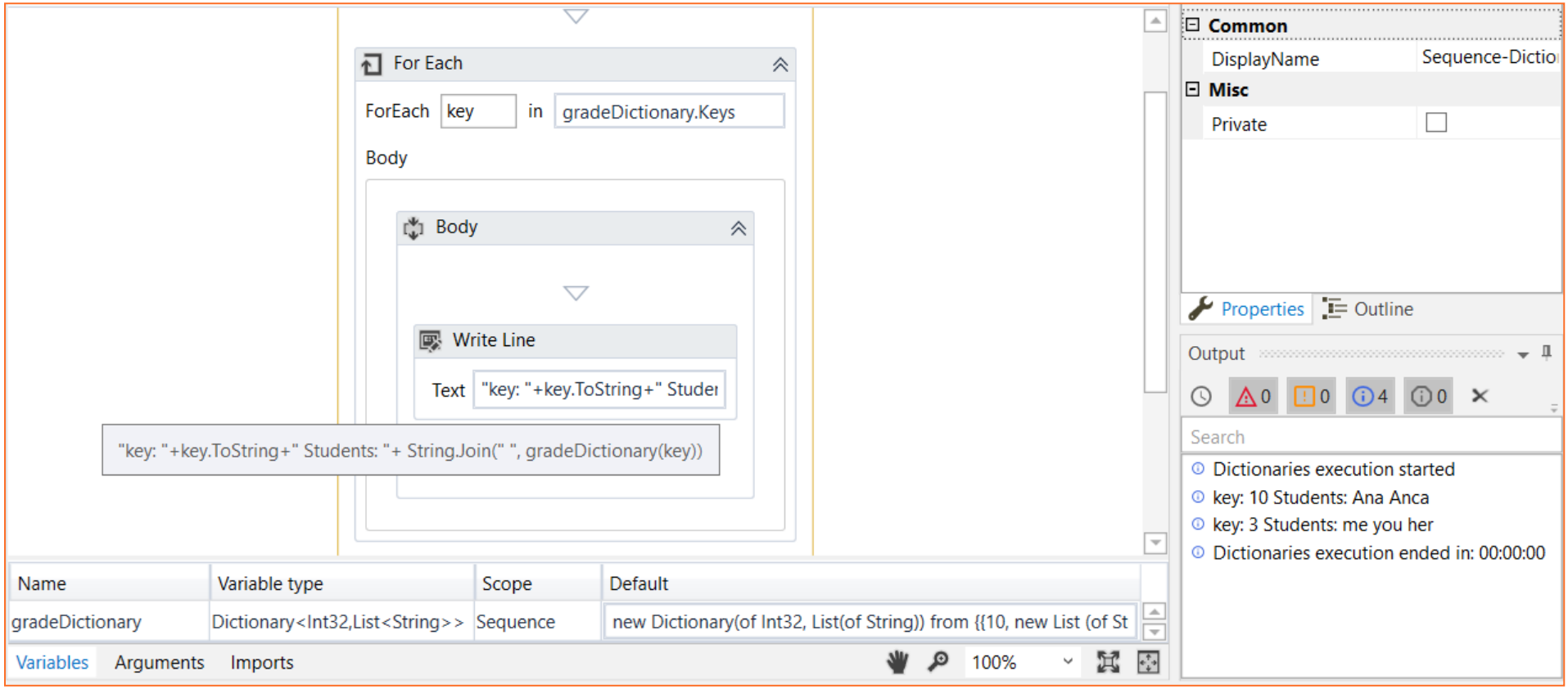
see **Demo6 - Dictionaries**

Dictionaries. Example 1



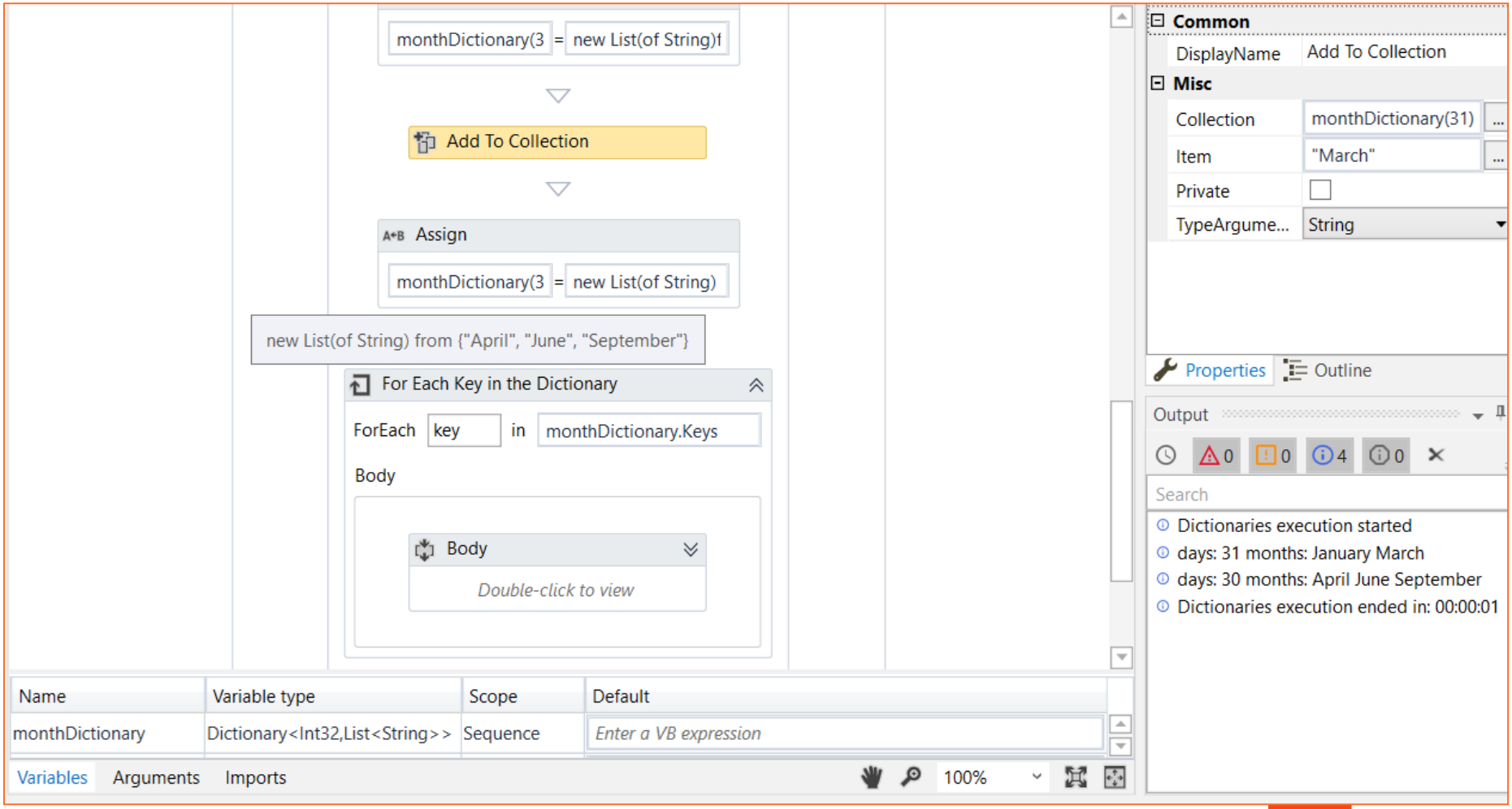
see **Demo6 - Dictionaries**

Dictionaries. Example 2



see **Demo6 - Dictionaries**

Dictionaries. Example 3



see **Demo6 - Dictionaries**

Demo 7

- **Create a process that performs the following actions:**
  - **1. *read* pairs of (continent, country):**
    - **1.1. *build* a dictionary of countries organized by continents;**
    - **E.g.:** {"Asia-Japan","North America-USA", "Europe-Romania", "South America-Argentina", "North America-Canada", "Asia-China", "Australia-Australia"}
  - **2. *print* the dictionary sorted by continents;**
  - **3. *write* the dictionary details into a .txt file;**
    - ***use Append Line activity.***

see **Demo7 - CountryDictionary**

Data Table. Details

- **Data Tables** characteristics in UiPath:
  - **a data structure with flexible length;**
  - it is similar to a Excel sheet consisting of rows and columns;
  - it can be iterated by using a **For Each Row** activity;
  - it is used for:
    - storing **data from Excel sheets** and .csv files;
    - web **data scrapping**.

Row/ Column	First	Last	Club Member
0	“John”	“Doe”	Yes
1	“Jane”	“Doe”	No
2	“Jane”	“Doe”	Yes
3	“John”	“Doe”	No

see **Demo8 - DataTables**

UiPath™

Data Table. Declaration. Instantiation. Initialization

- ways to **declare/instantiate/initialize** a dictionary:
  - **Variables Panel:**
    - **Name:** studentsTable; **Type:** DataTable;
  - **Read CSV** activity:
    - properties to be set:
      - FilePath = "members.csv“;
      - IncludeColumnNames = *checked*;
      - DataTable = membersDataTable.

Row/ Column	First	Last	Club Member
0	“John”	“Doe”	Yes
1	“Jane”	“Doe”	No
2	“Jane”	“Doe”	Yes
3	“John”	“Doe”	No

see **Demo8 - DataTables**

UiPath™

Data Table. Operations (1)

- ways to **convert Data Table to String**:
  - **Output Data Table** activity:
    - properties to be set:
      - Input = membersDataTable;
      - Output = <a String variable>;
- ways to **access data by rows** in a data table:
  - **For Each Row** activity:
    - variable to iterate **DataRow** objects in arrays, e.g., **row**;
    - accessing a field in a **row** formed of [First, Last, Club Member] attributes:
      - firstName= row("first").ToString;
    - field name is case insensitive, e.g., first, First;

see **Demo8 - DataTables**

UiPath™

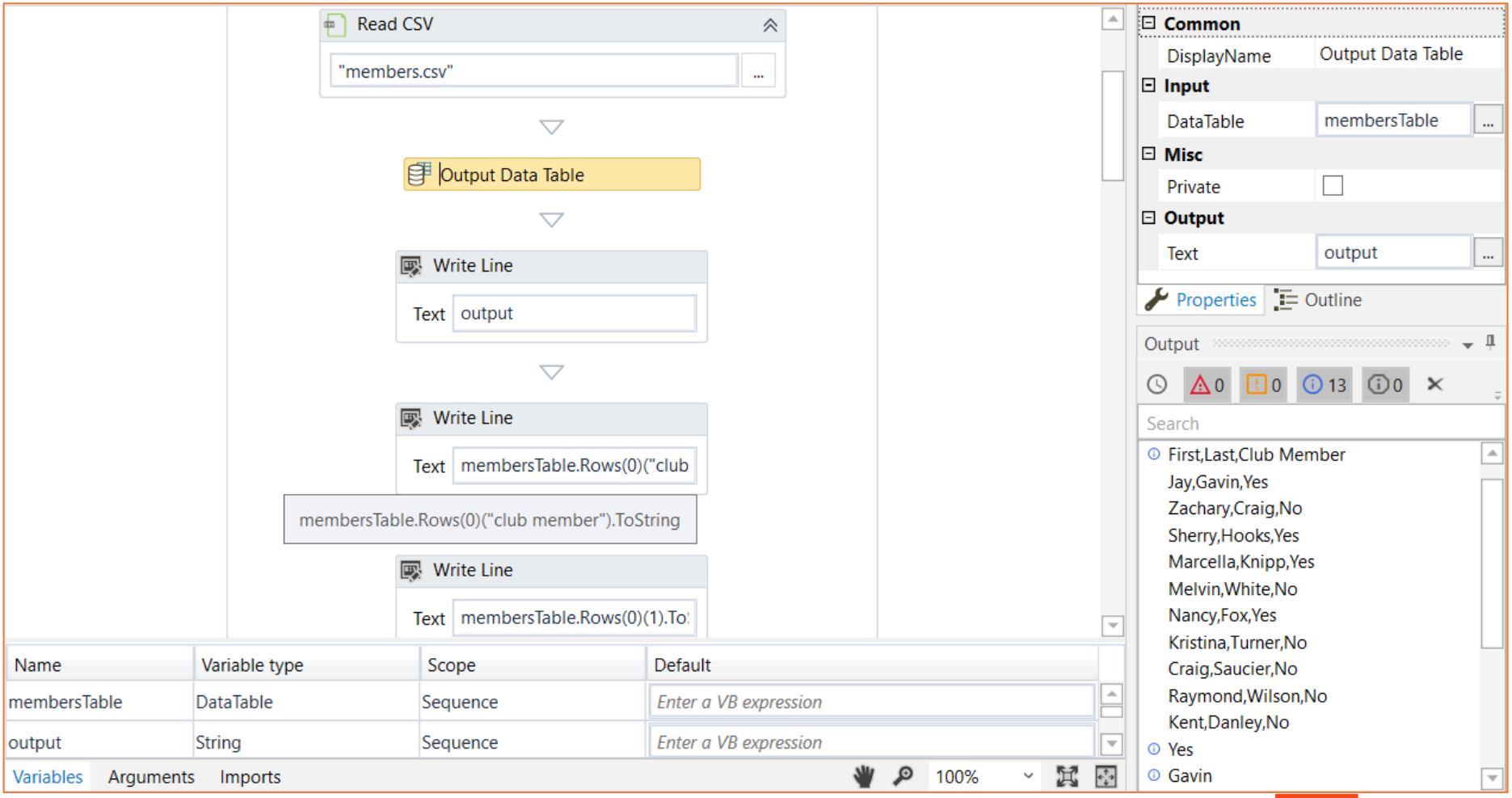
Data Table. Operations (2)

- ways to **access data by indexing rows/columns** in data table:
  - firstName = membersDataTable.Rows(1)(“first”).ToString;
  - status = membersDataTable.Rows(0)(“club member”).ToString;
  - lastName = membersTable.Rows(0)("Last").ToString;
  - lastName = membersTable.Rows(0)(1).ToString;
- ways to **filter data by using rules** in a data table:
  - **Select** method:
    - **Array of DataRow** filtered = membersTable.Select("first>'M' AND"+"[club member]='YES'");
    - the result is an **Array** iterated with **For Each** activity;
  - **Filter Data Table** activity:
    - it allows to follow a wizard that states the rules and the output columns;
    - the result is a **Data Table** iterated with **For Each Row** activity.

see **Demo8 - DataTables**

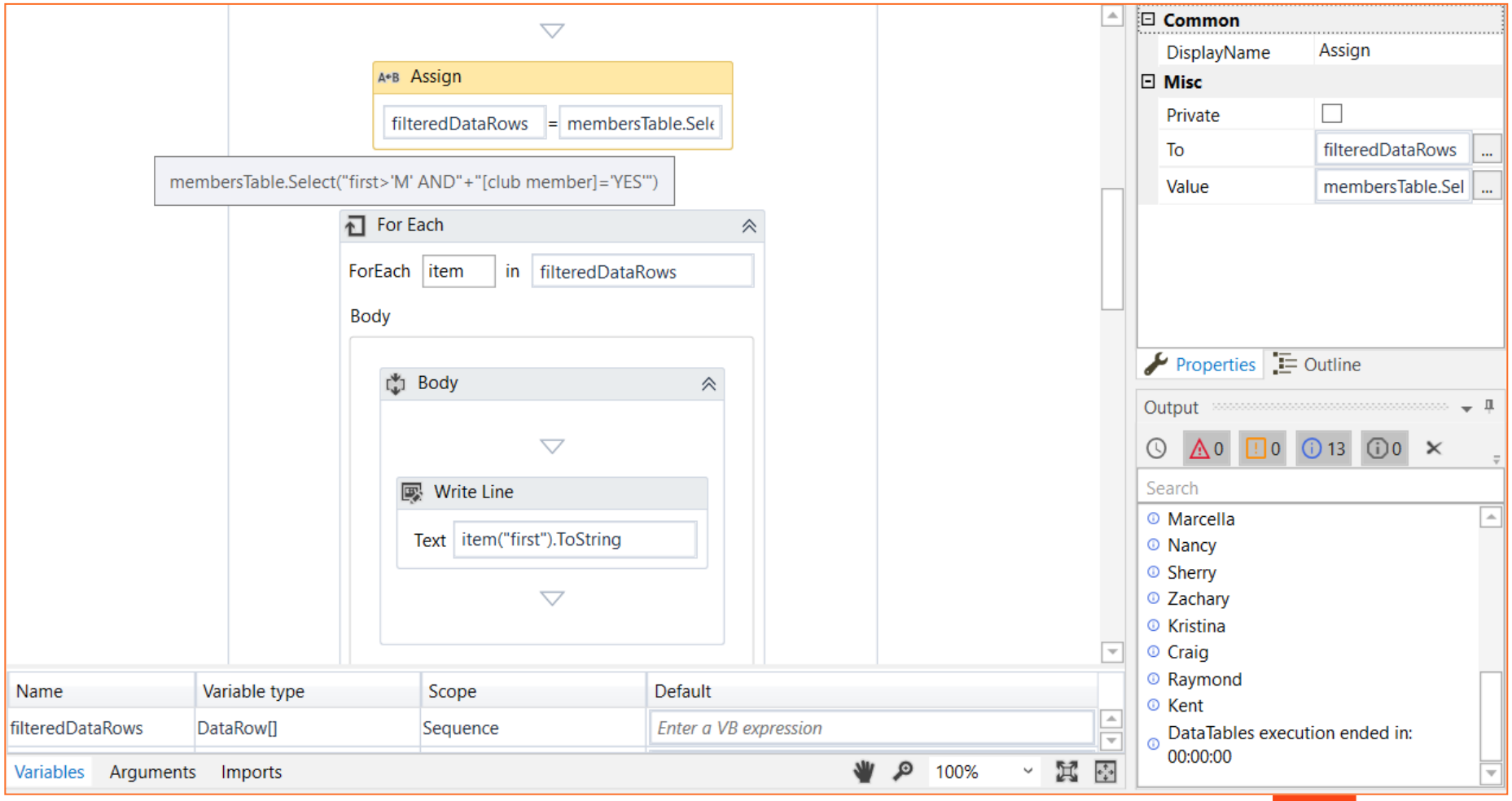
UiPath™

Data Tables. Example 1. Output Data Table



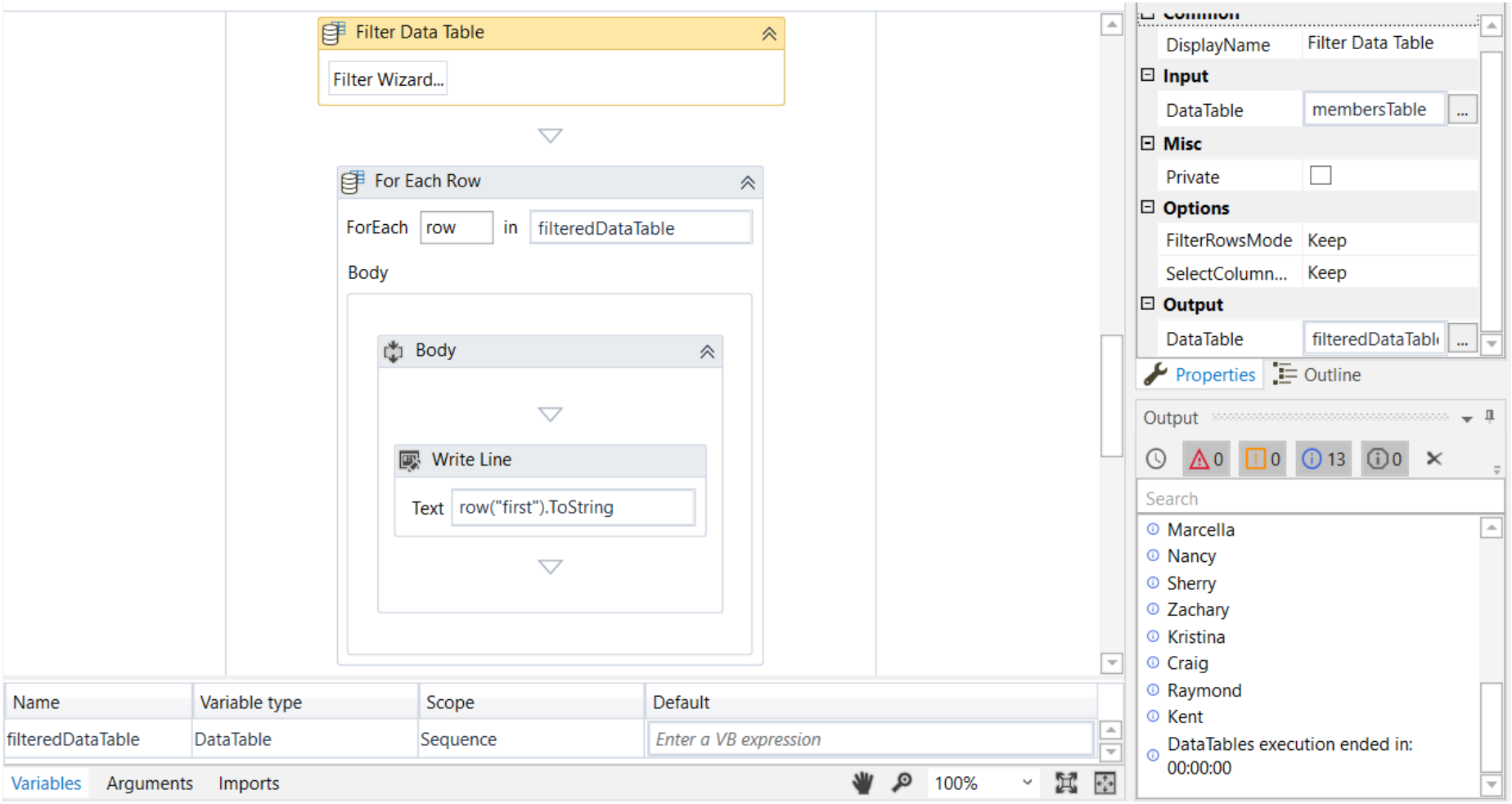
see **Demo8 - DataTables**

Data Tables. Example 2A. Select method



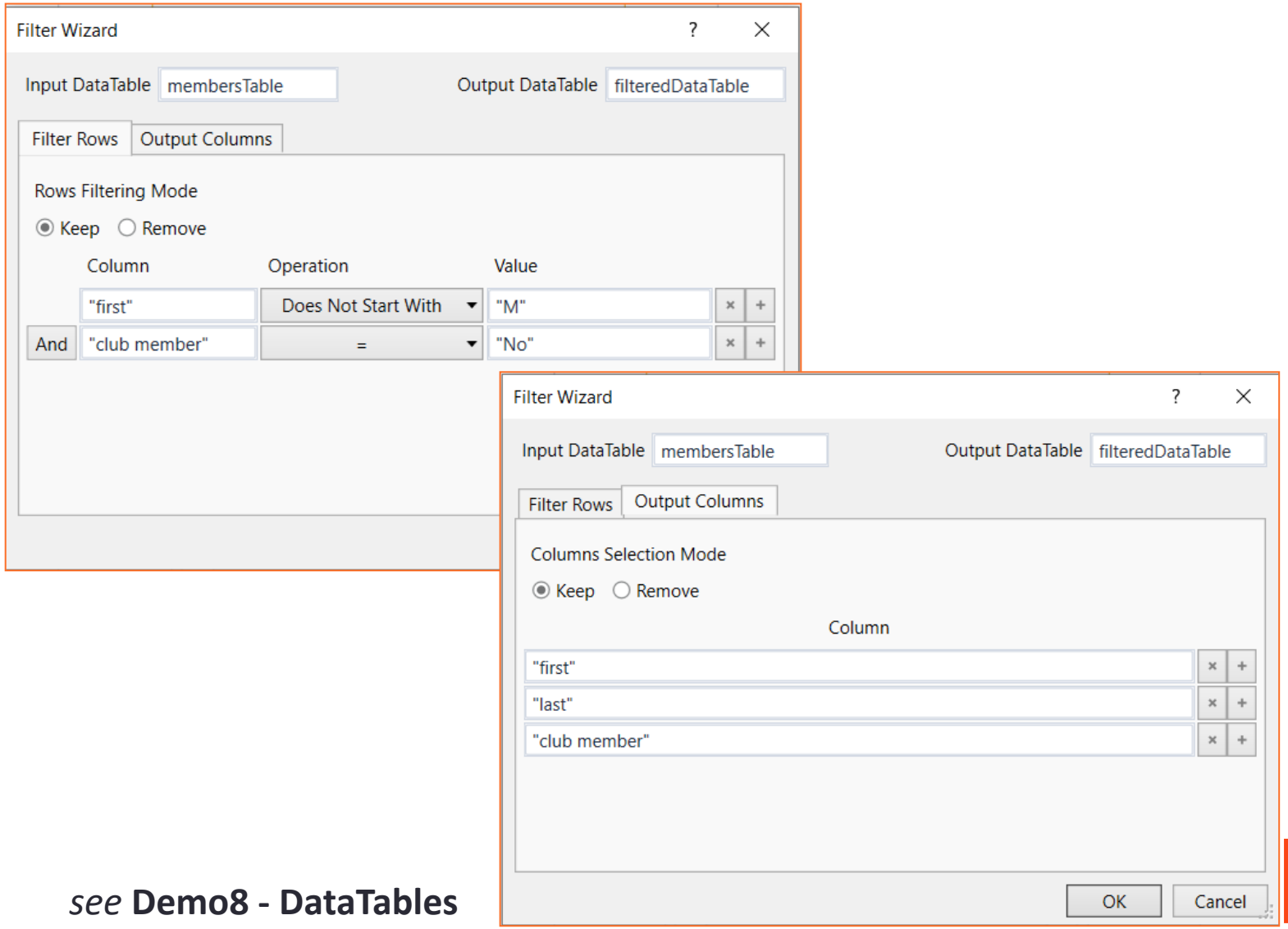
see **Demo8 - DataTables**

Data Tables. Example 2B. Filter Data Table



see **Demo8 - DataTables**

Data Tables. Example 2B. Filter Data Table - Wizard



see **Demo8 - DataTables**

Demo 9

- **Create a process that performs the following actions:**
  - **1. *read* "Students.csv" file with the following structure: Student, Specialisation, Group;**
    - ***use Read CSV activity;***
  - **2. *print* the .csv file content;**
    - ***use Output Data Table activity;***
  - **3. *enter* a specialisation;**
  - **4. *filter* students by specialisation and order them by group;**
    - ***use various ways:***
      - **filteredSortedDataTable =** (From row In studentsDataTable.Select("specialisation='"+spec+"'") Order By Convert.ToInt32(row("group")), row("student") Select row).ToArray.CopyToDatatable()
      - **filteredDataTable =** studentsDataTable.Select("specialisation='"+spec+"'")
      - **sortedDataTable <== Sort Data Table activity**
  - **5. *save* the resulted data into a .csv file.**
    - ***use Write CSV activity to write a Data Table object to a .csv file.***

see **Demo9 - StudentsDataTable**

References

- UiPath Docs
  - https://docs.uipath.com/studio/docs/introduction
- UiPath Forum
  - https://forum.uipath.com/
- UiPath Academy
  - https://academy.uipath.com/