



# Variables in OutSystems

---



# Topics

- Variables
  - Input Parameter
  - Output Parameter
  - Local Variable
- Data Types
  - Structures
  - Lists

# Variables

Variables are locations **in memory** that can hold data

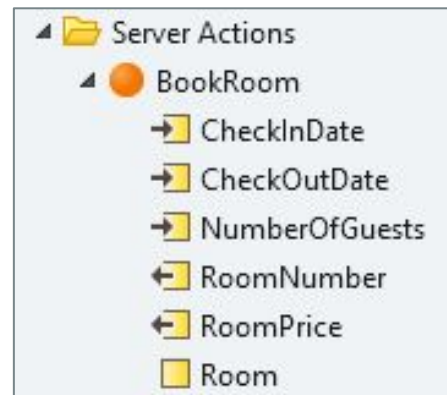
- Hold data of a particular data type
- Can be any data type

Variables are defined and exist in a particular **scope**

- Values can be accessed and modified in that scope
- If execution leaves that scope, the variable is destroyed

Variables can be:

- Input Parameters
- Output Parameters
- Local Variables



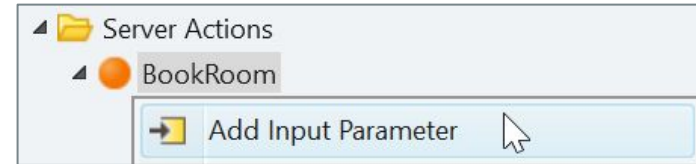
# Input Parameter

Passes a value from the outside scope

Can be set as **Mandatory**

- Requires that the parameter must have a value assigned to it

The variable is destroyed when execution leaves the scope of the parent element



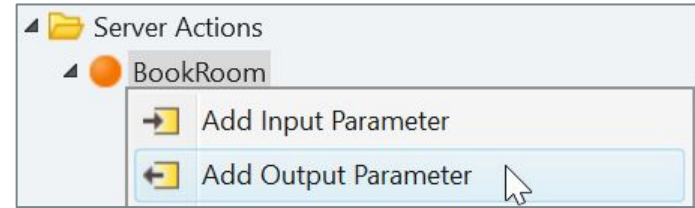
CheckInDate Input Parameter	
Name	CheckInDate
Description	...
Data Type	Date ▼
Is Mandatory	No ▼
Default Value	CurrDate()

# Output Parameter

**Returns** a value to the outside scope

A value **must be assigned** to the Output Parameter inside its scope

The variable **continues to exist** in the outside scope even after its parent element's scope is gone



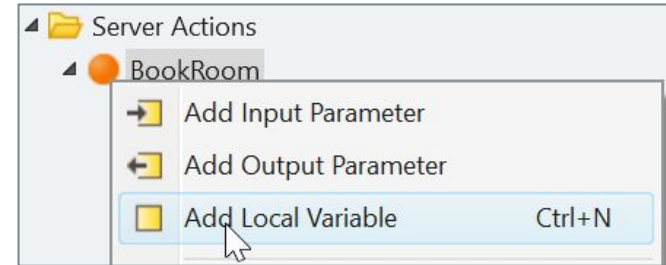
RoomNumber Output Parameter	
Name	RoomNumber
Description	...
Data Type	Integer ▼
Default Value	

# Local Variable

Exists exclusively within the scope of its element

Can be assigned and used “locally” inside that scope

The variable is destroyed when execution leaves the scope of the parent element



Room Local Variable	
Name	Room
Description	...
Data Type	Room
Default Value	

# OutSystems Data Types

OutSystems language is **Strongly Typed**

- Every variable must declare its data type
- That data type can not change

OutSystems supports:

- Basic Types
  - Integer, Text, Date, ...
- Compound Types
  - Entities
  - Static Entities
  - Structures
- Lists

Title Entity Attribute	
Name	Title
Description	...
Label	Title
Data Type	Text
Length	
Is Mandatory	
Default Value	

- Basic Types
- Text
- Integer
- Long Integer
- Decimal
- Boolean
- Date-Time

# Structures

Structures are **custom compound data types**

Used to store compound data **in memory**

- A Structure is the **definition** of a data type
- The Structure itself does not hold any value
  - Structures are not variables
- The Structure's data type is a **collection** of other simple and compound data types that are grouped together

Structures are defined by attributes of any data type

- Including other Structures, Entities and Lists

The screenshot shows the OutSystems Studio interface with the 'Data' tab selected. The left sidebar displays a tree view of the project structure, including 'Hotel', 'Entity Diagrams', 'Entities', and 'Structures'. Under 'Structures', there is a 'Booking' structure with attributes: 'BookingRef', 'CheckInDate', 'CheckOutDate', 'NumberOfGuests', and 'Room'. The 'Room' attribute is highlighted. The main area shows the 'Room' structure attribute definition table.

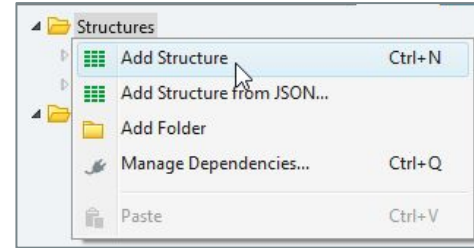
Room	
Name	Room
Description	
Label	Room
Data Type	Room
Is Mandatory	No
Default Value	
JSON...	



# Creating Structures

## Add Structure and set its Properties

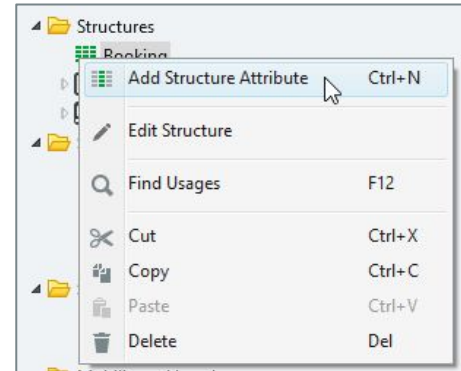
- Set its Name
- Can be public



Booking Structure	
Name	Booking
Description	...
Public	No
More...	...

## Add Structure Attributes

- Set their Names and Data Types
- Can be mandatory
- Can have a default value



Room Structure Attribute	
Name	Room
Description	...
Label	Room
Data Type	Room
Is Mandatory	No
Default Value	
JSON...	...

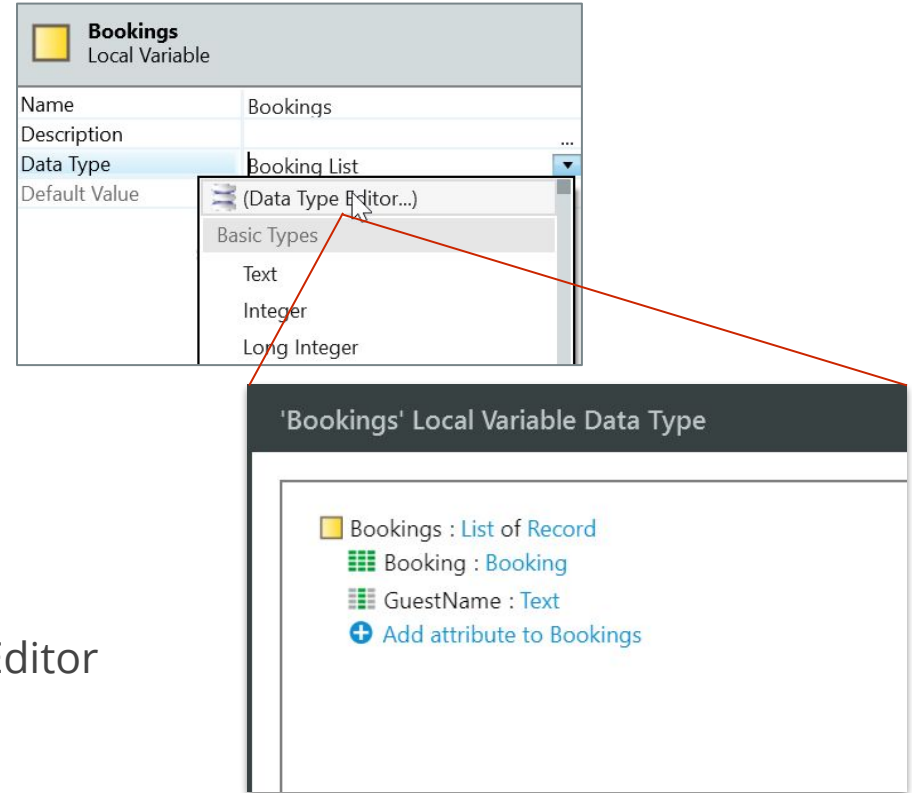
# Lists

A List is a collection of elements of the same data type (e.g. List of Integers)

Elements of OutSystems Lists can be:

- Basic types (e.g. Integer)
- Compound types (e.g. Entities or Structure)
- A union record between different types
  - e.g. Integer + Text
  - e.g. Booking + GuestName

Type of List can be defined in the Data Type Editor



# Summary

- Variables
  - Input Parameter
  - Output Parameter
  - Local Variable
- Data Types
  - Structures
  - Lists



**OutSystems Variables**  
**Thank You!**