AnyLogic – Tutorial 01

Introduction and Java in AnyLogic

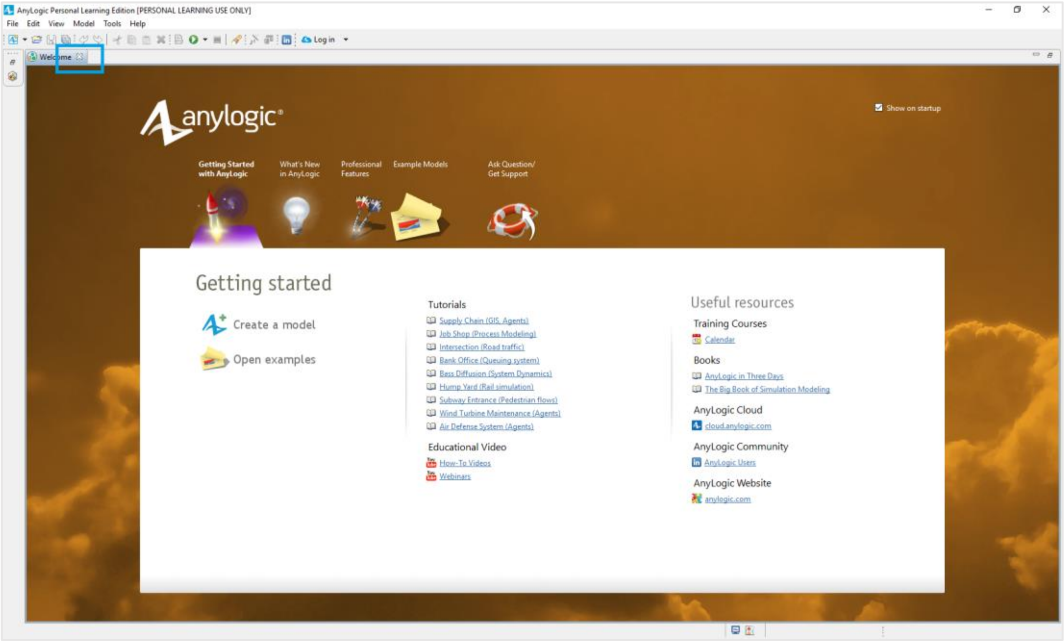
1. AnyLogic - Source

AnyLogic Personal Learning Edition 8.7.8

Download https://www.anylogic.com/downloads/

Java and Eclipse based application

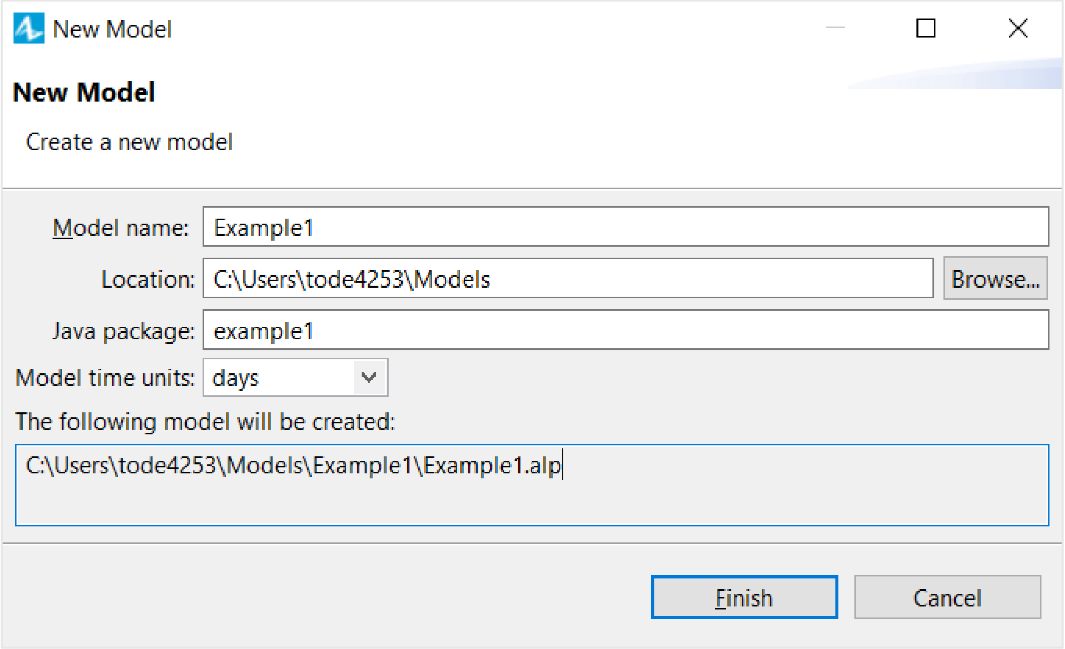
3. Getting started


AnyLogic Personal Learning Edition [PERSONAL LEARNING USE ONLY]
File Edit View Model Tools Help
Log in
Welcome
anylogic
☑ Show on startup
Getting Started with AnyLogic
What's New in AnyLogic
Professional Features
Example Models
Ask Question/ Get Support
Getting started
Create a model
Open examples
Tutorials
Supply Chain (GIS, Agents)
Job Shop (Process Modeling)
Intersection (Road traffic)
Bank Office (Queuing system)
Bass Diffusion (System Dynamics)
Hump Yard (Rail simulation)
Subway Entrance (Pedestrian flows)
Wind Turbine Maintenance (Agents)
Air Defense System (Agents)
Educational Video
How-To Videos
Webinars
Useful resources
Training Courses
Calendar
Books
AnyLogic in Three Days
The Big Book of Simulation Modeling
AnyLogic Cloud
cloud.anylogic.com
AnyLogic Community
AnyLogic Users
AnyLogic Website
anylogic.com

Welcome page

Close the Welcome page

Click *File* > *New* > *Model*



New Model

New Model

Create a new model

Model name: Example1

Location: C:\Users\tode4253\Models

Java package: example1

Model time units: days

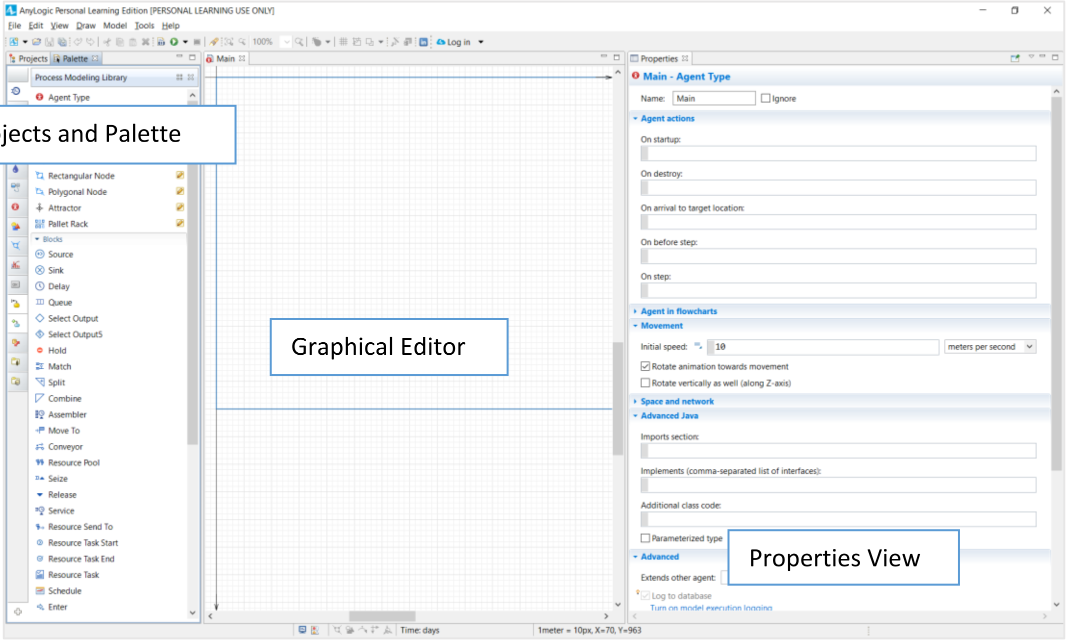
The following model will be created:

C:\Users\tode4253\Models\Example1\Example1.alp

New model

Click *Finish*

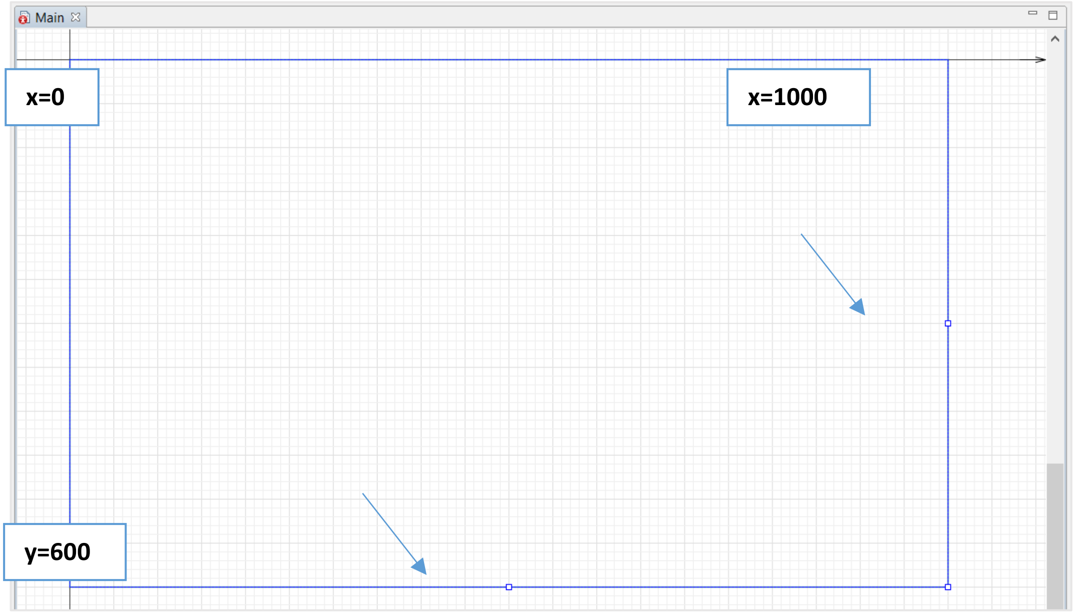
4. User Interface



AnyLogic Personal Learning Edition [PERSONAL LEARNING USE ONLY]
File Edit View Draw Model Tools Help
Projects
Palette
Main
Properties
Main - Agent Type
Projects and Palette
Graphical Editor
Properties View

Workspace

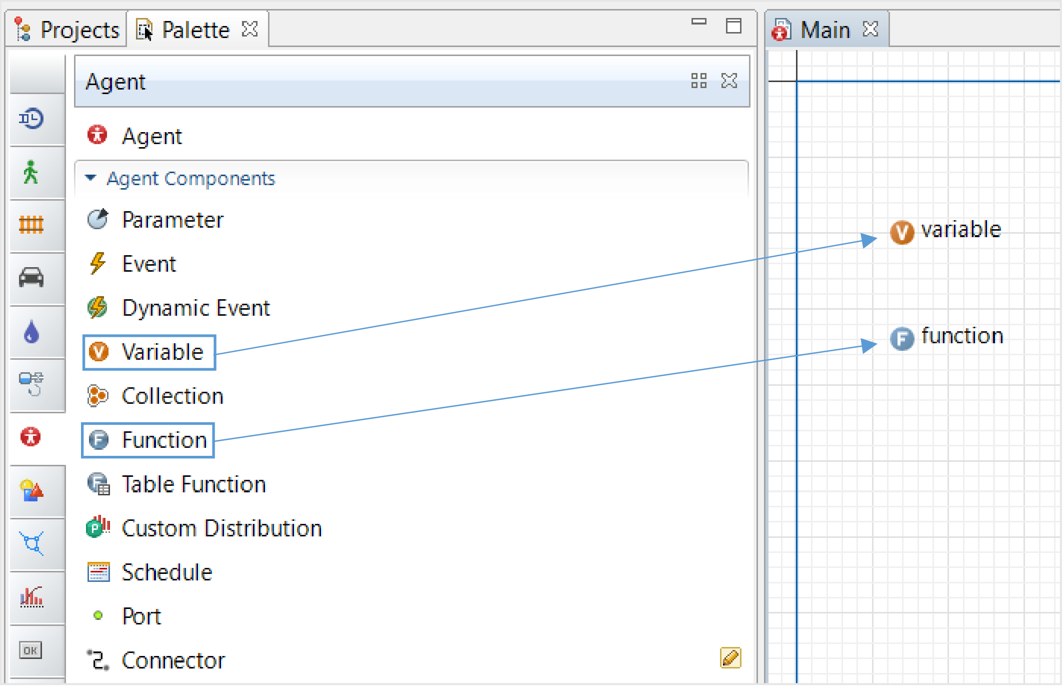
The Project view shows the currently opened models in a tree structure. In the graphical editor you can add elements by dragging them from the Palette to the canvas. The Palette shows a list of elements grouped by categories (e.g. Agent, System Dynamics, Statechart). In the Properties view you can modify properties of the selected item. Click **Tools > Reset Perspective** to get the default view position.



Main diagram

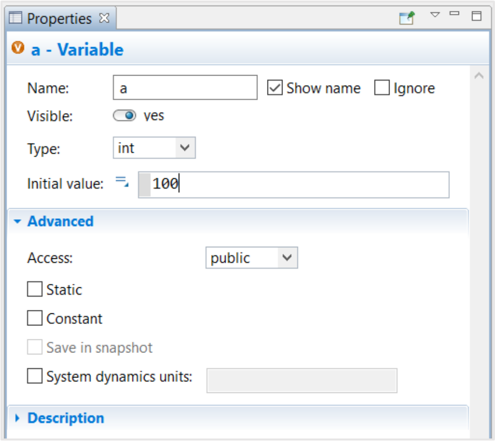
The blue frame in the Main diagram defines the size of the presentation window. You can change the size by dragging the handles. Default: width=1000px and height=600px.

5. JAVA and AnyLogic – Example 1



Variable and Function

Simulations can be extended with Java source code in AnyLogic. You can interactively create variables and functions by dragging and dropping the elements onto the canvas. Subsequently you can customize the properties. Statements within functions are implemented in the function body with a return value.



Properties

a - Variable

Name: a ☑ Show name ☐ Ignore

Visible: yes

Type: int

Initial value: 100

Advanced

Access: public

☐ Static

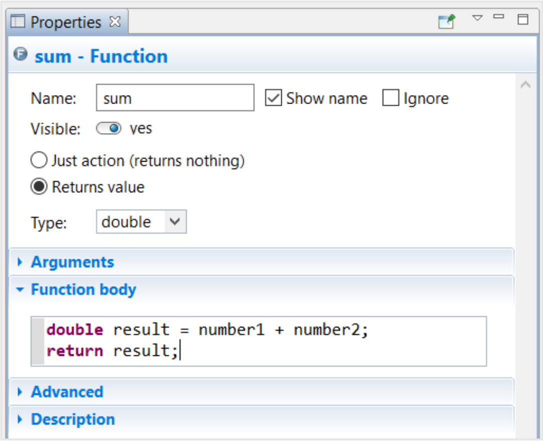
☐ Constant

☐ Save in snapshot

☐ System dynamics units:

Description

Properties Variable



Properties

sum - Function

Name: sum ☑ Show name ☐ Ignore

Visible: yes

○ Just action (returns nothing)

● Returns value

Type: double

Arguments

Function body

```
double result = number1 + number2;
return result;
```

Advanced

Description

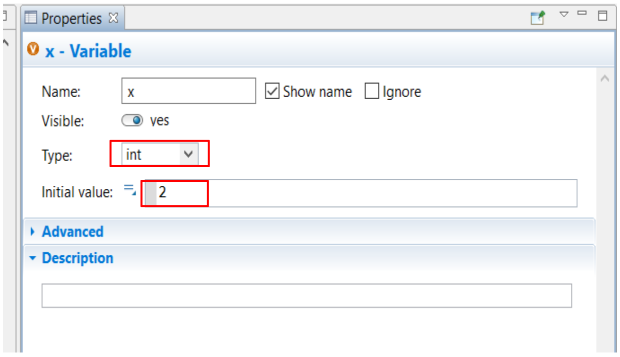
Properties Function

Create an integer variable *x* with an initial value of 2.

Create a function *textoutput* as *Just action (returns nothing)* with the following function body:

System.out.println("Hello world!");

System.out.println("The variable x has the value " + x);



Properties

x - Variable

Name: x ☑ Show name ☐ Ignore

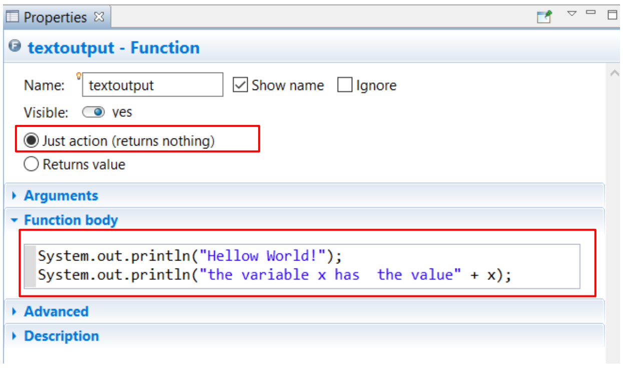
Visible: yes

Type: int

Initial value: 2

Advanced

Description



Properties

textoutput - Function

Name: textoutput ☑ Show name ☐ Ignore

Visible: yes

● Just action (returns nothing)

○ Returns value

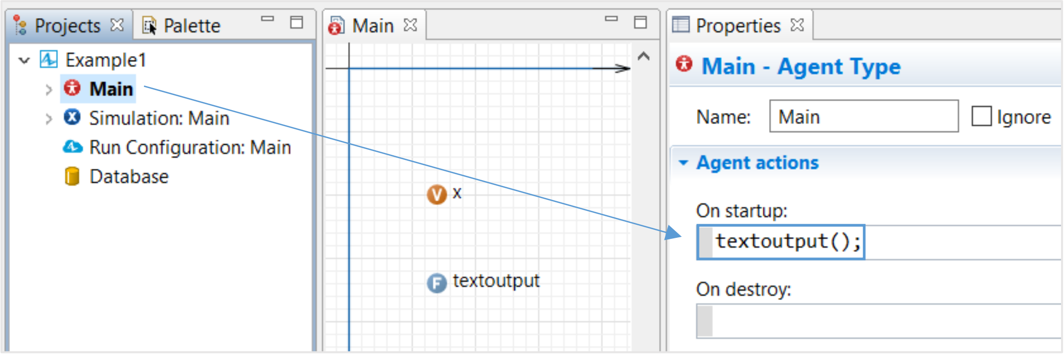
Arguments

Function body

```
System.out.println("Hellow World!");
System.out.println("the variable x has  the value" + x);
```

Advanced

Description



Properties

Main - Agent Type

Name: Main ☐ Ignore

Agent actions

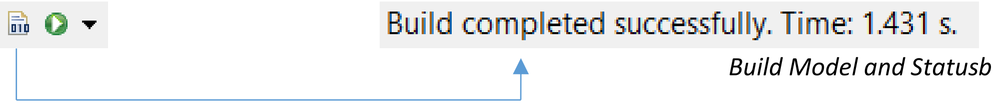
On startup:

```
textoutput();
```

On destroy:

Implement function call

Implement the function call at *On startup* in Main.

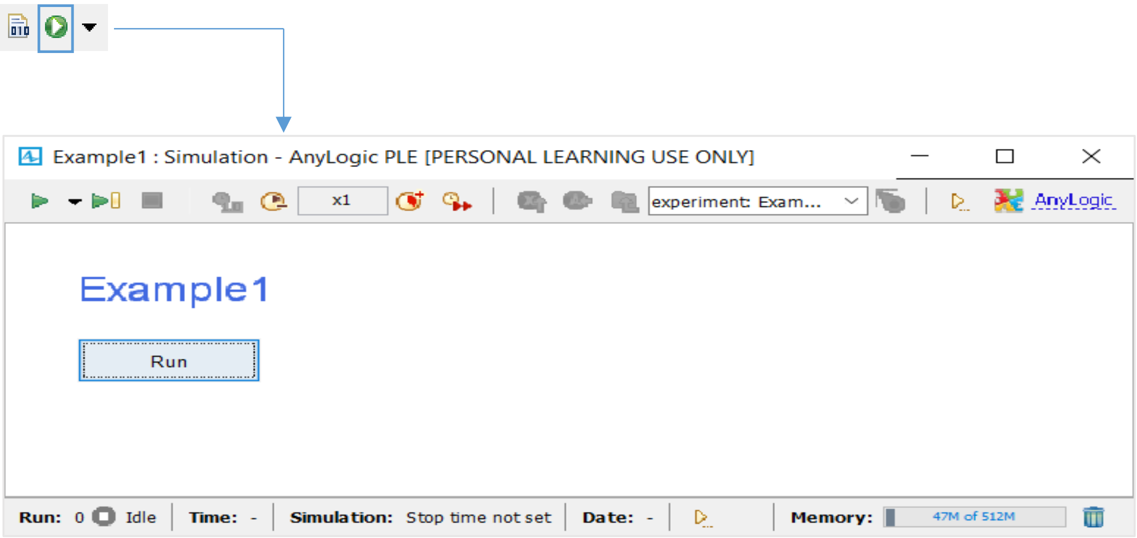


Build completed successfully. Time: 1.431 s.

Build Model and Statusbar

Build the model by clicking the button in the Navigationbar.

Build completed successfully appears in the Statusbar.



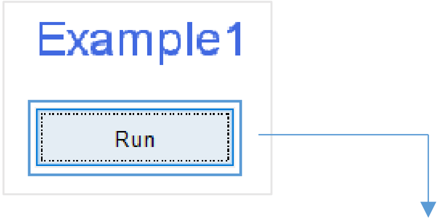
Example1 : Simulation - AnyLogic PLE [PERSONAL LEARNING USE ONLY]

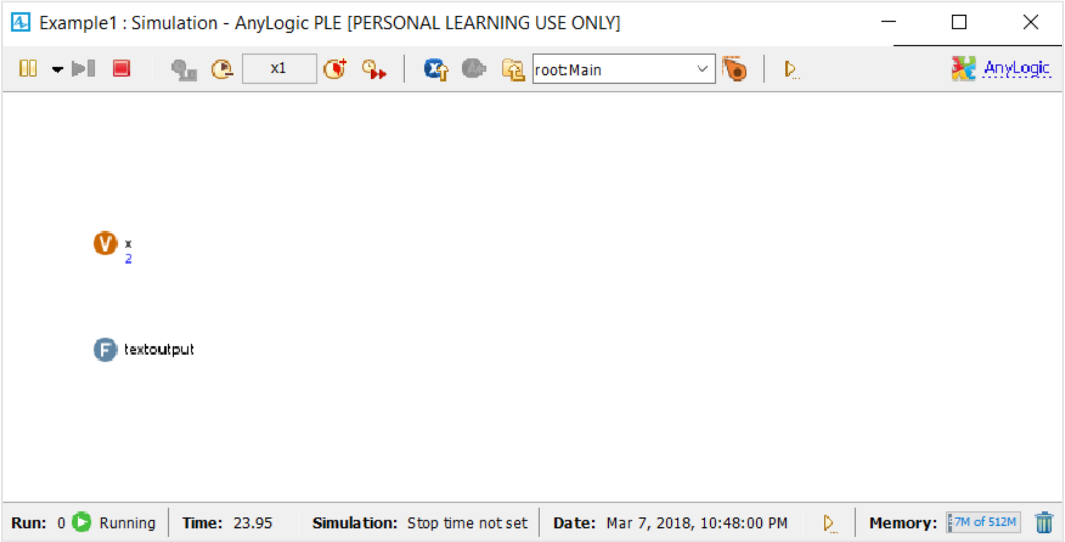
Run: 0 Idle | Time: - | Simulation: Stop time not set | Date: - | Memory: 47M of 512M

Presentation Window

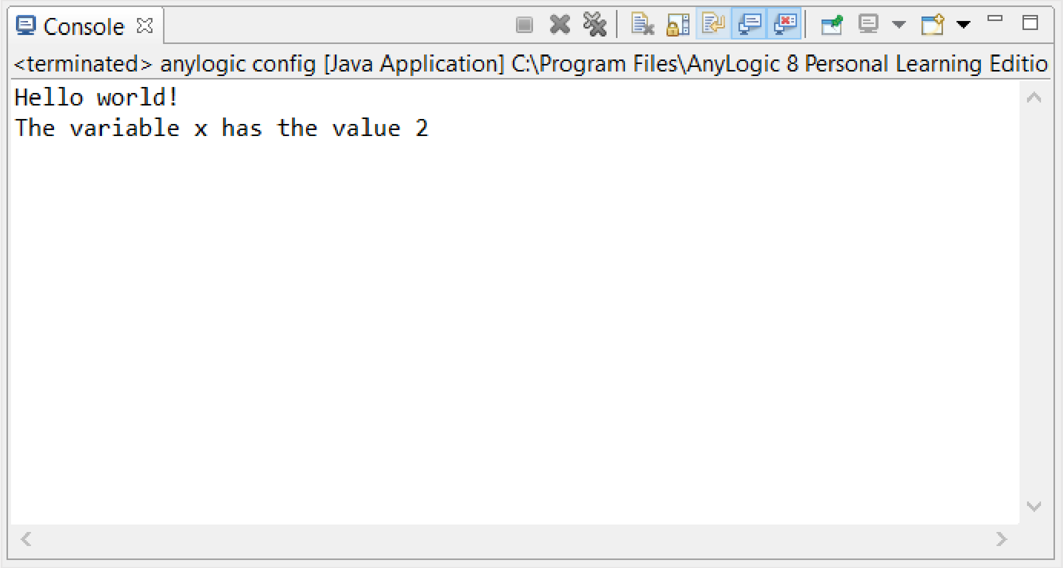
Run the model by clicking the button in the Navigationbar.

The presentation window appears.




Example1 : Simulation - AnyLogic PLE [PERSONAL LEARNING USE ONLY]
x1
root:Main
AnyLogic
x
2
textoutput
Run: 0 Running
Time: 23.95
Simulation: Stop time not set
Date: Mar 7, 2018, 10:48:00 PM
Memory: 7M of 512M

Simulation



```
<terminated> anylogic config [Java Application] C:\Program Files\AnyLogic 8 Personal Learning Editio
Hello world!
The variable x has the value 2
```

Console

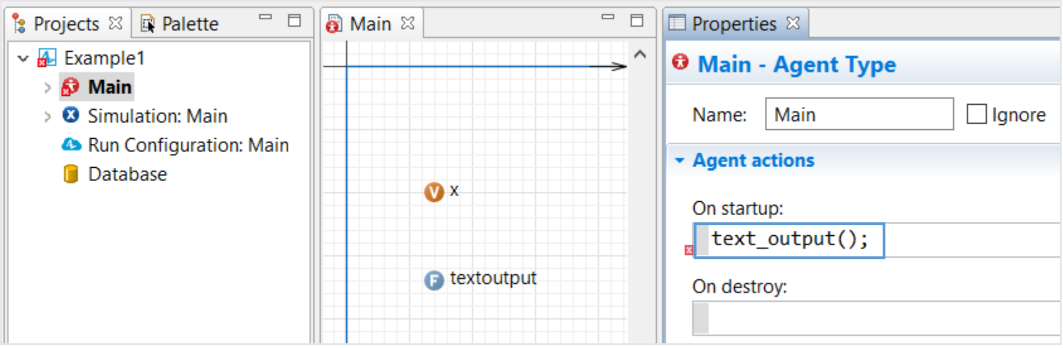
Run the simulation by clicking the Command button *Run*.

The Simulation and the Console window appear.

Stop the Simulation by clicking the red button *Terminate Execution*.

6. JAVA and AnyLogic – Problems View

Failed model checks will cause an error which are displayed in the Problems View.



Name: Main □ Ignore

On startup: `text_output();`

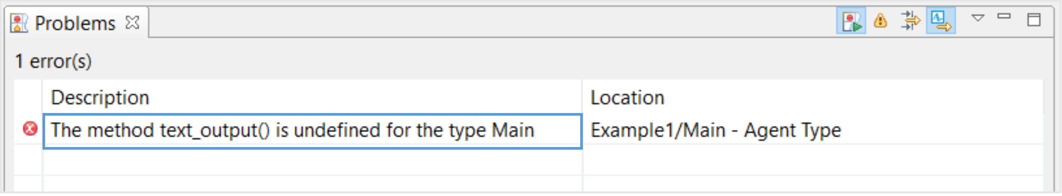
On destroy:

Rename Function

Rename the function call to *text_output();*

Build the model by clicking the button in the Navigationbar.

The Problems View Windows appears.

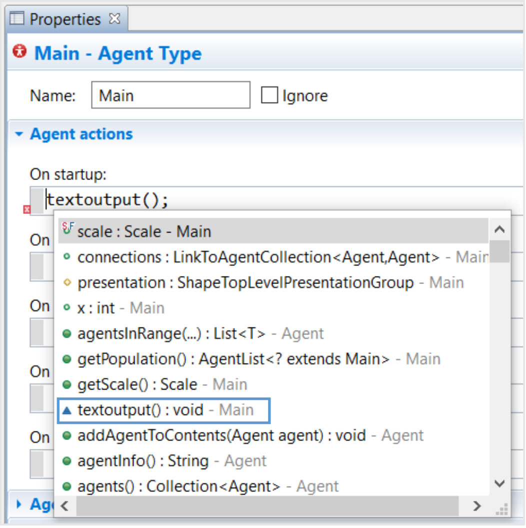


1 error(s)

Description	Location
The method text_output() is undefined for the type Main	Example1/Main - Agent Type

Problems View

Double-clicking on the error line will take you to the appropriate position in the model. You can fix the problem and build the model again. To avoid incorrect spelling of program elements, you can use the Code completion assistant.



Name: Main □ Ignore

On startup: `textoutput();`

```
scale : Scale - Main
connections : LinkToAgentCollection<Agent,Agent> - Main
presentation : ShapeTopLevelPresentationGroup - Main
x : int - Main
agentsInRange(...) : List<T> - Agent
getPopulation() : AgentList<? extends Main> - Main
getScale() : Scale - Main
textoutput() : void - Main
addAgentToContents(Agent agent) : void - Agent
agentInfo() : String - Agent
agents() : Collection<Agent> - Agent
```

Ctrl+Space

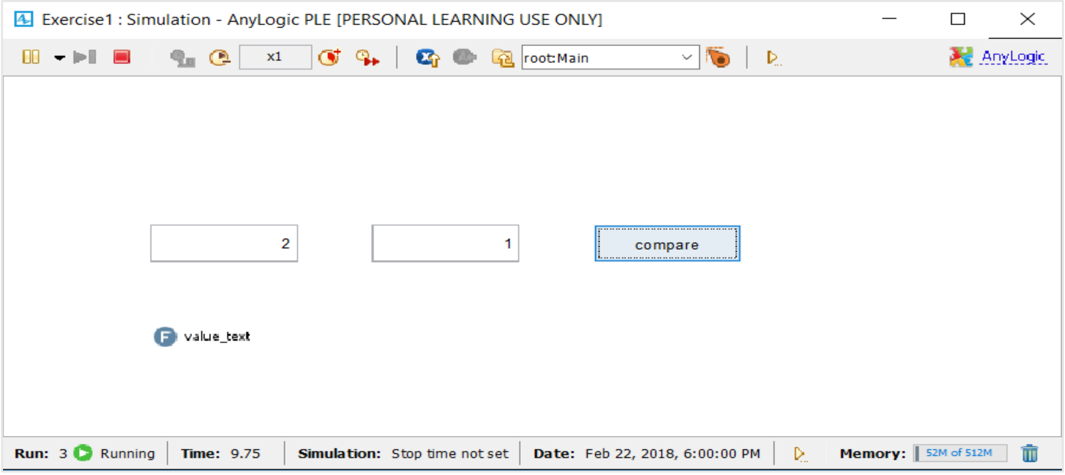
Alt+Space (Mac OS)

Code completion assistant

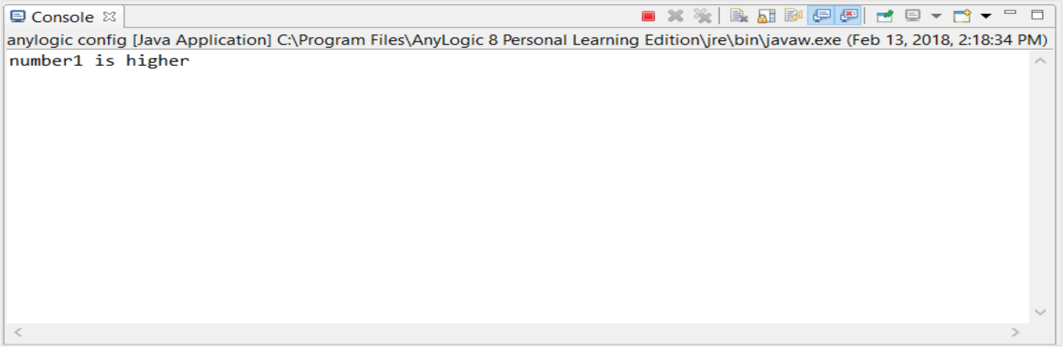
Click *File > Close*

7. JAVA and AnyLogic – Exercise 1

Create a model that compares two input numbers and outputs the result through the console.


Exercise1 : Simulation - AnyLogic PLE [PERSONAL LEARNING USE ONLY]
x1
root:Main
AnyLogic
2
1
compare
value_text
Run: 3 Running
Time: 9.75
Simulation: Stop time not set
Date: Feb 22, 2018, 6:00:00 PM
Memory: 52M of 512M

Simulation



```
anylogic config [Java Application] C:\Program Files\AnyLogic 8 Personal Learning Edition\jre\bin\javaw.exe (Feb 13, 2018, 2:18:34 PM)
number1 is higher
```

Console

Open Palette *Controls*

Create Edit Box. Name: *editBox1*

Create Edit Box. Name: *editBox2*

Create Button. Name: *cmdButton* Action: value_text();

Create Function. Name: *value_text* Just action (returns nothing).

Function body:

```
double num1 = editBox1.getDoubleValue();
double num2=editBox2.getDoubleValue();
      if ( num1>num2)
      {
            System.out.println("Number1 is higher " + num1);
      }
      else if (num1<num2){
            System.out.println("Number2 is higher " + num2);
      }
      else {
            System.out.println("Numbers are equal");
      }
```

Extension exercises

1. Bring the output from the console to a textbox

    Hint : add new textbox and comment the console print code in value_text function, use setText code instead

2. Change the Operation from Comparing to
    a. Addition
    b. Subtraction
    c. Multiplication
    d. division
    e. Exponential: x^y