UAB
Universitat Autònoma
de Barcelona

Introduction

AnyLogic

Author: Alejandro Rodrlguez Grau

Tutor: Prof. Dr. Angel A. Juan

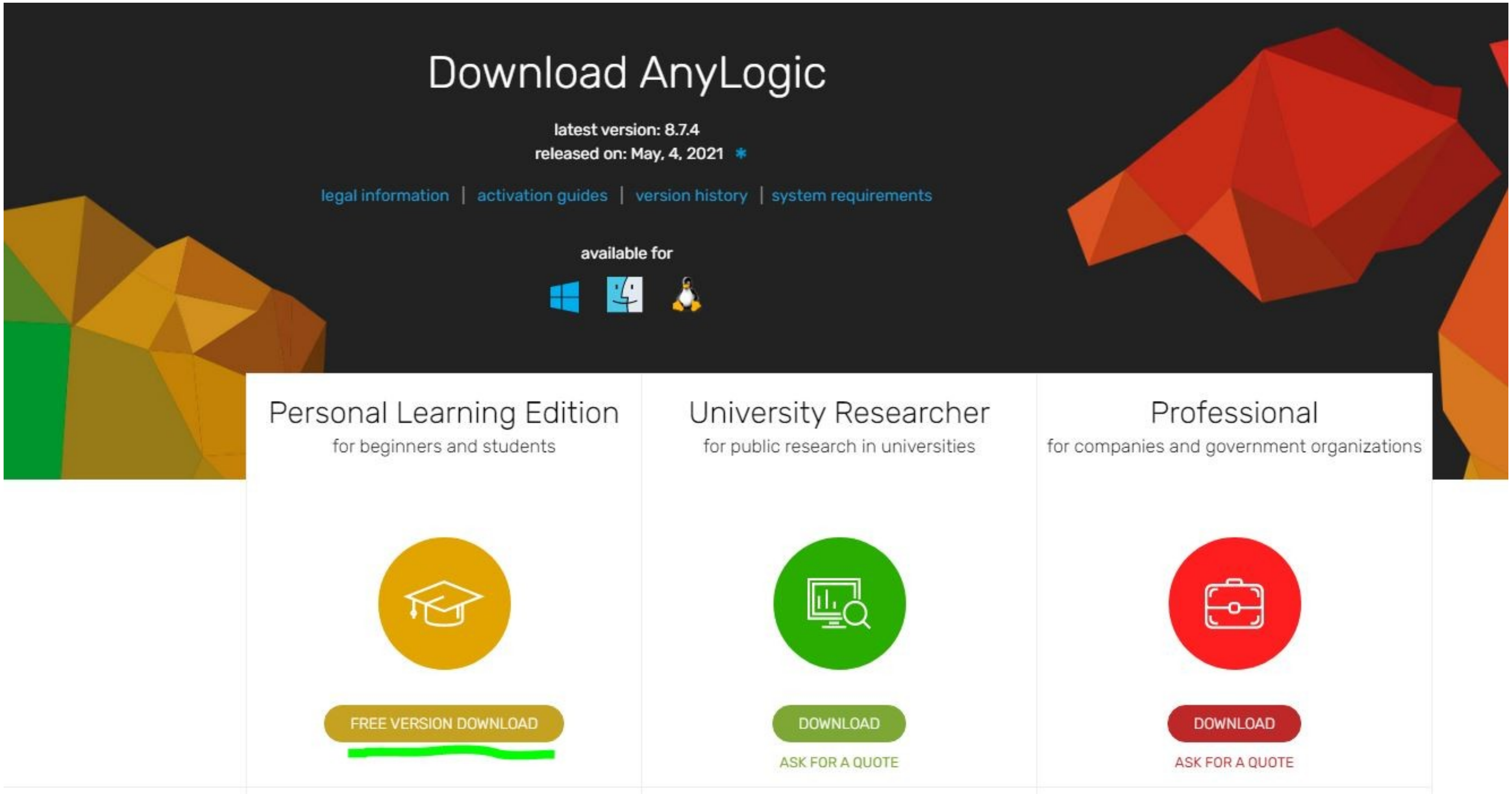
Full Course Index

- Introduction to AnyLogic (First Model)
- Basic models
- Heterogeneous model
- Advanced model

Introduction to AnyLogic

- **Download & Install**
- **Execute AnyLogic**
- **AnyLogic Interface**
- **First Model**
  - Definition
  - Building a New Model
  - Execution
  - Results
  - Experiments and randomness

1. Download & Install



Download AnyLogic

latest version: 8.7.4
released on: May, 4, 2021 *

legal information | activation guides | version history | system requirements

available for

Personal Learning Edition
for beginners and students

FREE VERSION DOWNLOAD

University Researcher
for public research in universities

DOWNLOAD

ASK FOR A QUOTE

Professional
for companies and government organizations

DOWNLOAD

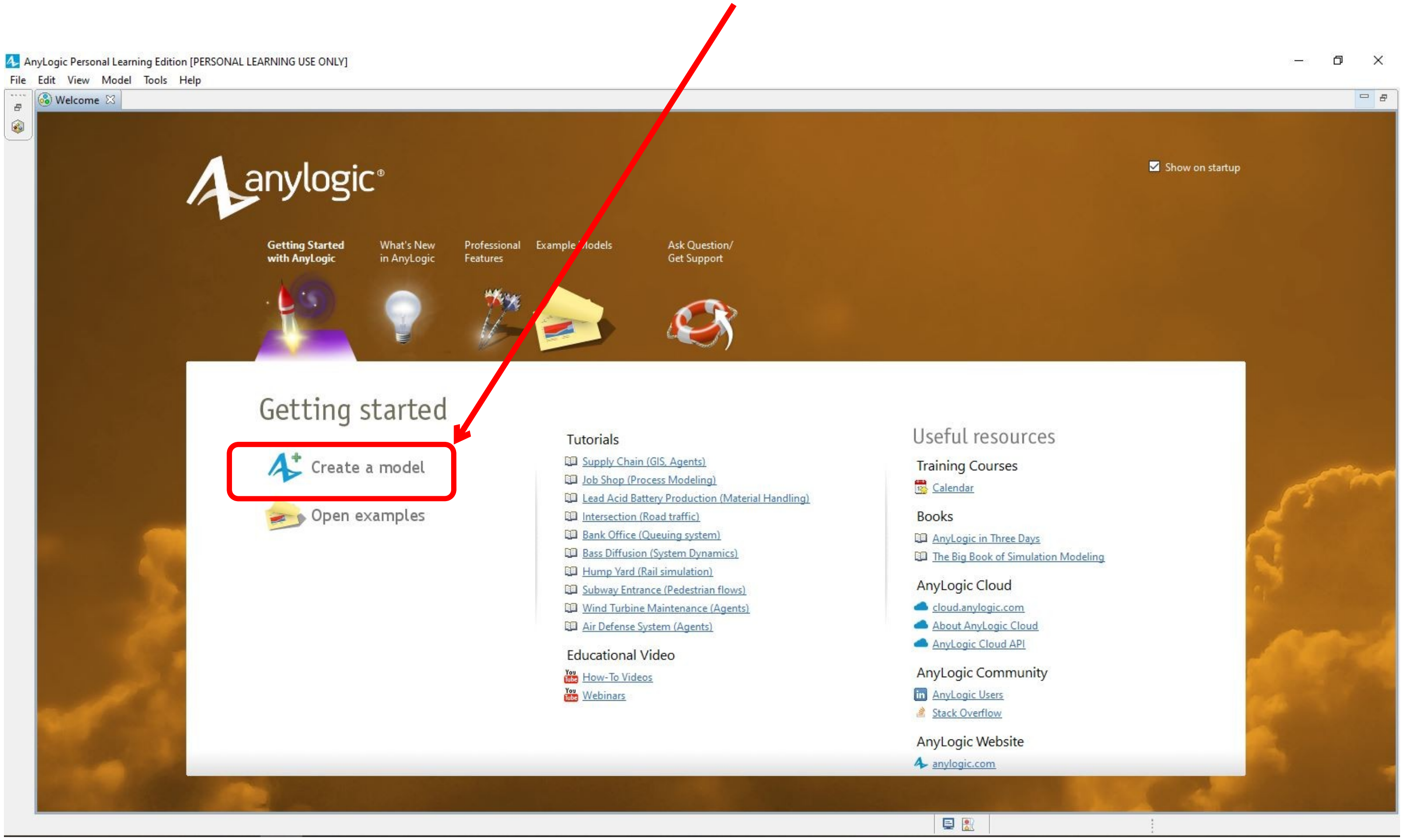
ASK FOR A QUOTE

From website: https://www.anylogic.com/downloads/

2. Execute (1/2)

- Once installed, we can execute the software. We will be greeted with the following intro screen. Here we can find online tutorials for further learning if desired, after or along this course.

- Lets get started and select *Create a Model.*


AnyLogic Personal Learning Edition [PERSONAL LEARNING USE ONLY]
File
Edit
View
Model
Tools
Help
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)
- Lead Acid Battery Production (Material Handling)
- 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
- About AnyLogic Cloud
- AnyLogic Cloud API

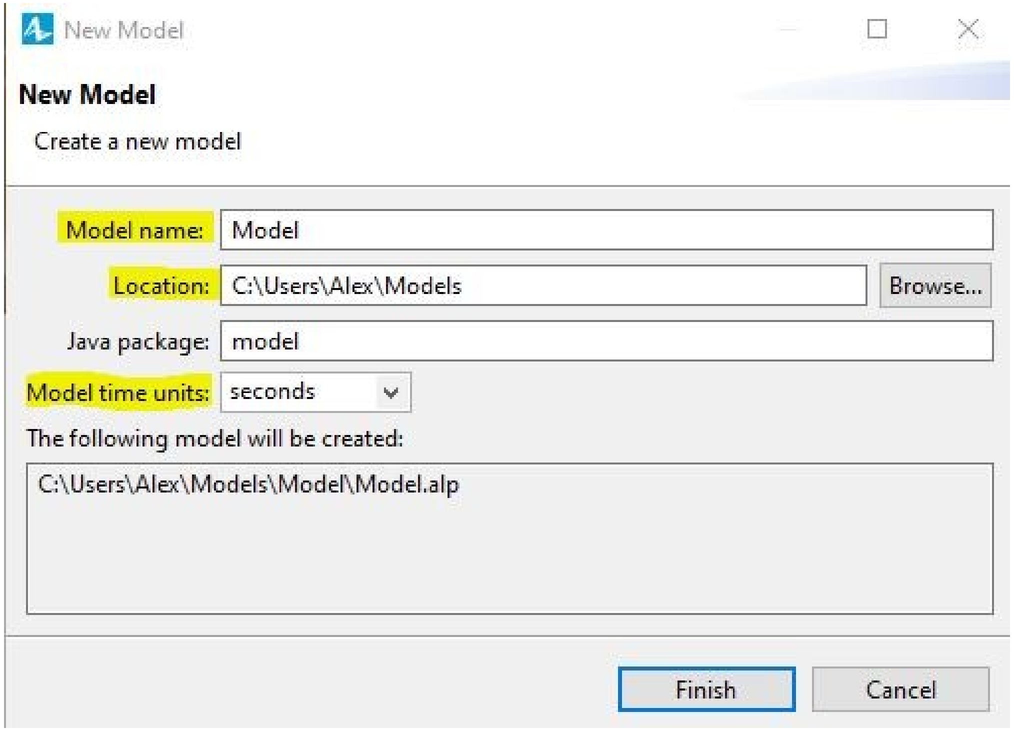
AnyLogic Community

- AnyLogic Users
- Stack Overflow

AnyLogic Website

- anylogic.com

2. Execute (2/2)



New Model

Create a new model

Model name: Model

Location: C:\Users\Alex\Models

Java package: model

Model time units: seconds

The following model will be created:

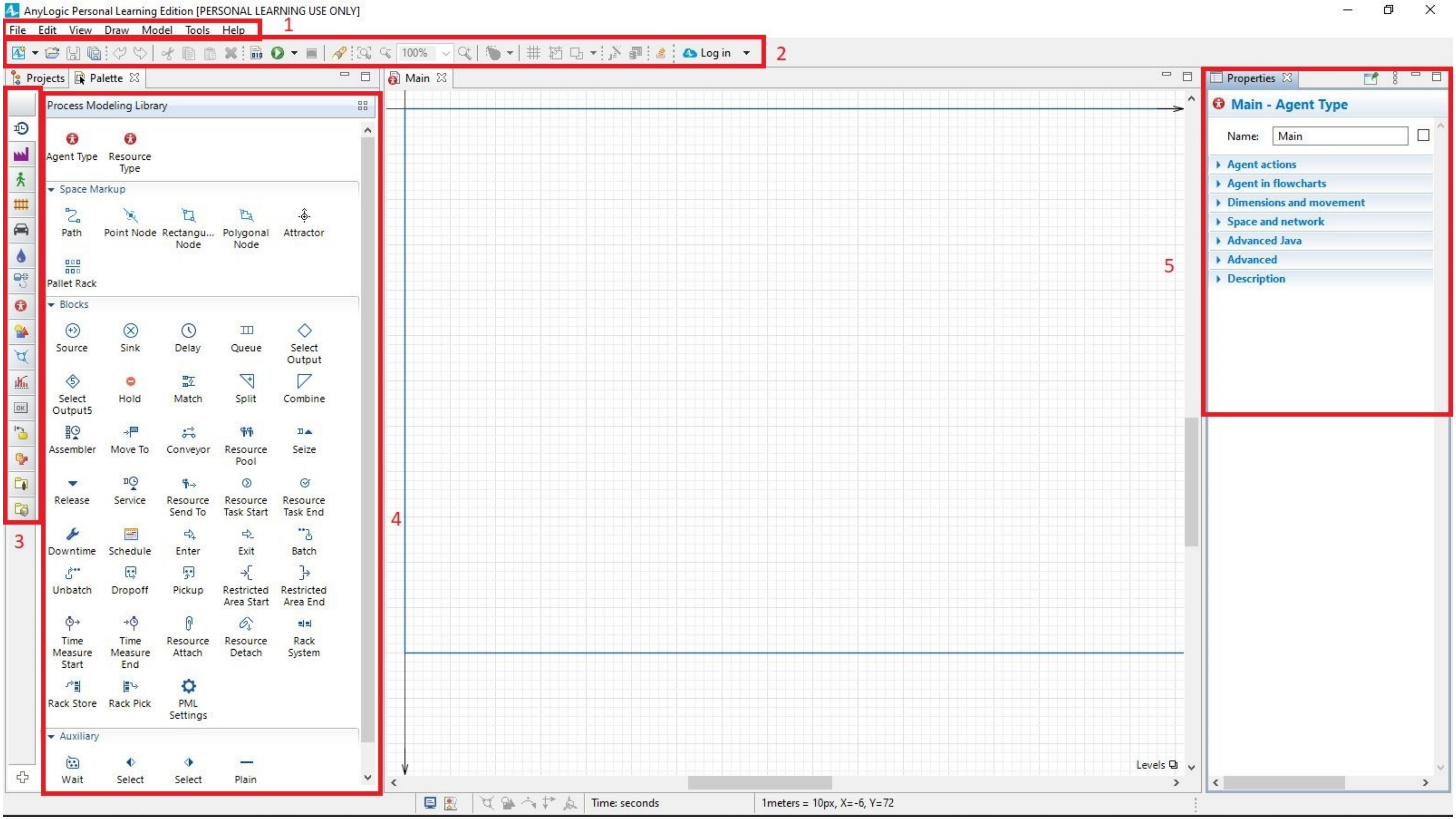
C:\Users\Alex\Models\Model\Model.alp

- **Model name** → When choosing a name for the model, make it easy and choose a short name. Beware of special characters as they could become a problem when executing a model.

- **Location** → It's suggested to create a folder near to the course slides and save there your models.

- **Models time units** → May differ between seconds, hours, days… depending on what are we simulating.

3. AnyLogic Interface (1/2)



Name: Main □

3. AnyLogic Interface (2/2)

- 1 Standard Main Ribbons [Used mainly to *Save as* and change the *View*]

- 2 Specific Ribbons [Similar purpose as 1]

- 3 Project Libraries [All objects are sorted in different *Libraries* depending on type]

- 4 Objects in the selected Project Library [Drag & Drop objects to the centre to add them to your model]

- 5 Properties of selected object/model. [When something is selected, *Properties* will show up here, also this will be where we will do most of our work changing numbers]

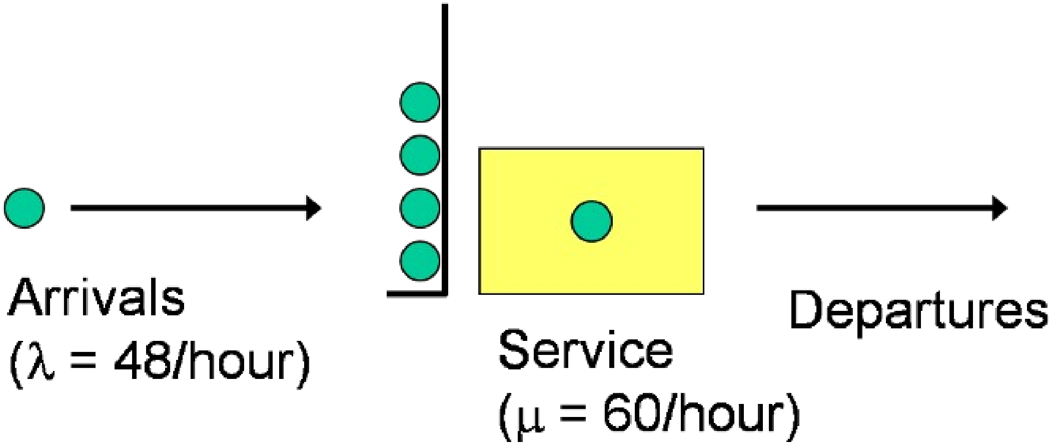
4. First Model

- Goal: to illustrate the simulation model-building process and the statistical analysis of simulation output result.

- Example: We'll use a simple single-server queueing system with Poisson arrival rate (average 48 entities/hour) and exponential service time (average 60 entities/hour).

- Further assumptions: (a) interarrival times and service times are assumed to be independent; (b) queue has infinite capacity; (c) queue discipline will be first-in-first-out (FIFO).

- Questions: number of entities in the queue? time an entity spends in the queue? utilization of the server? etc.



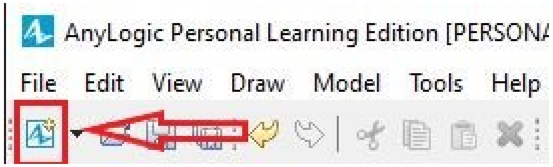
4.1. Definition

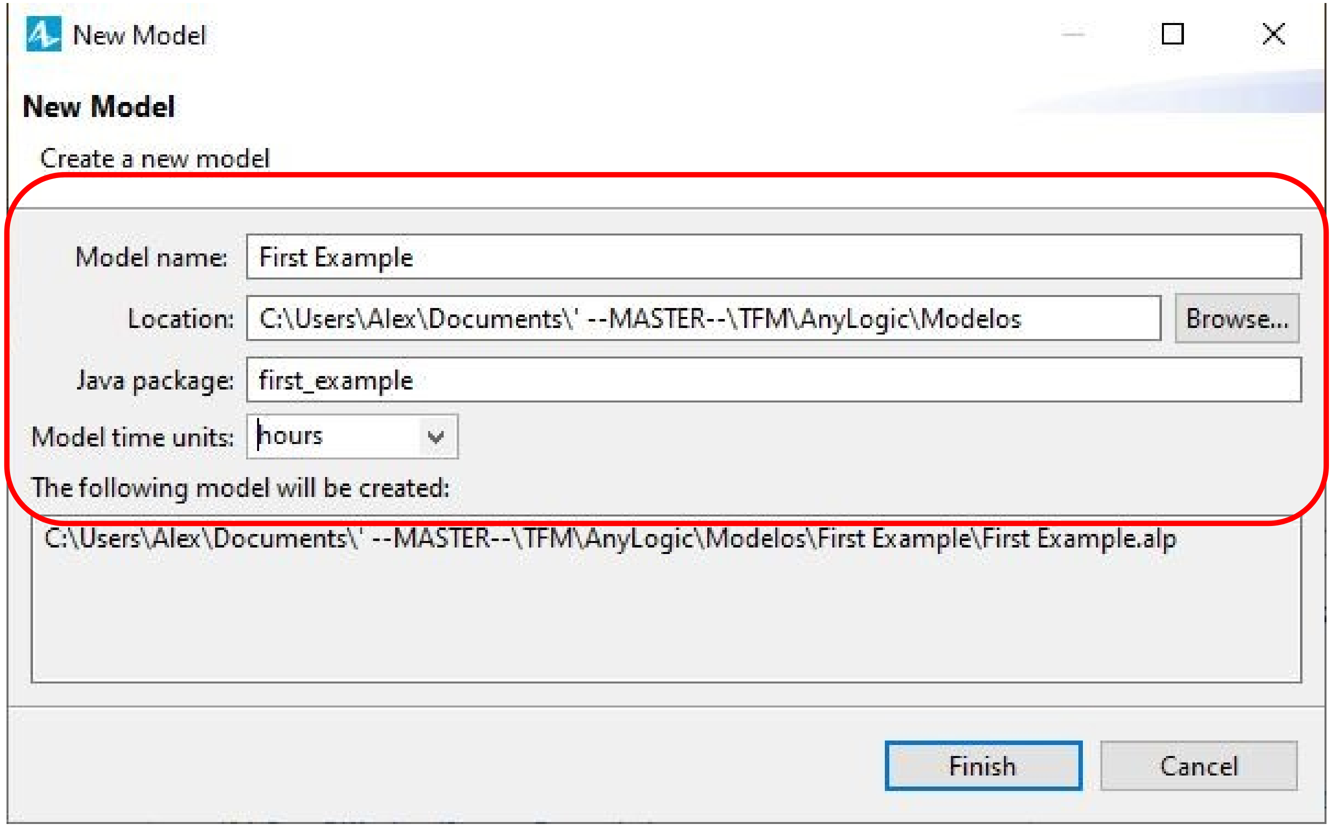
As shown on the next slides, we will start using the **Objects** from our **Libraries**, such as "Source", "Sink", "Service" and "Connectors". Drag & drop any **object** from the selected **library** to the main centre screen to add them to your model.

Select the **objects in** your model and change their **properties** according to our queue example model shown in the previous slide, and use the numbers suggested in the captions of each screenshot to come.

We will also change some names to make it easier to identify each **object**, in the following slide we can follow step by step this procedure.

4.2. Building a New Model


AnyLogic Personal Learning Edition [PERSONA
File Edit View Draw Model Tools Help



New Model

Create a new model

Model name: First Example

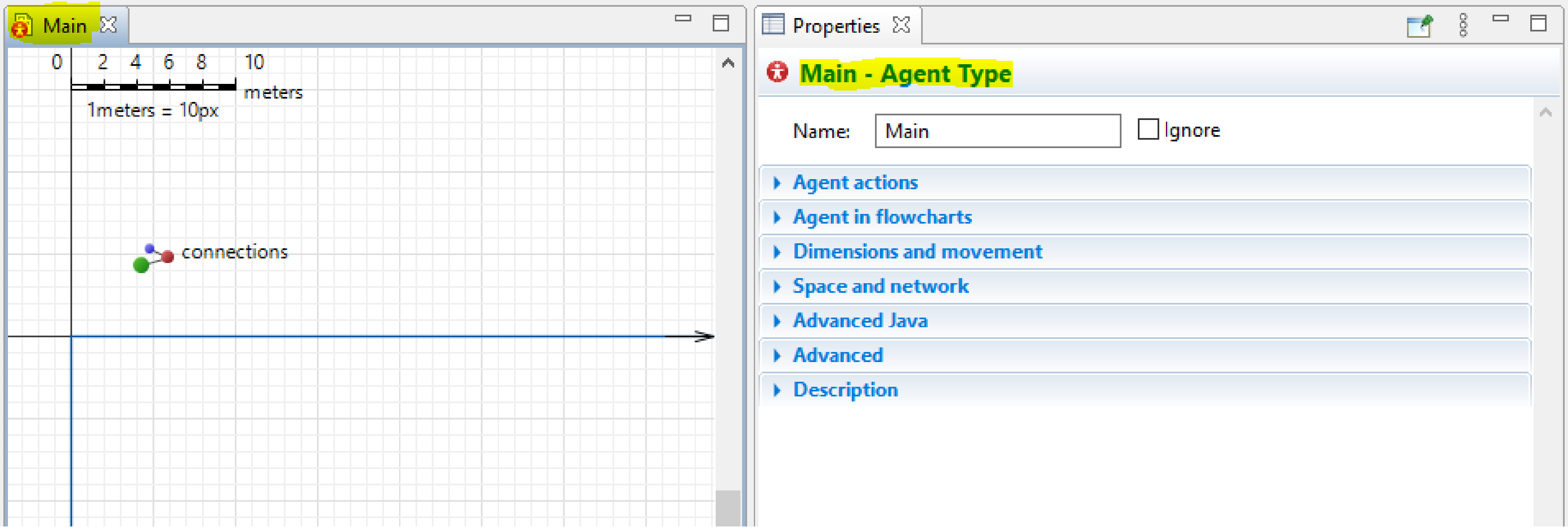
Location: C:\Users\Alex\Documents\' --MASTER--\TFM\AnyLogic\Modelos

Java package: first_example

Model time units: hours

The following model will be created:

C:\Users\Alex\Documents\' --MASTER--\TFM\AnyLogic\Modelos\First Example\First Example.alp



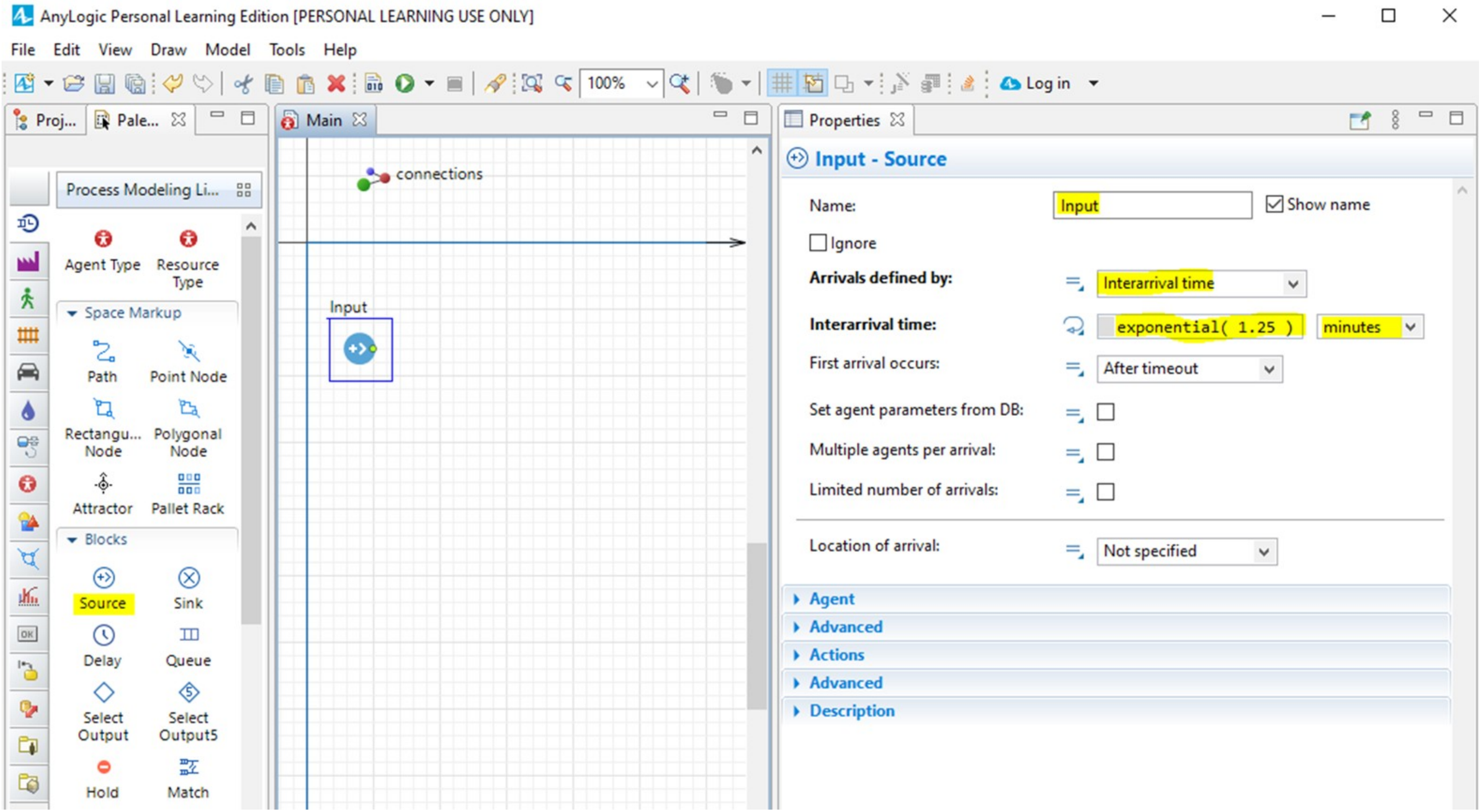
Properties

Main - Agent Type

Name: Main □ Ignore

- Agent actions
- Agent in flowcharts
- Dimensions and movement
- Space and network
- Advanced Java
- Advanced
- Description

One Agent is created by default (“Main”). Changes will be made in the Properties tab.



Input - Source

Name: Input ☑ Show name

□ Ignore

Arrivals defined by: Interarrival time

Interarrival time: `exponential( 1.25 )` minutes

First arrival occurs: After timeout

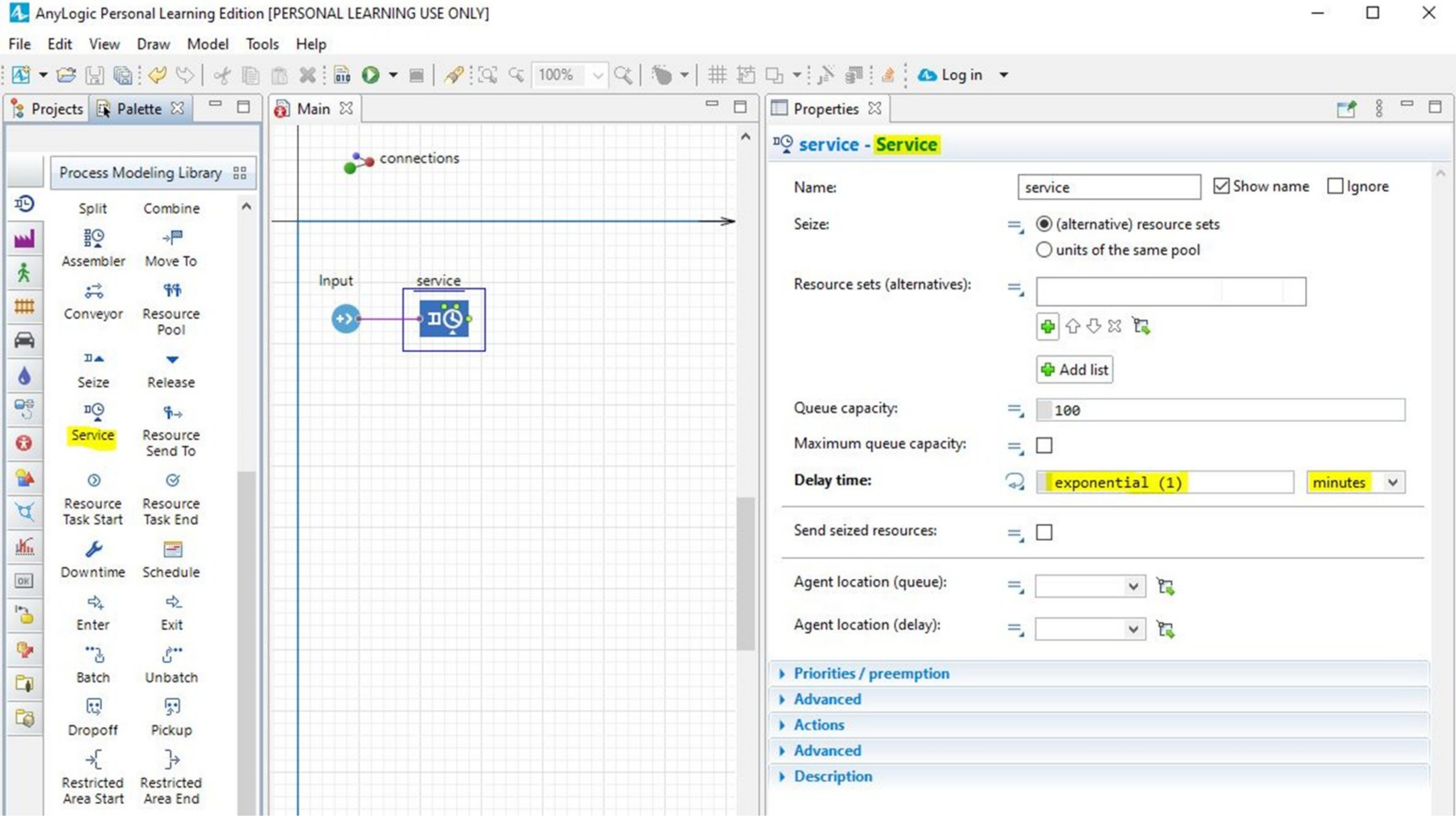
Set agent parameters from DB: □

Multiple agents per arrival: □

Limited number of arrivals: □

Location of arrival: Not specified

Add a “Source”, change it’s name, and specify an *Interarrival Time* as an **exponential distribution** with a mean time of 1.25 minutes.



AnyLogic Personal Learning Edition [PERSONAL LEARNING USE ONLY]

service - Service

Name: service ☑ Show name □ Ignore

Seize: ● (alternative) resource sets ○ units of the same pool

Resource sets (alternatives):

Queue capacity: 100

Maximum queue capacity: □

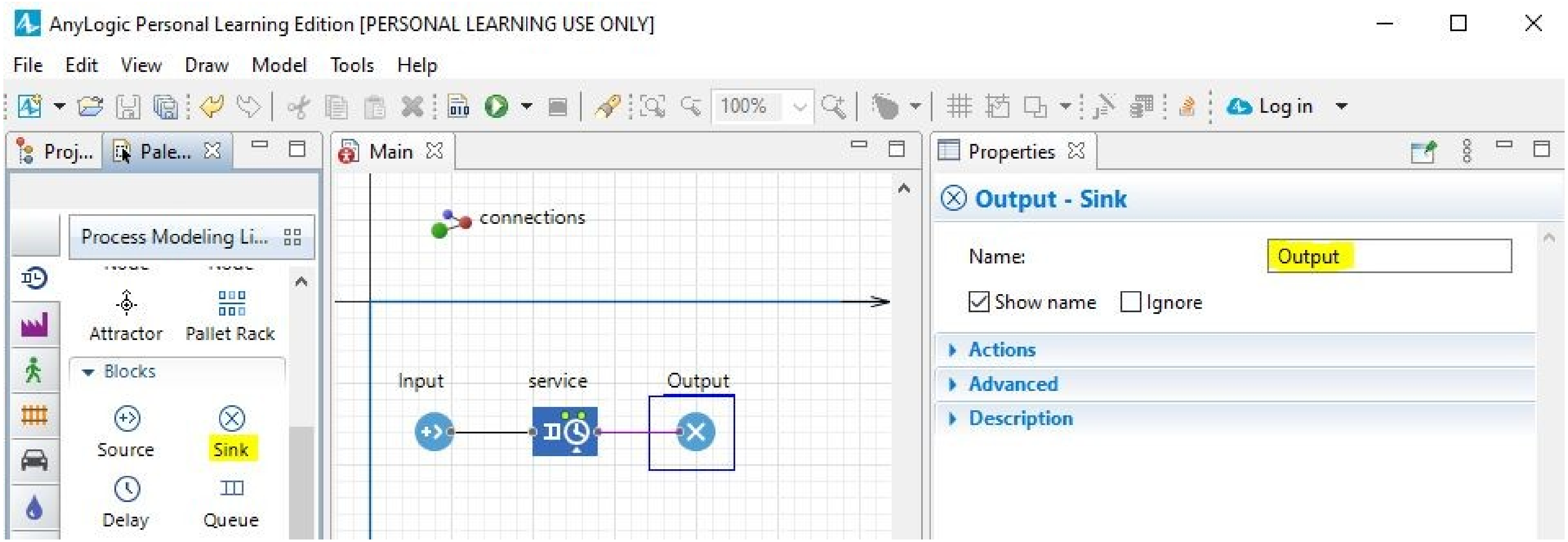
Delay time: exponential (1) minutes

Send seized resources: □

Agent location (queue):

Agent location (delay):

Add a “Service” and specify a *Delay Time* as another **exponential distribution** with a mean time of 1 minute.



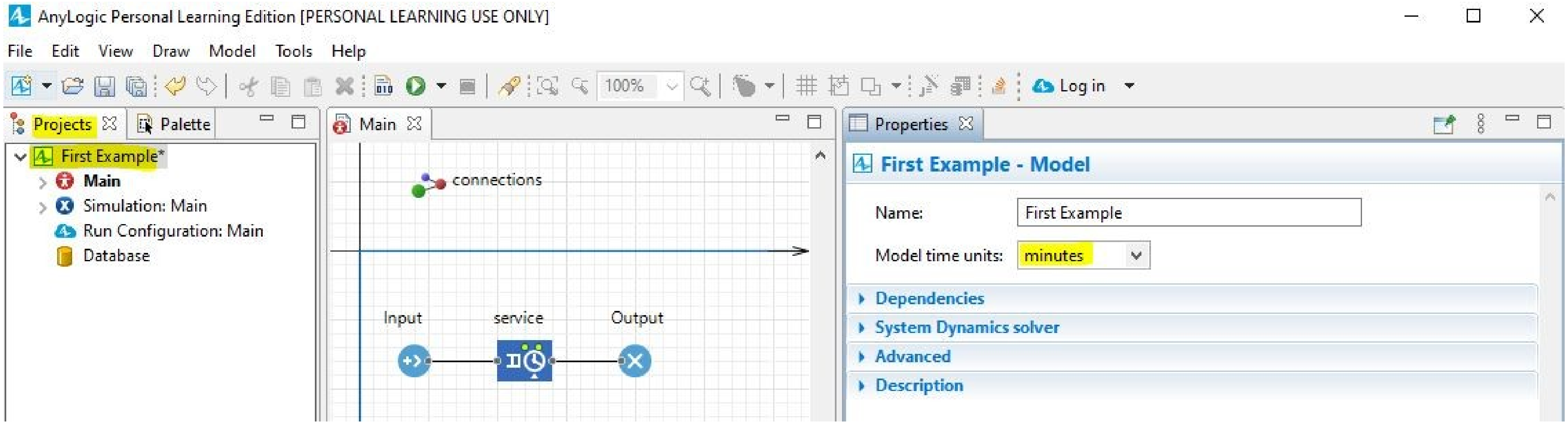
Name: Output

☑ Show name □ Ignore

- The last piece we need is a sink to send the entities that have finished the route, and can be discarded from our simulation.
- This is now a defined environment, with a start, an end, and a service between this points that simulates the theoretical distribution process time for the users/entities.
- Lets run this model so we can see where are the results such as “time in system” or other relevant information.

4.3. Execution (1/3)

Set the model time units to Minutes (if you use seconds, simulation will be slow, if you choose hours, the simulation will run fast), this depends on what are we simulating, how long can we wait for the results, and how detailed should the results be.

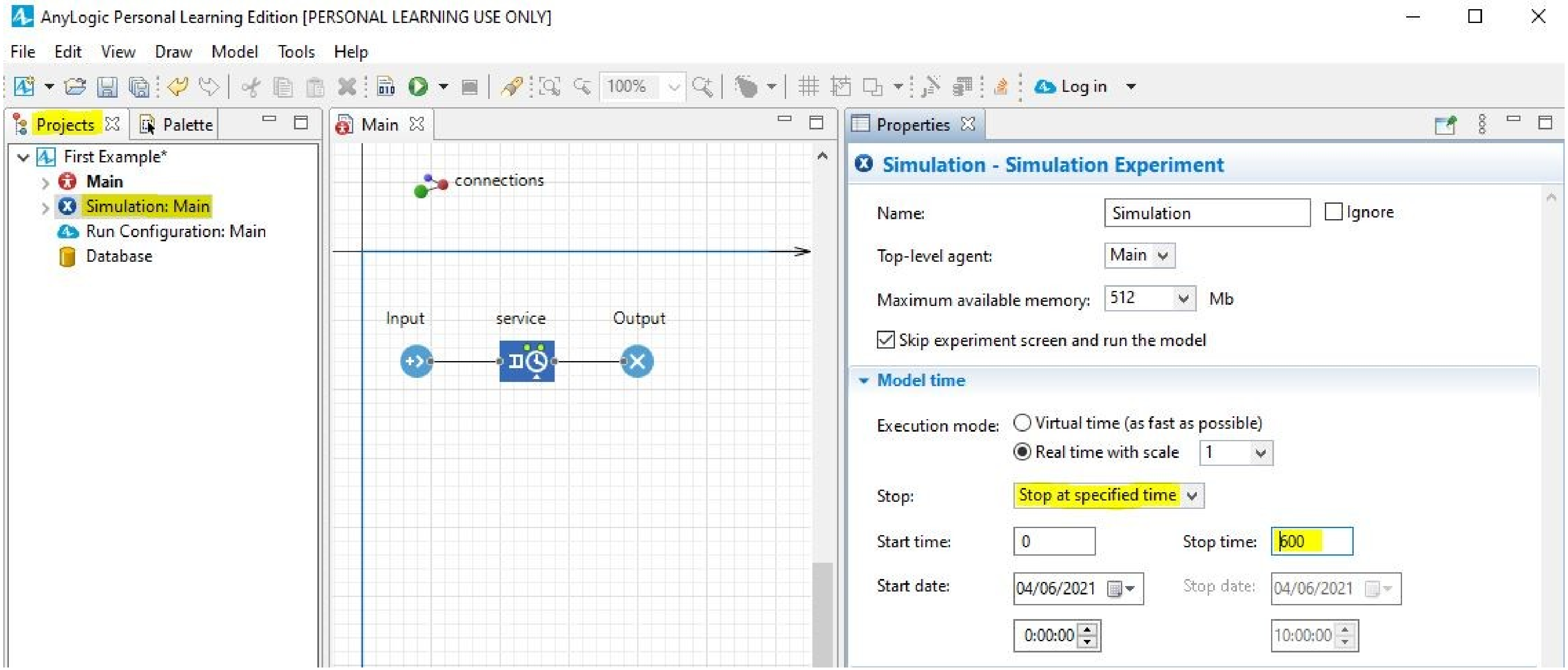


First Example - Model

Name: First Example

Model time units: minutes

4.3. Execution (2/3)



Simulation - Simulation Experiment

Name: Simulation □ Ignore

Top-level agent: Main

Maximum available memory: 512 Mb

☑ Skip experiment screen and run the model

Model time

Execution mode: ○ Virtual time (as fast as possible)
● Real time with scale 1

Stop: Stop at specified time

Start time: 0 Stop time: 600

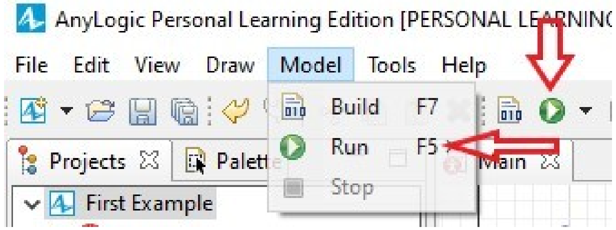
Start date: 04/06/2021 Stop date: 04/06/2021

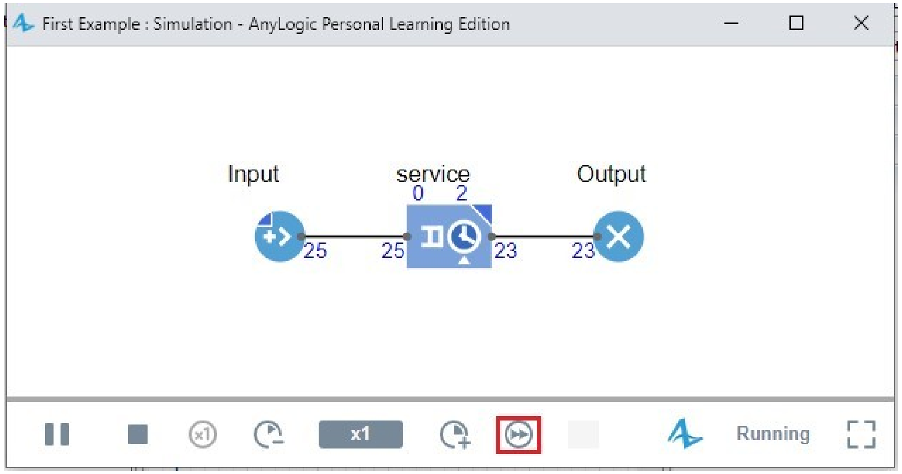
0:00:00 10:00:00

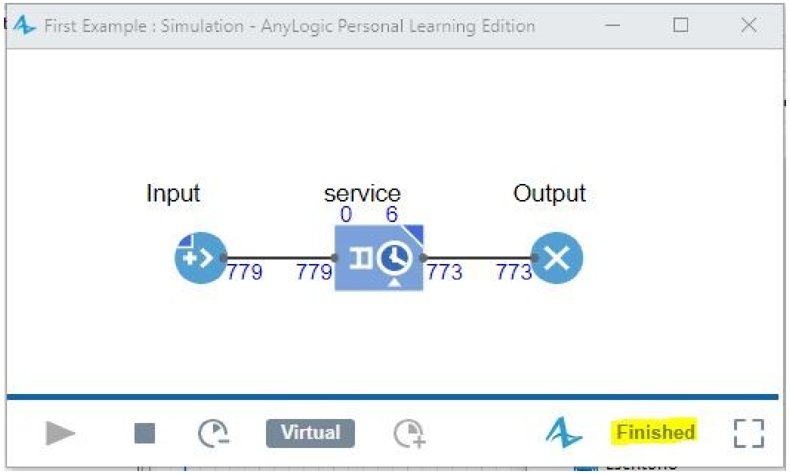
You must also remember that we don’t want the simulation to run forever, so lets change the *Simulation* **properties** so it stops after 10 virtual hours.

4.3. Execution (3/3)

You can find the “start simulation” button in 3 or 4 places, they all do the same, here we show a couple. Simulation will open a new window where we can follow the simulation at the rate we just specified.



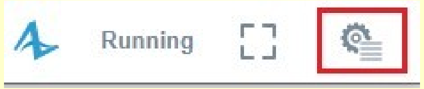

First Example : Simulation - AnyLogic Personal Learning Edition
Input
service
0
2
Output
25
25
23
23
x1
Running


First Example : Simulation - AnyLogic Personal Learning Edition
Input
service
0
6
Output
779
779
773
773
Virtual
Finished

5. Results

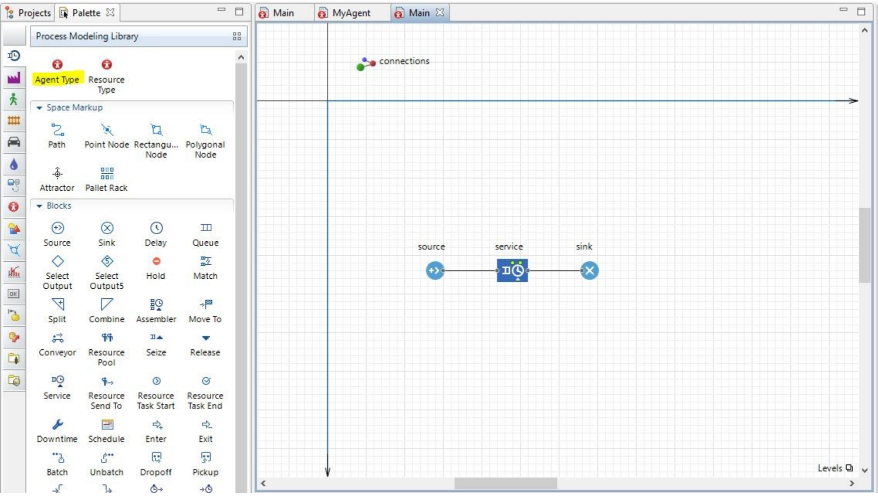
- Up to this point, simulation can be completed, but we will only see changes in the numbers as seen in the images in the previous slide, lets add a new **Agent**, and lets give it some **parameters**, so we can see more information on the simulation.

- The **simulation environment** has a special developer button, that when clicked, opens a console. Consoles are used to either input data, or show results during the simulation, here we can print our output chosen parameters/variables.

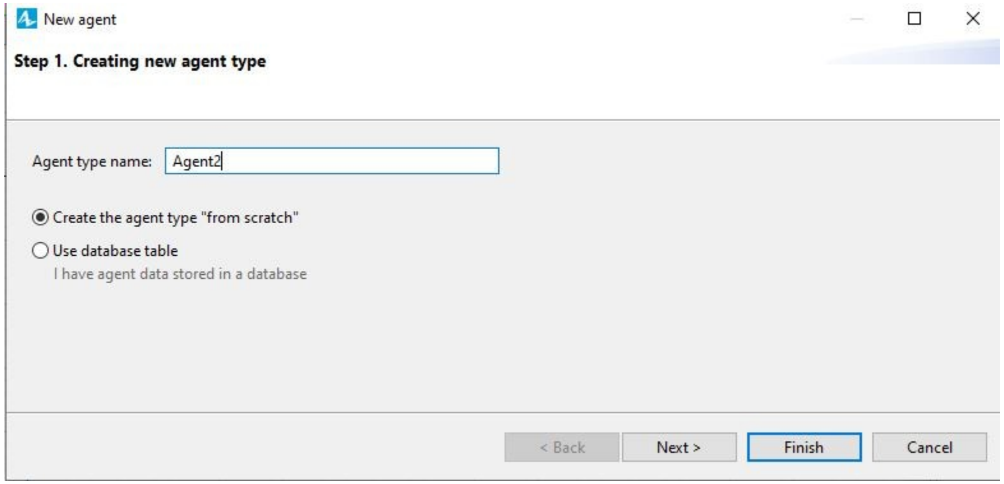


- We will make our model show an output in the console of "*time in system*" for each individual agent created in the simulation.

5.1 New Agent



Drag & Drop Agent Type


New agent

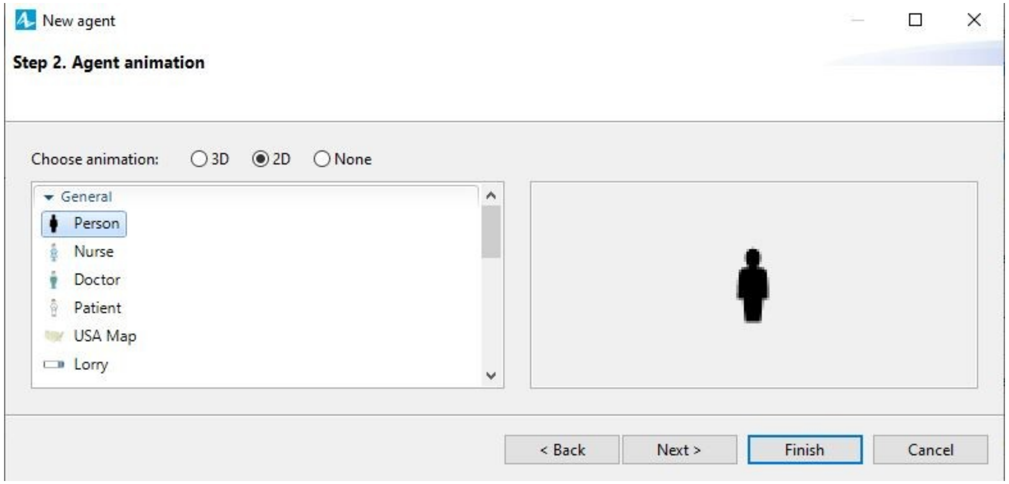
Step 1. Creating new agent type

Agent type name: Agent2

● Create the agent type "from scratch"

○ Use database table

I have agent data stored in a database


New agent

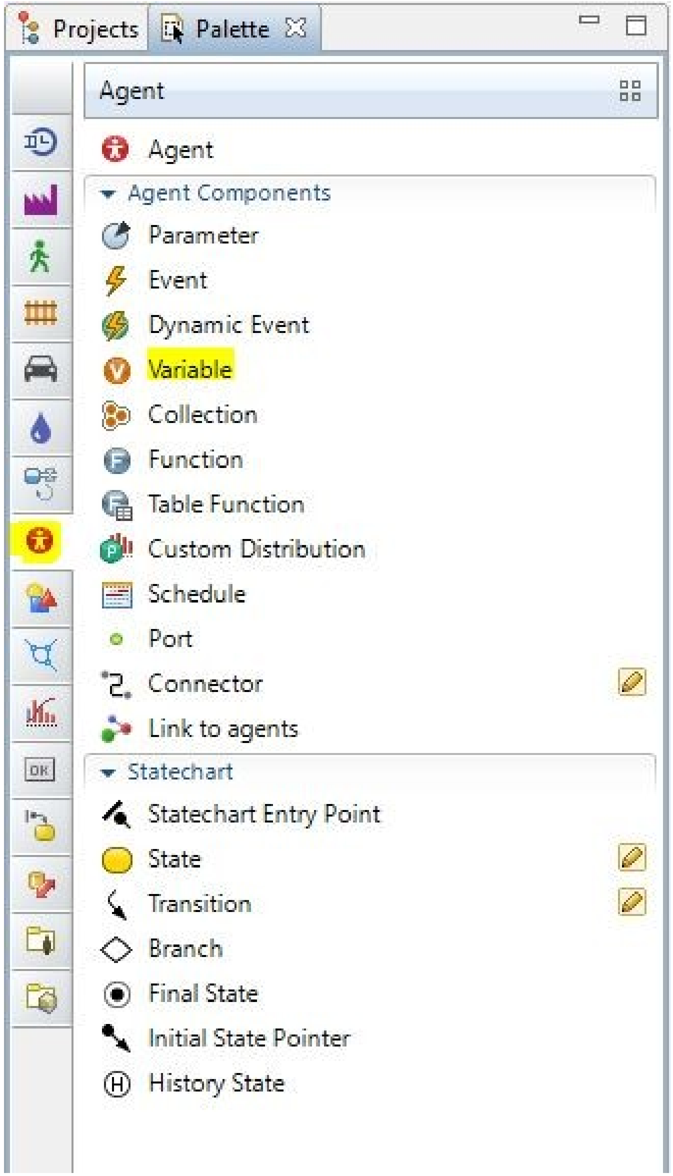
Step 2. Agent animation

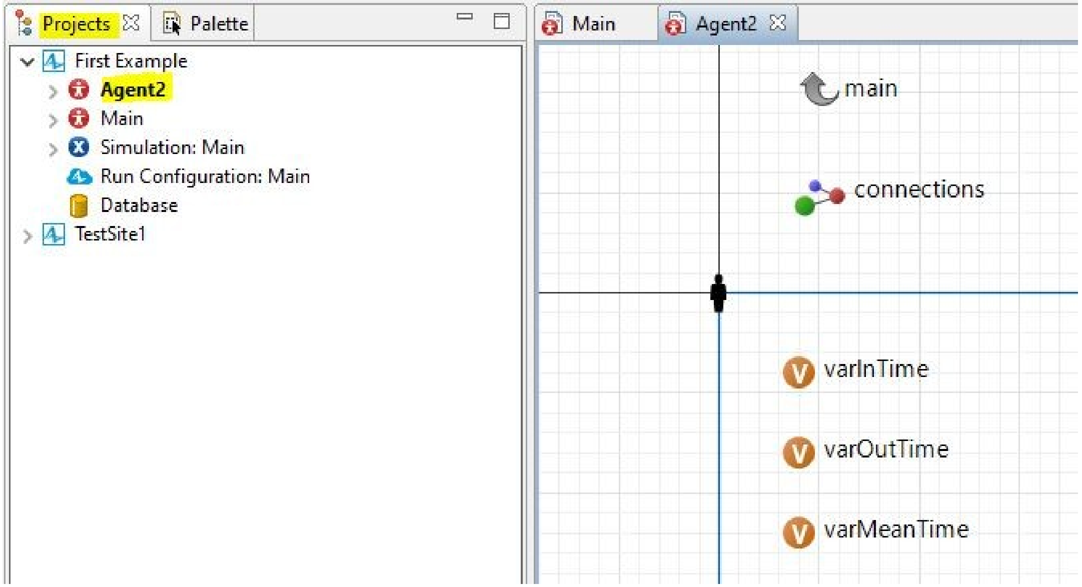
Choose animation: ○ 3D ● 2D ○ None

- General
  - Person
  - Nurse
  - Doctor
  - Patient
  - USA Map
  - Lorry

Skip step 3 for now, parameters will come later.

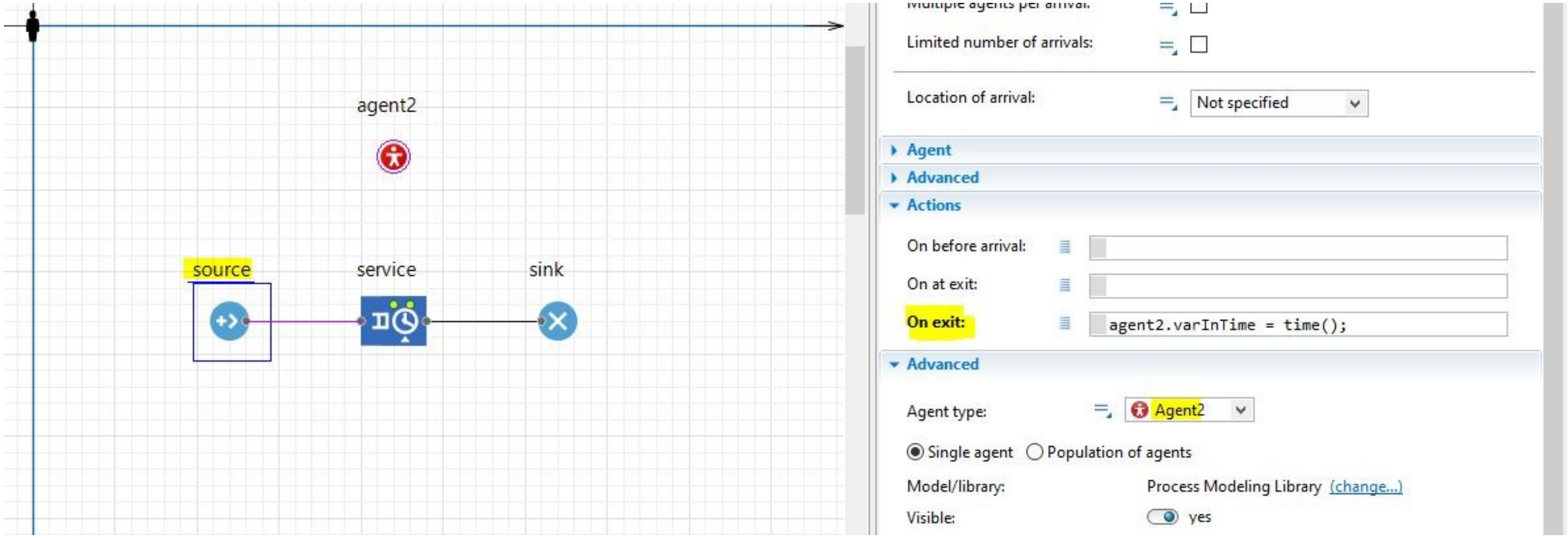
5.2 New Variables





Drag & Drop and choose the variable name and units (minutes)

5.3 Interaction Between Agents



Multiple agents per arrival: ☐

Limited number of arrivals: ☐

Location of arrival: Not specified

Agent

Advanced

Actions

On before arrival:

On at exit:

On exit:

```
agent2.varInTime = time();
```

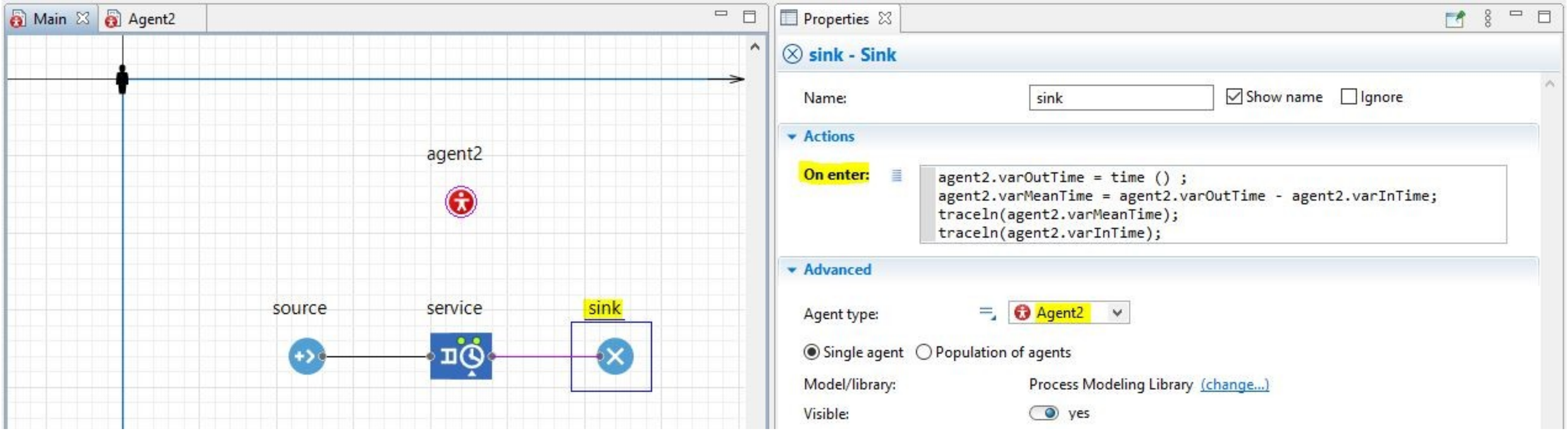
Advanced

Agent type: Agent2

● Single agent ○ Population of agents

Model/library: Process Modeling Library (change...)

Visible: yes



Properties

sink - Sink

Name: sink ☑ Show name ☐ Ignore

Actions

On enter:

```
agent2.varOutTime = time () ;
agent2.varMeanTime = agent2.varOutTime - agent2.varInTime;
traceln(agent2.varMeanTime);
traceln(agent2.varInTime);
```

Advanced

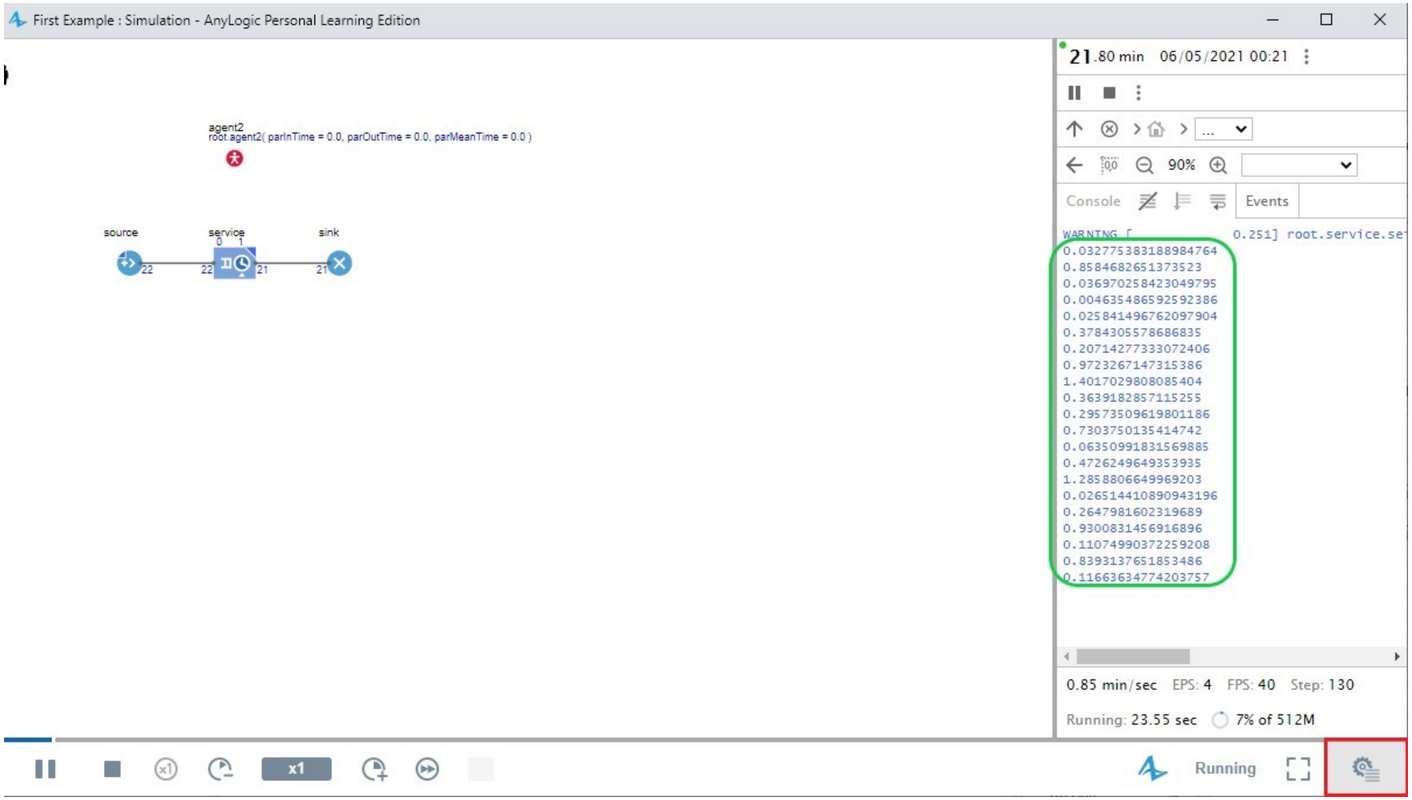
Agent type: Agent2

● Single agent ○ Population of agents

Model/library: Process Modeling Library (change...)

Visible: yes

5.4 Result Simulation


First Example : Simulation - AnyLogic Personal Learning Edition
agent2
root.agent2(parInTime = 0.0, parOutTime = 0.0, parMeanTime = 0.0)
source
service
sink
22
22
0
1
21
21
21.80 min
06/05/2021 00:21
90%
Console
Events
Running
x1

```
WARNING [          0.251] root.service.se
0.032775383188984764
0.8584682651373523
0.036970258423049795
0.004635486592592386
0.025841496762097904
0.3784305578686835
0.20714277333072406
0.9723267147315386
1.4017029808085404
0.3639182857115255
0.29573509619801186
0.7303750135414742
0.06350991831569885
0.4726249649353935
1.2858806649969203
0.026514410890943196
0.2647981602319689
0.9300831456916896
0.11074990372259208
0.8393137651853486
0.11663634774203757
```

```
0.85 min/sec   EPS: 4   FPS: 40   Step: 130
Running: 23.55 sec   7% of 512M
```

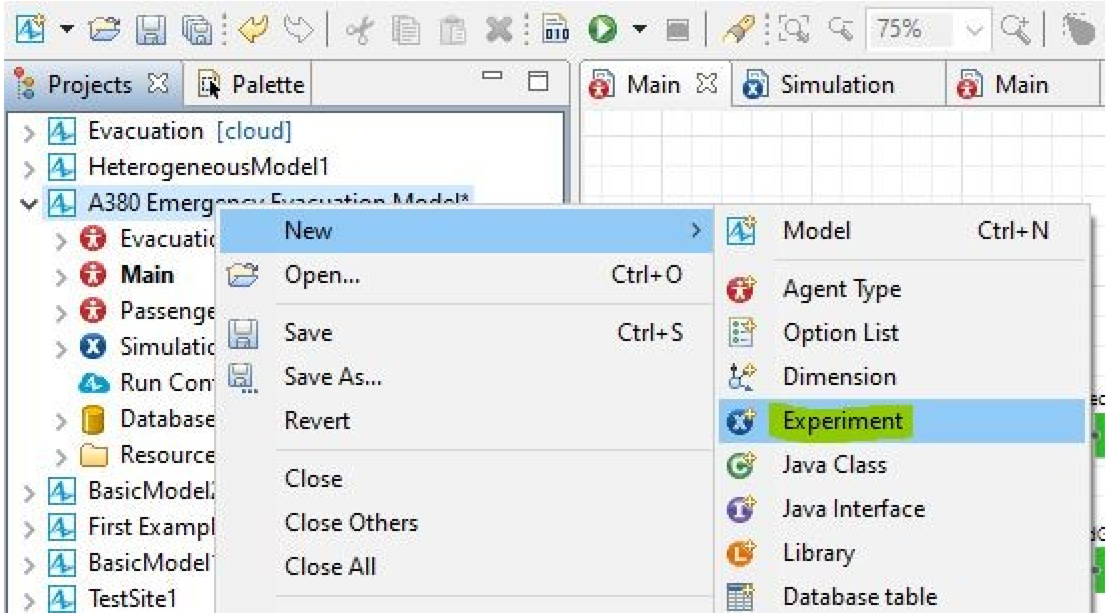
- In red, we see the button that shows us the console.
- In green we have the first results printed in the console that come from the variable "*varMeanTime*", which means that the result shown is the "mean time in system" for each agent (in minutes as specified).

If you followed all the steps, the console will also display time of entry to the system of the entity .

6. Create Experiments

- **Experiments** run several times the proposed simulation, and is used to compare results between simulations and detect bottlenecks or mistakes.

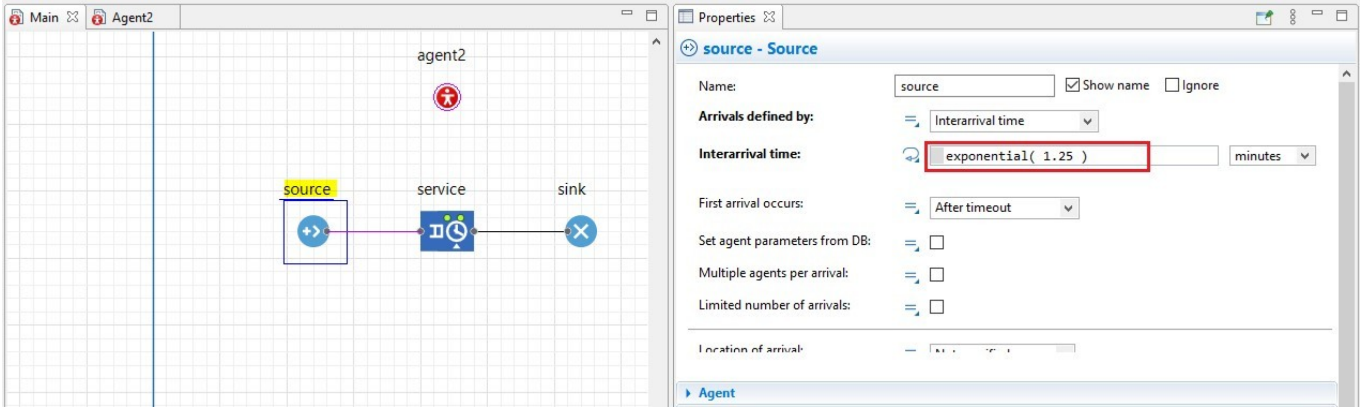
- In AnyLogic there are three types of randomness:
  - ✓ The Fixed seed (runs the same simulation repeatedly)
  - ✓ The Random seed (Introduces RNG)
  - ✓ The custom generator (not used in this slides, custom made)



7. Introducing Randomness (RNG)

Where are we using random functions? In the Source and in the Server:

But if executed twice, results will be exactly the same in the console, in the same sequence. This is because we haven't told the software to make unique or random repetitions yet.



source - Source

Name: source ☑ Show name ☐ Ignore

Arrivals defined by: Interarrival time

Interarrival time: `exponential( 1.25 )` minutes

First arrival occurs: After timeout

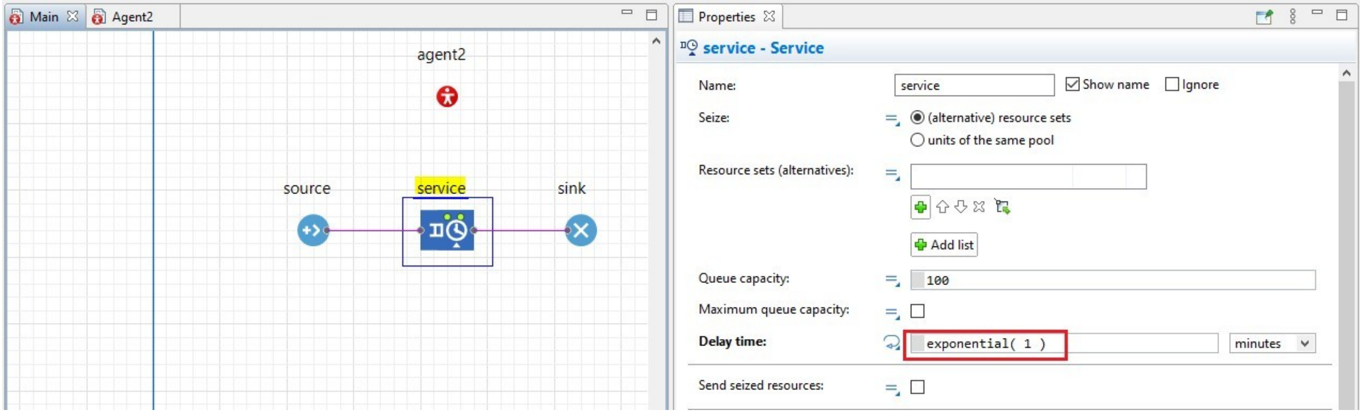
Set agent parameters from DB: ☐

Multiple agents per arrival: ☐

Limited number of arrivals: ☐

Location of arrival:

Agent



service - Service

Name: service ☑ Show name ☐ Ignore

Seize: ● (alternative) resource sets ○ units of the same pool

Resource sets (alternatives):

Add list

Queue capacity: 100

Maximum queue capacity: ☐

Delay time: `exponential( 1 )` minutes

Send seized resources: ☐