

Creating and registering a new SimulationExecutionListener

A *SimulationExecutionListener* Class is a listener for Events that will be activated during execution of a model. All available Events are listed below.

- An execution is started.
- An element is activated.
- An element is deactivated.
- A signal event is triggered.
- An operation is called.
- A behavior is called.
- A runtime object is created.
- An execution is terminated.

```
public class SimulationExecutionListener {
    void executionStarted(SimulationExecution execution);
    void elementActivated(Element element, Collection<?> values);
    void elementDeactivated(Element element, Collection<?> values);
    void eventTriggered(SignalInstance signal);
    void operationCalled(Operation operation, ParameterValueList pvl, Object_ caller, Object_
target, boolean isSynchronous);
    void behaviorCalled(Behavior behavior, ParameterValueList pvl, Object_ caller, Object_ target,
boolean isSynchronous);
    void objectCreated(Object_ sender, Object_ object);
    void valueChange(StructuredValue context, FeatureValue feature, Object oldValue, Object
newValue);
    void executionTerminated(SimulationExecution execution);
    void configLoaded(Element config);
    void busyStatusChange(StructuredValue context, Object oldValue, Object newValue);
}
```

Once you have created the execution listener, you can register it to a list of global execution listeners.

To register a new execution listener to a global list, type the following code

```
SimulationManager.registerSimulationExecutionListener(listener); //listener is an instance of
SimulationExecutionListener.
```

All registered listeners will be cleared and removed automatically when the execution is terminated.



Note

ExecutionAdapter and *ExecutionListener* are not valid in Cameo Simulation Toolkit 19.0 and in later versions.