

# Лабораторная работа № 0-1

## Имитационное моделирование в среде AnyLogic. Общие понятия.

AnyLogic обеспечивает поддержку всех этапов имитационного моделирования: для различных типов динамических моделей – дискретных, непрерывных и гибридных, детерминированных и стохастических. Создание модели, ее выполнение, оптимизация параметров, анализ полученных результатов, верификация модели – все эти этапы удобно выполнять в среде AnyLogic. Этот инструмент обладает большим спектром разнообразных возможностей проведения как отдельных прямых экспериментов типа “*if-then*”, так и серий таких экспериментов для решения разнообразных обратных задач. Удобный интерфейс и разнообразные средства поддержки разработки в AnyLogic делают не только использование, но и создание компьютерных имитационных моделей в этой среде моделирования доступными даже для тех, кто в области вычислительной техники и программирования не является профессионалом.

### 1. Моделирование в AnyLogic

AnyLogic используется для разработки имитационных исполняемых моделей и последующего их прогона с целью их анализа. Разработка модели выполняется в графическом редакторе AnyLogic с использованием многочисленных средств поддержки, упрощающих работу. Построенная модель затем компилируется встроенным компилятором AnyLogic и запускается на выполнение. В процессе выполнения модели пользователь может наблюдать ее поведение, изменять параметры модели, выводить результаты моделирования в различных формах и выполнять разного рода компьютерные эксперименты с моделью.

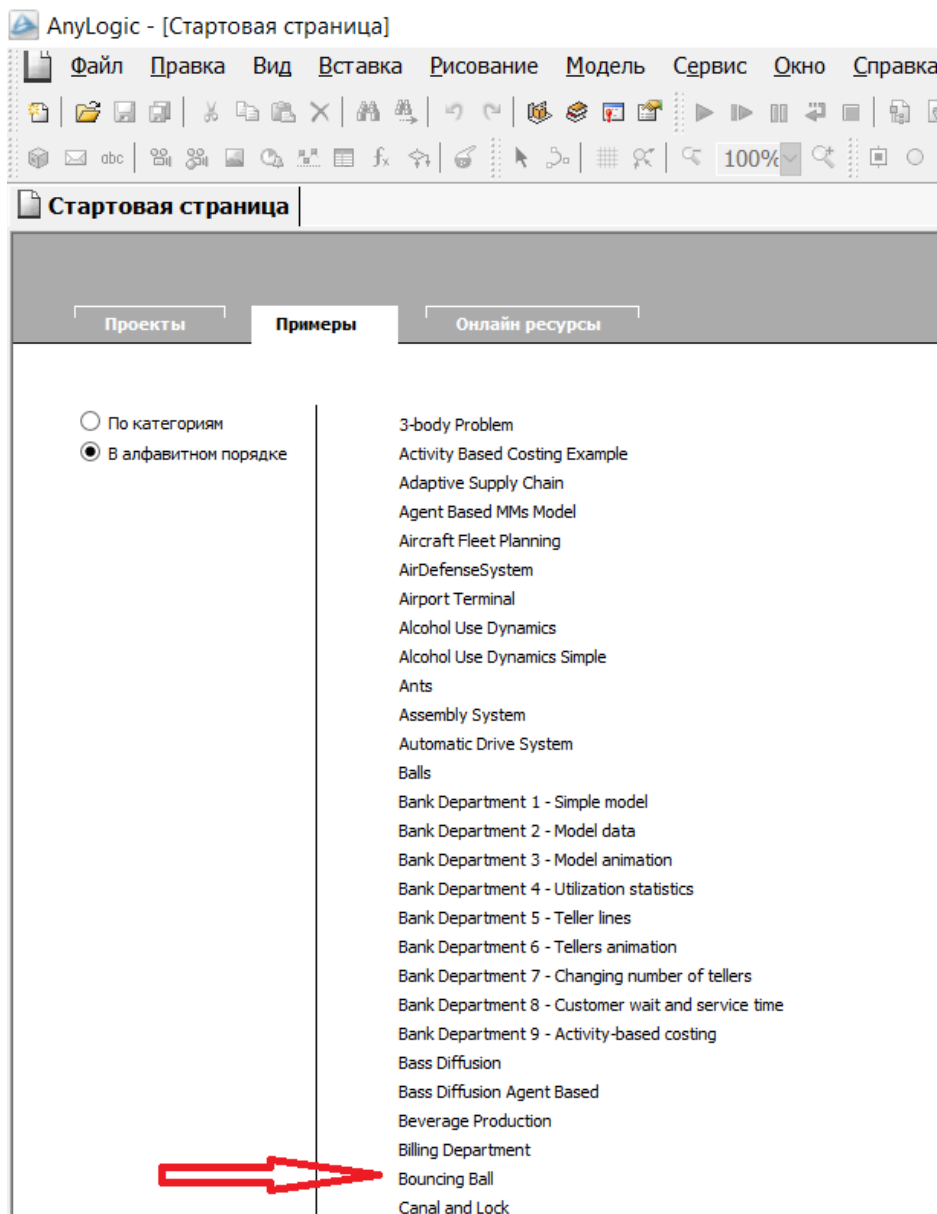
Графический редактор AnyLogic позволяет разработчику модели описывать структуру моделируемой системы, ее подсистемы и поведение объектов в модели графически, выполняя генерацию соответствующего программного кода автоматически. Графическая разработка легче, быстрее, понятнее, чем написание программных текстов.

В данной главе мы начнем с простейшей модели, уже разработанной в AnyLogic, чтобы составить первое представление о структуре инструмента и его функциях на этих фазах работы с моделью. Главная цель этой главы – научиться работать с окнами редактора и исполнителя модели, запуская уже разработанные модели и внося в них некоторые изменения.

## 2. Модель “прыгающий мячик”

### Открытие проекта

Проанализируем простую модель *Bouncing Ball*, созданную для имитации поведения прыгающего мяча. Для открытия этого проекта в AnyLogic выберете в примерах *Bouncing Ball*.



На экране появится следующее окно (рис.1).

AnyLogic при открытии проекта всегда открывает среду разработки моделей – графический редактор. Рис.1 показывает основные составляющие пользовательского интерфейса этого редактора.

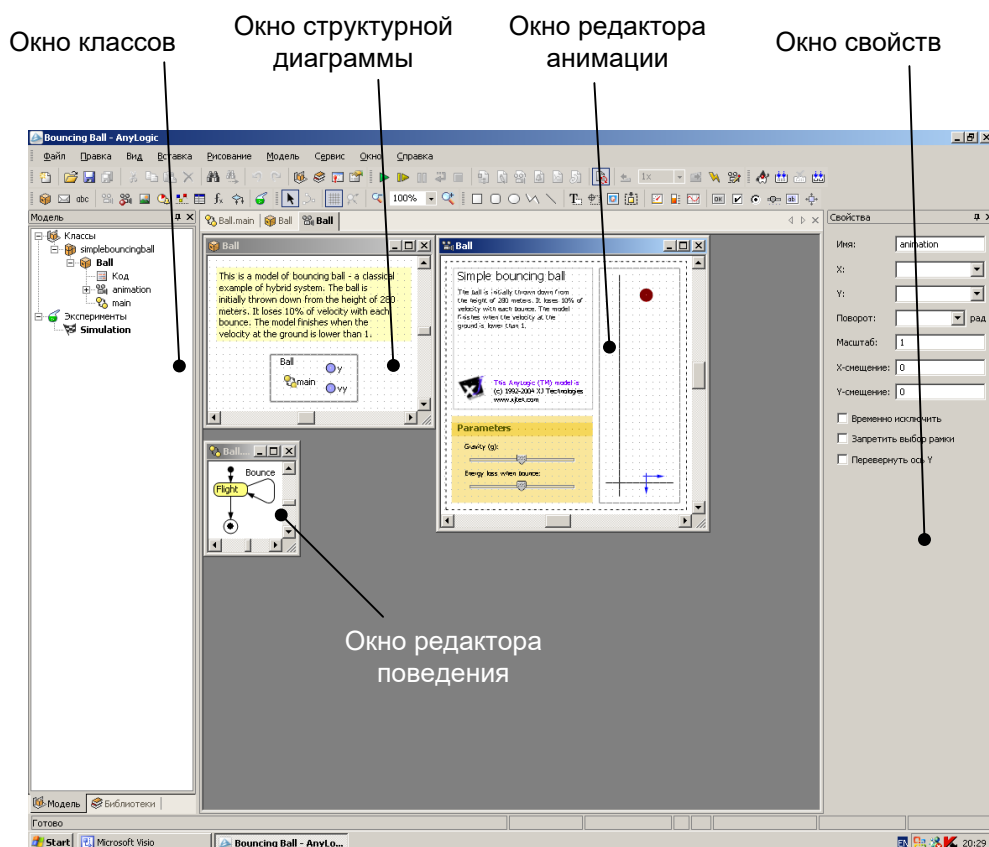



Рис.1. Окно редактора AnyLogic

## Структурная диаграмма

При построении модели нужно задать ее структуру (т.е. описать, из каких частей состоит модель системы) и поведение отдельных объектов системы. В AnyLogic структурными элементами модели является так называемый *активный объект*. Активный объект имеет структуру и поведение. Элементы структуры - это другие активные объекты, включенные как составные элементы данного активного объекта, и связи, которые существуют между включенными активными объектами. Структура активного объекта задается графически в специальном окне редактора – структурной диаграмме. Поведение, представленное в своем окне (окне редактора поведения), определяет реакции активного объекта на внешние события – логику его действий во времени. Кроме того, в дополнительном окне редактора анимации можно построить анимацию активного объекта.

На рис.1 для нашего примера структура единственного объекта модели – мяча - задается в окне с именем *Ball* прямоугольником, внутри которого содержатся его переменные (координата  $y$  и скорость  $vy$ ) и иконка поведения  с именем *main*. Наш простой объект не содержит никаких включенных в него объектов, поэтому в прямоугольнике с именем *Ball* нет никаких других вложенных прямоугольников и связей между ними. Рядом с прямоугольником *Ball* на желтом поле содержится поясняющий текст – комментарий.

## Окно поведения активного объекта

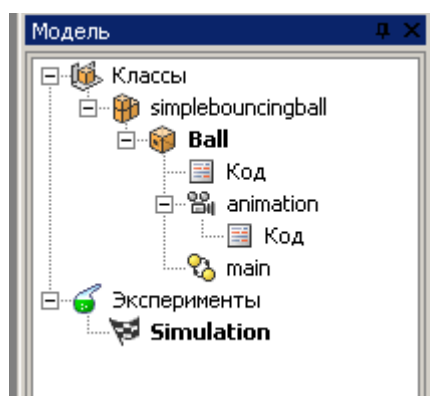
Поведение мяча представлено в окне *Ball.main*, которое содержит простейшую “*карту состояний*” (или стейтчарт - “*statechart*”) с двумя состояниями и двумя переходами. Стейтчарты – это модифицированные графы переходов конечного автомата, одной из самых простых и в то же время мощных формальных моделей, описывающих дискретное поведение систем во времени с помощью визуального представления состояний системы и переходов между ними под воздействием событий или условий. В последнее время стейтчарты доказали свое удобство как средство графического описания поведения весьма сложных систем. Стейтчарт модели прыгающего мяча состоит из одного состояния и перехода, ведущего из этого состояния в него же.

## Окно редактора анимации активного объекта

В этом окне для модели строится двумерное или трехмерное анимационное представление, которое помогает понять, что происходит с моделью с течением времени. Для данной модели в окне анимации построено изображение прыгающего мяча, представленного закрашенным кругом. Элементы анимационной картинке имеют свои параметры (для круга это, например, его координаты центра), которые могут быть связаны с переменными и параметрами модели. Изменение переменных модели во времени ведет к изменению графического образа что позволяет пользователю наглядно представить динамику моделируемой системы с помощью динамически меняющейся графики. В нашем примере координата *y* центра круга связана с переменной *y* активного объекта *Ball*, и таким образом изменение переменной модели связывается на анимационной картинке с перемещением мяча.

## Окно классов

Главное окно редактора содержит также окно классов, показывающее дерево всех объектов проекта:

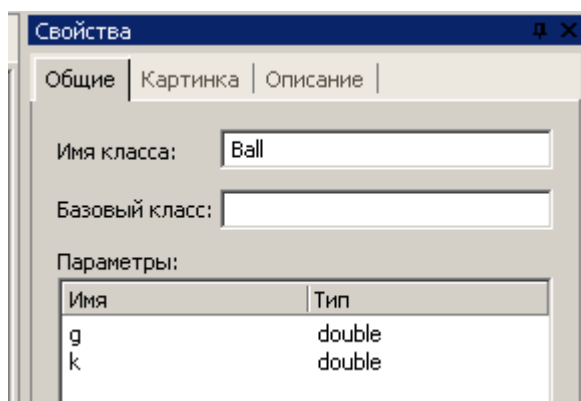


В нашем примере проект *simplebouncingball* содержит один класс активных объектов *Ball*, составляющими которого являются его код, анимация с именем *animation* (со своим кодом) и стейтчарт с именем *main*. Один из объектов в дереве классов имеет название *Эксперименты*, он объединяет группу компьютерных экспериментов, которые могут быть выполнены с

моделью. В открытом нами проекте “прыгающий мяч” в группу экспериментов входит только один эксперимент с именем **Simulation**, выделенный жирным шрифтом – это *текущий* эксперимент, именно в соответствии с установленными в текущем эксперименте параметрами будет происходить выполнение модели после ее компиляции.

## Окно свойств

В окне свойств редактора AnyLogic для каждого выделенного элемента модели указываются его свойства (параметры). При выделении какого-либо элемента в любом из окон редактора (в окне структуры, окне поведения, окне анимации или в окне классов) справа появляется окно свойств, показывающее параметры именно этого выделенного элемента. Выделить элемент можно щелчком левой кнопкой мыши на нем. Например, при выделенном окне редактора структуры объекта *Ball* окно свойств содержит три вкладки, *Общие*, *Картинка* и *Описание*. Во вкладке *Общие* кроме имени этого объекта указываются его параметры:




Щелкните мышкой на нескольких элементах окон редактора (на переменных и на поле окна редактора структуры, на переходах и состояниях окна поведения, на графических элементах окна редактора анимации, на объекте **Simulation** дерева классов проекта и т.п.). Вы увидите, что для каждого элемента модели окно свойств имеет свою структуру и содержит специфическую информацию и параметры, характеризующие именно данный элемент. Если выделено несколько элементов, окно свойств не показывает ничего.

Например, кликните мышкой в окне редактора структуры на поясняющем тексте, расположенном на желтом фоне. В окне свойств для текста будет представлено только один параметр – сам текст. Его можно редактировать и наблюдать, как в поле текста окна редактора структуры этот текст изменяется.

Модель “прыгающий мяч” уже построена и готова к запуску.

## Запуск модели

Для запуска модели кликните кнопку  на панели инструментов. Этим действием запустится компилятор, который построит исполняемый код модели в языке Java, оттранслирует его и затем запустит модель на исполнение. При этом откроется окно наблюдения (*viewer*) (рис.2). Окно наблюдения в данном проекте включает несколько окон:

- (a) окно переменных и параметров, в котором в дереве с корнем *root* можно наблюдать мгновенные значения всех переменных и параметров ( $v$ ,  $y$ ,  $g$  и  $k$ ),
- (b) ожившее окно анимации с прыгающим мячом,
- (c) окно структуры,
- (d) окно поведения с подсвеченным красным цветом тем состоянием, в котором в данный момент находится моделируемый объект,
- (e) два окна графиков, которые показывают изменение переменных объекта ( координаты  $y$  и скорости  $v$ ) в модельном времени.

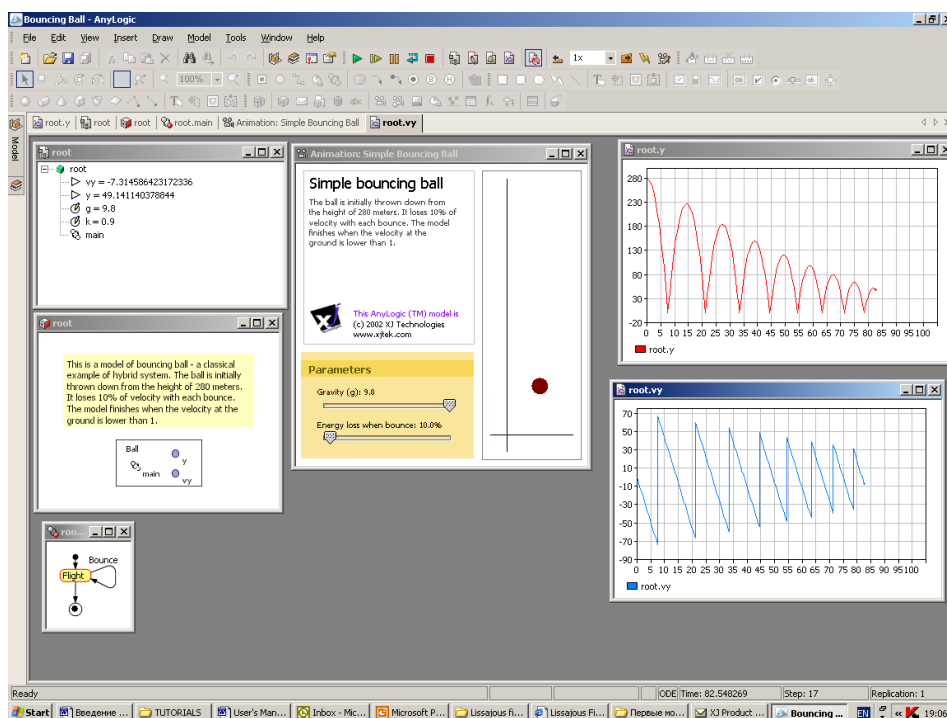







Рис 2. Окно наблюдения проекта Bouncing Ball

Анимация в *AnyLogic* создается в виде динамических графических объектов, которые дают возможность наглядно представить динамику моделируемой системы, т.е. поведение ее во времени. Средства анимации позволяют пользователю легко создать виртуальный мир (совокупность графических образов, мнемосхему и т.п.), управляемый динамическими параметрами модели по законам, определенным пользователем с помощью уравнений и логики моделируемых объектов. Поэтому окно анимации можно назвать “экспериментальным стендом” для проведения компьютерного эксперимента с моделью.

## Эксперименты с моделью



В данном примере в окне анимации кроме движущегося изображения мяча можно видеть текстовый комментарий и так называемые “слайдеры” или “бегунки” – подвижные указатели для изменения параметров модели во время ее выполнения. Двигая слайдеры, можно менять в этой модели два параметра – ускорение свободного падения  $g$  и долю  $k$  потери энергии прыгающим мячом при каждом отскоке. Изменение параметров позволяет исследовать поведение модели в различных условиях – это и есть компьютерный эксперимент.

Изменение параметров и переменных возможно также и без слайдеров. Двойной щелчок мыши на переменной или параметре в окне с именем *root* при остановленном выполнении модели вызывает окно с текущим значением этой переменной, которое можно редактировать. Последующий запуск приведет к продолжению выполнения модели уже с измененным значением параметра.

Для проведения компьютерных экспериментов необходимо использовать (кроме уже известной кнопки  компиляции и запуска модели на выполнение) также кнопки запуска выполнения модели по шагам , останова , повторного запуска с исходными начальными условиями , а также кнопку разрушения скомпилированной модели и возврата в редактор . Вместо использования этих кнопок можно в основном меню *Модель* выполнить команды *Запустить*, *Выполнить шаг* и т.п.

Проведите несколько экспериментов с моделью, изменяя параметры и переменные модели либо слайдерами, либо вызывая эти параметры двойным кликом для изменения в окне *root*. При работающей модели поле анимации можно двигать, нажав на нем правой кнопкой мыши.

## Управление скоростью выполнения модели

В AnyLogic скорость выполнения модели может быть установлена максимальной (и модель будет выполняться в режиме виртуального времени, с максимально возможной скоростью выполнения соответствующего программного кода) либо соответствующей по возможности реальному физическому времени с некоторым коэффициентом (единица модельного времени равна одной секунде реального). Переключение с виртуального времени на реальное и наоборот осуществляется кнопкой  панели инструментов, а уменьшение коэффициента ускорения модельного времени относительно реального выполняется с помощью двух кнопок и расположенного между ними окна  на этой панели. Окно указывает коэффициент ускорения модельного времени относительно реального (здесь 1x означает единичный коэффициент ускорения).

Проведите несколько экспериментов с различными скоростями выполнения модели, используя кнопки останова, рестарта, запуска.

## Предварительно определенные эксперименты с моделью

Запуск модели на выполнение производится в AnyLogic в соответствии с некоторым набором ограничений на переменные и значений параметров модели, а также с некоторыми дополнительными установками (например, точность, шаг численных методов и т.п.). Совокупность всех установок для проведения компьютерного эксперимента с моделью называется в AnyLogic “экспериментом”. Все эксперименты, возможные для выполнения в данном проекте, представлены как элементы группы (корня) с именем *Эксперименты* в окне классов проекта. Один такой эксперимент с названием *Simulation* уже построен при



создании нового проекта с выбранными по умолчанию установками, он и выбран в качестве текущего (название текущего эксперимента показано жирным шрифтом).


Установка значений параметров, реального либо виртуального времени при выполнении эксперимента, условие прекращения выполнения эксперимента и многое другое, относящееся к проведению эксперимента можно до запуска модели установить в окне свойств объекта ***Simulation***, являющегося в данном проекте единственным элементом группы *Эксперименты* в дереве классов модели. В окне *Свойства* объекта ***Simulation*** вы можете увидеть эти возможности и поменять установки, прежде чем запустить модель. Например, во вкладке *Дополнительные* установите условие остановки выполнения модели по времени 100 единиц модельного времени.

Для одной и той же модели в AnyLogic можно определить несколько различных экспериментов на этапе построения модели.

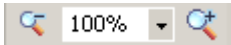
### 3. Работа с окнами


Работать с окнами при разработке и исследовании модели требуется постоянно. Поэтому необходимо уметь открывать нужные окна, изменять их размеры, закрывать, сворачивать и разворачивать.

Вновь откройте редактор проекта *Bouncing Ball*. В редакторе и при работе модели открытие и закрытие окна дерева классов и окна свойств выполняется соответствующими кнопками  и  панели инструментов. Кроме того, открыть эти окна можно выбрав в главном меню команды *Вид|Модель и Вид|Свойства* соответственно. Откройте и закройте эти окна несколько раз в редакторе и в окне наблюдения при выполнении модели.


Окна структуры, поведения или анимации можно открыть двойным кликом мыши на именах соответствующих объектов дерева объектов в окне классов, если они закрыты. Попробуйте закрыть  и открыть несколько раз каждое из окон редактора.

С любым объектом (графиком, переменной, активным объектом в окне редактора и т.п.) в редакторе AnyLogic связано контекстное меню, которое появляется при щелчке правой кнопкой мыши на этом объекте.

Размеры окон можно менять. Для помещения изображения в центр окна после изменения размеров окна в его контекстном меню выберите команду *Перейти в центр*. Масштаб изображения в этих окнах регулируется кнопками и окном  панели инструментов, а также командой *Показать все* контекстного меню. Для каждого из окон редактора сделайте его активным, щелкнув на нем мышью, измените его размеры, пользуясь правой кнопкой мыши, поместите изображение в центр нового окна, измените масштаб изображения в ту и другую сторону.

Запустите модель на выполнение по шагам (). В появившемся окне наблюдения можно закрыть и затем открыть окно (с предопределенным именем *root*) экземпляра корневого класса модели, показывающего текущие значения всех переменных и параметров модели (командой



*Вид* | *Корневой объект модели* или кнопкой  панели инструментов). Двойной клик на переменной или параметре в этом окне вызовет появление диалогового окна модификации этого объекта. Тот же эффект вызовет и команда *Изменить* контекстного меню этого объекта.

Команда *Вид* | *Анимация* откроет окно анимации модели, если оно закрыто. Заметьте, что закрытие окон графиков уничтожает их, они не сохраняются и их нужно будет снова создавать.