ALICE

Three Dimensional
Event-driven
Interactive
Animations

Object Oriented Language
using 4 GL Drag and Drop methodology
What is an Object?

1. Objects have properties … the qualities of the object
   - location in space
   - orientation
   - color, transparency
   - visibility

2. Objects have methods which change some of their properties
   - move
   - turn
   - change color
   - become invisible or visible

3. Objects have methods which can tell other objects the value of one its properties. In Alice these are called functions.
   - how far away are you?
   - what color are you?
   - what direction are you facing?
What is an Object?
- continued -

4. Objects are often composed of sub-objects. Sub-objects inherit properties of the parent object

5. A Class of objects is a “generic type” of the object. We create specific instances of classes … and call each instance an object. For example if we have a penguin class, we can create penguin1, penguin2 etc.

6. You may begin with a “pre-built” object and modify it’s properties, methods and functions. You may then save the object. It becomes a class, or a template, from which you may create new objects which “inherit” the properties, methods and questions of the class. Be sure such objects are entirely self-contained (i.e. do not call other objects which may not exist in a different world)
Object Oriented Programming

• The system starts by the programmer putting objects into a world.
  • They may be “pre-built” objects or
  • Objects built by the programmer

• The animation consists of objects
  • responding to events (from the keyboard or mouse) or
  • interacting with each other.

• Objects respond to events or interact with other objects through their methods. Invoking an object’s method causes it to change one of its properties.
Methods

• Methods cause an object to change its state – to change one of its properties

• When one object “sends a message” to another object it does so by invoking one of that object’s methods ... the syntax is object.method

• Information can be passed to the method when it is invoked, and thought of as part of the message ... this information is called a parameter (or parameters)

Can: Use information sent to them by parameters

Can: Be “built-in” or constructed by the user

Can: Use both sequential (do in order) and concurrent (do together) processing

Can: Use control structures of the type used in imperative languages – loops and conditionals (if-else)

Can: Use loops that iterate a fixed number of times (Loop n times) or loop while a condition is true (While (condition is true))
Functions

- Functions are the name Alice gives to a special type of method which returns (passes back) information about the state of an object ... sometimes called Query Methods in OOP

- Some Functions are “built in”
- Other Functions can be constructed
- Some functions return a numerical value (“distance to ”)
- Some functions return a boolean value (“is to the left of ... “)
Events

• Sometimes methods are invoked when one object sends a message to another object -- invokes the method of another object -- these are called “programmatic”

• Other times methods are invoked by a click of a mouse or a move of a mouse or the press of a key … these are called “events”

• Parameters can be passed to methods by an event, and the parameters can be selected by the event causing object (for example … the mouse can click on an object to select it as a parameter).