Primitive Methods & Functions in Alice 2.2

Tables listing the primitive (built-in on all objects) methods and functions in Alice 2.2 (9/1/2008 edition).

Object Primitive Methods (Alice 2.2)

Method	Argument(s)	Notes
move	Direction – up/down, left/right, forward/back Amount (in metres)	Moves the object from its current location. Can be done relative to other objects, but generally from World's perspective.
turn	Direction – as above Amount (in revolutions)	Turn the object in the direction specified and the amount (1 = 1 full revolution)
roll	Direction – left/right Amount (in revolutions)	Rolls the object on its side
resize	Amount – Scaling factor	Multiplies the size of the object by the amount passed..
say	What – string	A speech bubble from the object
think	What – string	A thought bubble
playSound	Sound – A predefined set of sounds or an imported sound file	
moveTo	asSeenBy – the object to move to	Moves the object to the location of the target object.
moveToward	Target – object to be moved towards Amount – Distance to move	Move the object towards another.
moveAwayFrom	Target – object to move away from Amount – Distance to move	Move the object away from another
orientTo	asSeenBy – target object	Make the object share the same facing as the target object
turnToFace	Target – Object to a face	Turns the object to face the target. Only turns in the horizontal plane.
pointAt	Target – Object to face	Turns the object so its *forward* direction faces the centre of the target object
setPointOfView	asSeenBy – Point of View to assume	Sets the point of view of the object to that specified. A point of view is a location and orientation (facing).
setPose	Pose – Pose to Assume	Makes the object assume the pose. A pose is a recorded set of positions for the sub-objects that compose an object. Generally used for limb positioning.
standUp	<none>	Orients the object's *up* direction to the worlds

		up direction (nothing else!)
moveAtSpeed	Direction - up/down, left/right, forward/back Speed – metres per second	Makes the object move at the specified speed in the direction specified. Note that the default duration is 1 second (after which it stops)
turnAtSpeed	Direction - up/down, left/right, forward/back Speed – revolutions per second	Makes the object turn at the specified speed in the direction specified. Note that the default duration is 1 second (after which it stops)
rollAtSpeed	Direction – left/right Speed – revolutions per second	Makes the object roll at the specified speed in the direction specified. Note that the default duration is 1 second (after which it stops)
constrainToPointAt	Target – Object	Makes the calling object instantly point at (see that method) the passed object

Object Primitive Functions (Alice 2.2)

Function	Arguments(s)	Returned	Notes
isCloseTo	Threshold – distance Object - Object	True/False	Returns true if within threshold metres of the target object.
isFarFrom	Threshold – distance Object - Object	True/False	Returns true if beyond threshold metres of the target object.
distanceTo	Object – Object	Distance	Returns the distance between the centre of the invoking object and the target object
distanceToTheLeftOf	Object – Object	Distance	Returns the distance the invoking object is to the left of the target object
distanceToTheRightOf	Object – Object	Distance	Returns the distance the invoking object is to the right of the target object
distanceAbove	Object – Object	Distance	Returns the distance the invoking object is above the target object
distanceBelow	Object – Object	Distance	Returns the distance the invoking object is below the target object
distanceInFrontOf	Object – Object	Distance	Returns the distance the invoking object is in-front-of the target object
distanceBehind	Object – Object	Distance	Returns the distance the invoking object is behind the target object
getWidth	<none>	Size (metres)	Returns the width of the object
getHeight	<none>	Size (metres)	Returns the height of the object
getDepth	<none>	Size (metres)	Returns the depth of the object
isSmallerThan	Object – Object	True/false	Returns true if the invoking object is smaller

			than the passed object.
isLargerThan	Object – Object	True/false	Returns true if the invoking object is larger than the passed object.
isNarrowerThan	Object – Object	True/false	Returns true if the invoking object is narrower than the passed object.
isWiderThan	Object – Object	True/false	Returns true if the invoking object is wider than the passed object.
isShorterThan	Object – Object	True/false	Returns true if the invoking object is shorter than the passed object.
isTallerThan	Object – Object	True/false	Returns true if the invoking object is taller than the passed object.
isToTheLeftOf	Object – Object	True/false	Returns true if the invoking object is to the left of the passed object.
isToTheRightOf	Object – Object	True/false	Returns true if the invoking object is to the right of the passed object.
isAbove	Object – Object	True/false	Returns true if the invoking object is above the passed object.
isBelow	Object – Object	True/false	Returns true if the invoking object is below the passed object.
isInFrontOf	Object – Object	True/false	Returns true if the invoking object is in-front-of the passed object.
isBehind	Object – Object	True/false	Returns true if the invoking object is behind the passed object.
getPointOfView	<none>	Point of View	Returns the object's current point of view (position and orientation).
getPosition	<none>	Position	The object's position (but not orientation) in space.
getQuaternion	<none>	Quaternion	Returns the object's orientation.
getCurrentPose	<none>	Pose	Returns the object's current pose
partName	Key – String	Object	Returns the object's component object by the name of key (e.g., get the "rightLeg" sub-object of an object).