Technical

- `begin3d()`

  Description Initializes jReality and sets up the rendering environment. Deletes all former existing geometry, states or properties. In this way, the rendering environment is resetted in order to render a new frame with possibly new data.

  See also *end3d()*

- `end3d()`

  Description Signalizes that all relevant geometry input was made and the rendering can be started by the engine. Any changes to the geometry and appearance made after this command is executed will be ignored. This command should be called at the end of each script.

  See also *begin3d()*

General

- `gsave3d()`

  Description Pushes the current appearance states to save them for later use. The appearance state contains general size and color states as well as the corresponding states for the different primitives. This command corresponds to the Cindy `gsave()` command.

  See also *gsave(), grestore3d()*

- `grestore3d()`

  Description Pops the appearance state to restore the last pushed appearance. In this way, the last pushed settings for general size and color settings as well as the corresponding states for the different primitives are restored. This command corresponds to the Cindy `grestore()` command.

  See also *grestore(), gsave3d()*

- `color3d(color : [<real1>, <real2>, <real3>])`

  Description The color state of all primitives is set to `color`. So this command sets the color for all types of primitives. This command corresponds to the Cindy `color()` command.

  See also *color(), pointcolor(), pointcolo3d(), linecolor3d()*

- `size3d(size : <real>)`

  Description The size state of all primitves that uses a size parameter or similar is set to `size`. This command corresponds to the Cindy `size()` command.

  See also *size(), pointsize(), pointsize3d(), linesize3d()*

- `opacity3d(opacity : <real>)`

  Description Sets the opacity for every kind of geometry to `opacity`. A value of 0 represents total transparency whereas 1 results in opaque geometry. This command corresponds to the Cindy `opacity()` command.

  See also *opacity()*

- `shininess3d(shininess : <real>)`

  Description Changes the shininess for every kind of geometry to `shininess`. The shininess influences how specular a geometry surface is displayed.

- `background3d(color : [<real1>, <real2>, <real3>])`

  Description Sets background color to `color`.

Point

- `draw3d(position : [<real1>, <real2>, <real3>])`

  Description Adds a point at `position`. The appearance of the point references the current general appearance. This command corresponds to the Cindy `draw()` command.

  See also *draw()*

- `pointcolor3d(color : [<real1>, <real2>, <real3>])`

  Description Changes the point color state of the current general appearance. Corresponds to the Cindy `pointcolor()` command.

  See also *pointcolor(), color(), color3d()*

- `pointsize3d(size : <real>)`

  Description Changes the point size of the current general appearance. Corresponds to the Cindy `pointsize()` command.

  See also *pointsize(), size(), size3d()*

Line, Ray, Segment

- `draw3d(position1 : [<real1>, <real2>, <real3>], position2 : [<real1>, <real2>, <real3>])`

  Description Adds a line, ray or segment to the scene. The type of the object is specified by the modifier `type`. This command corresponds to the Cindy `draw()` command.

  See also *draw()*

  Modifiers TODO

- `connect3d(positions : <list<[<real1>, <real2>, <real3>]>>)`

  Description Adds a line strip to the scene. The position of the points are specified in the `positions` list. This command corresponds to the Cindy `connect()` command.

  See also *connect()*

- `lincolor3d(color : [<real1>, <real2>, <real3>])`

  Description Changes the line, ray and segment color of the current general appearance to `color`. This command corresponds to the Cindy `linecolor()` command.

  See also *linecolor(), color(), color3d()*

- `linesize3d(size : <real>)`

  Description Changes the line, ray and segment size of the current general appearance state to `size`.

  See also *size(), size3d()*

Sphere

- `drawsphere3d(position : [<real1>, <real2>, <real3>], radius : <real>)`

  Description Adds a sphere to the scene. The center of the sphere with radius `radius` is placed at `position`.

CIRCLE

- `drawcircle3d(position : [<real1>, <real2>, <real3>], radius : <real>, normal : [<real1>, <real2>, <real3>])`

    Description Adds the outline of a circle to the scene. The circle with radius `radius` is centered at `position`. The orientation of the circle is defined by its normal `normal`.

    See also *fillcircle3d()*

- `fillcircle3d(position : [<real1>, <real2>, <real3>], radius : <real>, normal : [<real1>, <real2>, <real3>])`

    Description Adds a circle to the scene. The circle with radius `radius` is centered at `position`. The orientation of the plane containing the circle is defined by its normal

POLYGON

- `drawpoly3d(positions : <list<[<real1>, <real2>, <real3>]>>)`

    Description Adds the outline of a polygon to the scene. The vertex positions are specified in `positions`. This command corresponds to the Cindy `drawpoly()` command.

    See also *drawpoly(), fillpoly(), fillpoly3d()*

- `fillpoly3d(positions : <list<[<real1>, <real2>, <real3>]>>)`

    Description Adds a polygon to the scene. The vertex positions are specified in `positions`. This command corresponds to the Cindy `fillpoly()` command.

    See also *fillpoly(), drawpoly(), drawpoly3d()*

- `mesh3d()`

    Description Adds a polygon mesh to the scene.

MISCELLANEOUS GEOMETRY

- `drawtorus3d(position : [<real1>, <real2>, <real3>], normal : [<real1>, <real2>, <real3>], oradius : <real>, iradius : <real>)`

    Description Adds a torus to the scene. The torus is centered at `position` and its orientation is defined by its normal *normal*. The outer and inner radius are defined by `oradius` and `iradius`.

LIGHT

- `setlight3d(lightid : <int>)`

    Description Changes light parameters for the light specified by its ID `lightid`. Which property of the specific light should be changed is defined by modifiers.

    Modifier

Modifier	Type	Effect
enabled	`<bool>`	Enables/Disables light source
position	`[<real1>, <real2>, <real3>, <real4>]`	Sets the homogeneous position
color	`[<real1>, <real2>, <real3>]`	Sets the light color

CAMERA

- `lookat3d(eyeposition : [<real1>, <real2>, <real3>], lookatposition : [<real1>, <real2>, <real3>], up : [<real1>, <real2>, <real3>])`

    Description Changes the camera position and orientation to match the specified parameters. With this command, the camera is positioned at `eyeposition` looking at `lookatposition`. The camera up vector is specified by `up`.