**CuraApplication 클래스**

* addCommandLineOptions()

**startSplashWindowPhase()**

setRequiredPlugins

# Misc.:

"ConsoleLogger", #read the log.

"CuraEngineBackend",

"FileLogger", #read the log.

"XmlMaterialProfile", #Cura crashes without this one.

"Toolbox", #This contains the interface to enable/disable plug-ins.

"PrepareStage",

"PreviewStage", #shows the list of the plugin views that are installed

"MonitorStage", #Major part of Cura's functionality.

"LocalFileOutputDevice", #Major part of Cura's functionality. "LocalContainerProvider", # profiles or setting definitions.

# Views:

"SimpleView", "SolidView", #Displays models.

# Readers & Writers:

"GCodeWriter", "STLReader", "3MFWriter",

# Tools:

"CameraTool", #Needed to see the scene.

"SelectionTool", #Dependency of the rest of the tools.

"TranslateTool", #You'll need this for almost every print.

Set the setting version for Preferences

closeApplication()

Main Flow

app.run()

#Initializing machine error checker"

MachineErrorChecker()

processEvents()

# Initializing machine manager

\_setLoadingHint()

getMachineManager()

processEvents()

# Initializing container manager

ContainerManager()

processEvents()

# Check if we should run as single instance or not.

# If so, set up a local socket server which listener which coordinates

# multiple Cura instances and accepts commands.

# Setup scene and build volume

\_setLoadingHint()

# initialize info objects

PrintInformation.PrintInformation()

CuraActions.CuraActions()

processEvent()

# Initialize setting visibility presets model.

SettingVisibilityPresetsModel()

# Initialize Cura API

self.\_cura\_API.initialize()

   self.\_output\_device\_manager.start()

   self.\_welcome\_pages\_model.initialize()

   self.\_add\_printer\_pages\_model.initialize()

   self.\_whats\_new\_pages\_model.initialize()

# Detect in which mode to run and execute that mode

##  Run Cura with GUI (desktop mode).

runWithGUI()

getController()

# get Translate Tool and set enabledAxis

# set default background color for scene

# Initialize platform physics

PlatformPhysics.PlatformPhysics(controller, self.\_volume)

# Initialize camera

# Initialize camera tool

# Initialize camera animations

# Initialize QML engine

# Initialize UI state

# Hide the splash screen