ISXEQ2 - The ReforgeWindow

To use this test script, create a file in your /innerspace/scripts folder called ReforgeWindow.iss and place the entire text below. Then, you can run the script by typing the following command in the InnerSpace console (while in the game): run ReforgeWindow

```
function main()
{
    variable index:collection:string Options
    variable iterator OptionsIterator
    variable int OptionCounter = 0

    if (!${ReforgeWindow.IsVisible})
    {
        echo "ERROR: The ReforgeWindow is not currently visible."
        return
    }

    echo "ReforgeWindow Information:"
    echo "- Item Name: ${ReforgeWindow.ItemName.GetProperty[LocalText]}"
    echo "- StatsText: ${ReforgeWindow.StatsText.GetProperty[LocalText]}"
    echo "- SourceDiff: ${ReforgeWindow.SourceDiff.GetProperty[LocalText]}"
    echo "- DestDiff: ${ReforgeWindow.DestDiff.GetProperty[LocalText]}"
    echo "- ReforgeItemNameAndPrice:
${ReforgeWindow.ReforgeItemNameAndPrice.GetProperty[LocalText]}"
    echo "- DecorationName: ${ReforgeWindow.DecorationName.GetProperty[LocalText]}"
    echo "- Current Copper: ${ReforgeWindow.Copper.GetProperty[LocalText]}"
    echo "- Current Silver: ${ReforgeWindow.Silver.GetProperty[LocalText]}"
    echo "- Current Gold: ${ReforgeWindow.Gold.GetProperty[LocalText]}"
    echo "- Current Platinum: ${ReforgeWindow.Platinum.GetProperty[LocalText]}"
    echo "- AttributeSlider:"
    echo "-- LowerLimit: ${ReforgeWindow.AttributeSlider.GetProperty[LowerLimit]}"
    echo "-- UpperLimit: ${ReforgeWindow.AttributeSlider.GetProperty[UpperLimit]}"
    echo "-- Value: ${ReforgeWindow.AttributeSlider.GetProperty[Value]}
(${Math.Calc[${ReforgeWindow.AttributeSlider.GetProperty[Value]}/${ReforgeWindow.AttributeSlider.G
etProperty[UpperLimit]}*100].Precision[1]} %)"
    echo "- ReforgeItemIcon:"
    echo "-- IconID: ${ReforgeWindow.ReforgeItemIcon.GetProperty[IconID]}"
    echo "-- IconType: ${ReforgeWindow.ReforgeItemIcon.GetProperty[IconType]}"
    echo "- DecorationSlot:"
    echo "-- IconID: ${ReforgeWindow.DecorationSlot.GetProperty[IconID]}"
    echo "-- IconType: ${ReforgeWindow.DecorationSlot.GetProperty[IconType]}"
    echo "- ReforgeButton: ${ReforgeWindow.ReforgeButton.GetProperty[LocalText]} (Enabled:
${ReforgeWindow.ReforgeButton.GetProperty[Enabled]})"
    echo "- CancelReforgeButton: ${ReforgeWindow.CancelReforgeButton.GetProperty[LocalText]}
(Enabled: ${ReforgeWindow.CancelReforgeButton.GetProperty[Enabled]})"
    echo "- SourceDropdown:"
        ReforgeWindow.SourceDropdown:GetOptions[Options]
        Options:GetIterator[OptionsIterator]
        echo "-- NumOptions: ${Options.Used}"
        if (${OptionsIterator:First(exists)})
        {
            do
            {
                if (${OptionsIterator.Value.FirstKey(exists)})
                {
                    do
                    {
                         echo "--- Option #${OptionCounter}::
 '${OptionsIterator.Value.CurrentKey}' => '${OptionsIterator.Value.CurrentValue}'"
                    }
                    while ${OptionsIterator.Value.NextKey(exists)}
                }
                OptionCounter:Inc
            }
            while ${OptionsIterator:Next(exists)}
        }
```

```
    echo "- DestDropdown:"
        OptionCounter:Set[0]
        ReforgeWindow.DestDropdown:GetOptions[Options]
        Options:GetIterator[OptionsIterator]
        echo "-- NumOptions: ${Options.Used}"
        if (${OptionsIterator:First(exists)})
        {
            do
            {
                if (${OptionsIterator.Value.FirstKey(exists)})
                {
                    do
                    {
                         echo "--- Option #${OptionCounter}::
 '${OptionsIterator.Value.CurrentKey}' => '${OptionsIterator.Value.CurrentValue}'"
                    }
                    while ${OptionsIterator.Value.NextKey(exists)}
                }
                OptionCounter:Inc
            }
            while ${OptionsIterator:Next(exists)}
        }

    ;; NOTES:
    ;;
    ;; 1. Example of setting the AttributeSlider to 25%:
    ;;      -
ReforgeWindow.AttributeSlider:SetProperty[Value,${Math.Calc[25*${ReforgeWindow.AttributeSlider.Get
Property[UpperLimit]}/100].Precision[0]}]
    ;;    (Please note that you must use an integer value, which is why the "Precision[0]" is
important.)
    ;;
    ;; 2. To set a DropDownBox to a particular value, one would iterate through the "options" (as
shown above), and then when
    ;;    the option "CurrentKey" and/or "CurrentValue" matches the goal, use the "Set" method
using the current Counter (zero-based).
    ;;    For example:  ReforgeWindow.DestDropdown:Set[${OptionCounter}]
}
```