

## FlightScript™

### Applescript Functions

FlightCheck provides several scripting functions which you can access via AppleScript®. While the subject of scripting itself is beyond the scope of this manual, the examples given below can be used to build your scripts:

#### Obtaining a List

Executing the following script will return a list of colors:

```
tell application "FlightCheck"
```

```
activate
```

```
get colorlist
```

```
end tell
```

#### Getting the Current View

Executing the following script will return the current view for fonts:

```
tell application "FlightCheck"
```

```
activate
```

```
get font view
```

```
end tell
```

#### Getting the Number of Problems

Executing the following script will return the number of problem images (using the currently selected view):

```
tell application "FlightCheck"
```

```
activate
```

```
get image problems
```

```
end tell
```

### Conditional Scripts

Executing the following conditional script will return a list of colors only if the current view is "Spot Colors", set the current font view to All Fonts only if there are font problems, and will return a list of images only if there are image problems:

tell application "FlightCheck"

activate

if (get color view) = ("Spot Colors") then get color list

if (get font problems) > 0 then set font view to All Fonts

if (get image problems) > 0 then get image list

end tell

Scripting Functions

The list of possible scripting commands is as follows:

COMMAND

RESULT

flightcheck "file name"  
FlightCheck named file

get color list

returns a list of colors

get font list

returns a list of fonts

get image list

returns a list of images

get image names list

returns a list of images by full pathname

get RGB images

returns a list of RGB images

set color view to \_\_\_\_  
sets the view for colors

Used Colors

All Colors

Spot Colors

Process Colors

Trapped Colors

Unused Colors

Mismatched Colors

Problem Colors

set font view to \_\_\_\_  
sets the view for fonts

All Fonts

Active Fonts

Database Fonts

Missing Fonts

Missing PS Fonts

Menu Styled

Problem Fonts

set image view to \_\_\_\_  
sets the view for images

All Images

Missing Images

Modified Images

Unused Images

Stored Images

Bad Resolution

OPI Images

RGB Images

Compressed

Fill "None"

Stylized Images

Bitmap Frames

Problem Images

get color view

returns the view for colors

get font view

returns the view for fonts

get image view

returns the view for images

load "controls name"  
loads named Ground Controls file

select set "set name"

selects named Ground Controls set

get ground controls title  
returns name of Ground Controls

do full flightcheck

recheck file

get flightcheck results  
returns the list of results

print results

prints the results

show specific items

opens specific items

show general items

closes specific items

print report

prints a FlightCheck report

save report

saves a FlightCheck report

save report "report name"  
saves a FlightCheck report to file name

collect report

saves a collect report

collect report "report name"  
save a collect report to file name

collect job

collects the job

collect job "job name"  
collects job to folder name

get document title

returns document name

get application info

returns application info

get file info

returns file info

get page info

returns page info

get page setup info

returns page setup info

get printer info

returns printer info

get trap info

returns trap info

open form "form name"  
opens named custom form

print form

prints the current form

close form

closes the current form

#### Scripting a Workflow

You can use the following list of AppleScript commands which you can send to FlightCheck in order to create a workflow that can automatically check documents and obtain the results.

Note that lines beginning with "-" dashes are commented out. Simply remove the dashes at your discretion before running the script:

tell application "FlightCheck"

-activate

with timeout of 120 seconds



--IN CASE OF LARGE FILE

--flightcheck "my drive:my folder:my document file"

--end timeout

-GET A FILE VIA AN OPEN DIALOG:

-set theFile to choose file

-flightcheck theFile

-load "my drive:my folder:my ground controls file"

-select set "my set"

-do full flightcheck

-get flightcheck results

-print report

-save report

-save report "my report"

-collect report

-collect report "my report"

-collect job

-collect job "my job"

-open form "my drive:my folder:my form file"

-print form

-close form

-quit application "FlightCheck" saving no  
end tell