



## 从命令行运行 (Linux)

`runscript`命令启动eggPlant Functional并运行脚本，不会显示eggPlant Functional的图形用户界面 (GUI)。（要了解如何通过命令行控制eggPlant Functional的全屏GUI，参见下文的[命令行中的GUI选项](#)。）

1. 键入eggPlant Functional应用的完整路径、`/runscript`，以及一个空格。
2. 键入一个脚本或套件的完整路径，并运行。
3. 要添加一个额外的脚本或套件，插入一个空格，然后键入该脚本或套件的完整路径。

示例: `/usr/GNUstep/Local/Applications/Eggplant.app/runscript /Users/Alex/appTests.suite`

`"/usr/GNUstep/Local/Applications/Eggplant.app/runscript" "/Users/Alex/appTests.suite/scripts/test1.script"`

eggPlant Functional在“结果”文件夹中，为当前脚本的套件保存每个脚本运行的结果。然后，在所有脚本和套件结束运行以后，eggPlant Functional报告成功运行的脚本和套件的数量（返回值）。（套件中的脚本均成功运行，则该套件成功运行。）

注：为设置eggPlant Functional只报告成功运行次数，将`-ReportFailures`选项设置为“no”。

### 传递参数

可以将参数传递给单个脚本，或即将运行的多个脚本中的最后一个。在`runscript`行的末尾，键入`-params`，然后输入参数，以空格隔开。

示例: `"/usr/GNUstep/Local/Applications/Eggplant.app/runscript" /Users/Alex/appTests.suite/Scripts/test1.script -params "SUT1" "phonebook.txt"`

每个参数都作为一个文本串通过。

### “运行脚本”选项

下面介绍的选项，可以添加到`runscript`行中。要添加多个选项，每个选项前需插入一个空格。

#### -CommandLineOutput

用途：控制是否输出脚本日志（运行窗口的日志区域显示的数据。）要输出脚本日志，将“CommandLineOutput”全局属性设置为`yes`。不输出脚本日志，将“CommandLineOutput”全局属性设置为`no`。

示例: `"/usr/GNUstep/Local/Applications/Eggplant.app/runscript" /Users/Alex/appTests.suite/scripts/test3.script -CommandLineOutput NO`

也可以在脚本中设置`CommandLineOutput`全局属性。全局属性在脚本中修改后，始终会覆写初始设置。

注：输出设置为标准错误 (`stderr`) 后，始终会显示。从命令行运行时，要更精确地控制输出，可以在脚本中，使用以下语句：

```
write "THIS IS SOME OUTPUT" to error
```

#### 连接信息 (-colorDepth, -host, -password, -port, -username)

用途：-host引数是脚本中使用的特定SUT的服务器ID。如果加入了-host引数，还应该指定色深、密码、端口以及用户名。（用户名是指你在VNC服务器的用户名称，主要是UltraVNC使用的。）

示例: `"/usr/GNUstep/Local/Applications/Eggplant.app/runscript" /Users/Alex/Docs.suite/scripts/myTests/test2.script -host SUT1 -port 5901 -password broncos -colorDepth 16`

注：来自运行脚本内的连接命令或者计划表中的连接信息会覆写该连接。（该连接仍然保持打开，但是新连接会变为当前激活的连接。）

#### -GlobalResultsFolder

用途：设置保存脚本结果的文件夹的名称。文件夹路径可以是绝对路径，也可以关联到脚本的套件上路径上。

示例: `"/usr/GNUstep/Local/Applications/Eggplant.app/runscript" /Users/Alex/Docs.suite/scripts/test2.script -GlobalResultsFolder ~/MyResults`

#### -DefaultDocumentDirectory

用途：相对文件路径的起始目录，比如帮助套件的路径。

示例: `"usr/GNUstep/Local/Applications/Eggplant.app/runscript" -DefaultDocumentDirectory "~/MyDocuments/eggPlantTests"`

#### -repeat

用途：重复所运行的脚本x次。

示例: `"/usr/GNUstep/Local/Applications/Eggplant.app/runscript" "/Users/Alex/myTests.suite" -repeat 4`

#### -ReportFailures

用途：在脚本执行以后，返回失败脚本和套件的数量给命令行。如果没有将“-ReportFailures”设置为“yes”，eggPlant Functional会向命令行报告成功脚本和套件的数量。

示例: `"/usr/GNUstep/Local/Applications/Eggplant.app/runscript" "/Users/Alex/Docs.suite" -ReportFailures yes`

**运行计划**

可以为整个套件而不是单个脚本，指定一个运行脚本命令，来调用一个计划。该操作会运行套件中的第一个计划（按字母顺序）。如果没有计划可以运行，eggPlant Functional会返回一个“No Scripts to Run”的错误。

示例：`"usr/GNUstep/Local/Applications/Eggplant.app/runscript" "/Users/Alex/Docs.suite"`

要运行含有多个计划的套件中的某个具体的计划，在套件路径中指定该计划的名称。

示例：`"usr/GNUstep/Local/Applications/Eggplant.app/runscript" "/Users/Alex/Docs.suite/Schedule1"`

注：这可以用于`-RunOnLaunch`标记。

## 从命令行启动eggPlant Functional的GUI

为从命令行启动eggPlant Functional应用，键入可执行的eggPlant Functional路径。

示例：`"usr/GNUstep/Local/Applications/Eggplant.app/Eggplant"`

注：当从命令行启动eggPlant Functional的GUI时，终端不会显示标准输出。输出会以日志的形式，保存在用户账号的`~/Library/Eggplant`目录中。

### 命令行选项

- AlertOnError**
- 用途：覆写eggPlant Functional的一般偏好设置**连接失败时显示提醒**。（取yes/no值。）
- 示例：`"usr/GNUstep/Local/Applications/Eggplant.app/Eggplant" -AlertOnError yes`
- AutoShowRunWindow**
- 用途：覆写eggPlant Functional的一般偏好设置**脚本运行时将运行窗口前移一层**。（要求yes/no值。）
- 示例：`"usr/GNUstep/Local/Applications/Eggplant.app/Eggplant" -AutoShowRunWindow`
- QuitAfterRun**
- 用途：运行给定脚本或套件，然后退出eggPlant Functional。（要求yes/no值。）（取no等同于省略该选项。）要取消退出，选择Eggplant菜单>退出，然后点击“取消”按钮。
- 示例：`"usr/GNUstep/Local/Applications/Eggplant.app/Eggplant" -QuitAfterRun`
- RestoreConnection**
- 用途：覆写eggPlant Functional的一般偏好设置**重新打开上次会话的连接**。（取yes/no值。）
- 示例：`"usr/GNUstep/Local/Applications/Eggplant.app/Eggplant" -RestoreConnection yes`
- ReopenDocuments**
- 用途：覆写eggPlant Functional的一般偏好设置**从上次会话重新打开套件和脚本**。（取yes/no值。）
- 示例：`"usr/GNUstep/Local/Applications/Eggplant.app/Eggplant" -ReopenDocuments no`
- RunOnLaunch**
- 用途：启动eggPlant Functional应用后，运行给定脚本或套件。
- 示例：`"usr/GNUstep/Local/Applications/Eggplant.app/Eggplant" -RunOnLaunch "/Users/Alex/Alex_Tests.suite"`