Figure 1 is the initial display of TestStand.exe when started from `CMD.EXE` *without* command line arguments.

Figure 2 demonstrates all overloads of static `MessageInColor.RGBWriteLine()`. This test uses a color scheme of `Yellow` text on a `DarkMagenta` background.

File C:\USERS\DAVE\DOCUMENTS\VISUAL STUDIO 2013\PROJECTS\WIZARDWRX_LIBS\CONSOLEAPPAIDS2\NOTES\TEST_RUN_WITHOUT_COMMAND_LINE_ARGUMENTS.DOCX

Page 1 of 10 pages © 2015, Simple Soft Services, Inc., d/b/a WizardWrx Current as of 2015-06-06 01:24:00

```
Command Prompt - "C:\Users\DAVE\Documents\Visual Studio 2013\Projects\WizardWrx_Libs\ConsoleAppAids2\TestStand\bin\Release\TestStand.exe"
Static MessageInColor.RGBWrite() Tests Begin

    Overload  1: True
    Overload  2: E
    Overload  3: FGH
    Overload  4: 1.25
    Overload  5: 3.333333
    Overload  6: 3.3333
    Overload  7: -255
    Overload  8: -125000
    Overload  9: Stuff 1
    Overload 10: This is it.
    Overload 11: 512
    Overload 12: 250000
    Overload 13: object arg0: object arg0 = Stuff 1
    Overload 14: params object [ ] arg: params object [ ] arg = Stuff 1
    Overload 15: GH
    Overload 16: object arg0, object arg1: object arg0 = Stuff 1, object arg1 =
Stuff 2
    Overload 17: object arg0 = Stuff 1, object arg1 = Stuff 2, object arg2 = Stu
ff 3
    Overload 18: object arg0 = Stuff 1, object arg1 = Stuff 2, object arg2 = Stu
ff 3, object arg3 = Stuff 4
Static MessageInColor.RGBWrite() Tests End
Capture this screen, then press the ENTER key.
```

Figure 3 demonstrates all overloads of static `MessageInColor.RGBWrite()`, using a new color scheme, `Black` text on a `Yellow` background.

```
Command Prompt - "C:\Users\DAVE\Documents\Visual Studio 2013\Projects\WizardWrx_Libs\ConsoleAppAids2\TestStand\bin\Release\TestStand.exe"
Instance MessageInColor.WriteLine() Tests Begin

    Overload  1: True
    Overload  2: E
    Overload  3: FGH
    Overload  4: 1.25
    Overload  5: 3.333333
    Overload  6: 3.3333
    Overload  7: -255
    Overload  8: -125000
    Overload  9: Stuff 1
    Overload 10: This is it.
    Overload 11: 512
    Overload 12: 250000
    Overload 13: object arg0: object arg0 = Stuff 1
    Overload 14: params object [ ] arg: params object [ ] arg = Stuff 1
    Overload 15: GH
    Overload 16: object arg0, object arg1: object arg0 = Stuff 1, object arg1 =
Stuff 2
    Overload 17: object arg0 = Stuff 1, object arg1 = Stuff 2, object arg2 = Stu
ff 3
    Overload 18: object arg0 = Stuff 1, object arg1 = Stuff 2, object arg2 = Stu
ff 3, object arg3 = Stuff 4
Instance MessageInColor.WriteLine() Tests End
Capture this screen, then press the ENTER key.
```

Figure 4 demonstrates all overloads of instance `MessageInColor.WriteLine()`, using a new color scheme of `Cyan` text on a `DarkMagenta` background.

```
Command Prompt - "C:\Users\DAVE\Documents\Visual Studio 2013\Projects\WizardWrx_Libs\ConsoleAppAids2\TestStand\bin\Release\TestStand.exe"

Instance MessageInColor.Write() Tests Begin

    Overload  1: True
    Overload  2: E
    Overload  3: FGH
    Overload  4: 1.25
    Overload  5: 3.333333
    Overload  6: 3.3333
    Overload  7: -255
    Overload  8: -125000
    Overload  9: Stuff 1
    Overload 10: This is it.
    Overload 11: 512
    Overload 12: 250000
    Overload 13: object arg0: object arg0 = Stuff 1
    Overload 14: params object [ ] arg: params object [ ] arg = Stuff 1
    Overload 15: GH
    Overload 16: object arg0, object arg1: object arg0 = Stuff 1, object arg1 =
Stuff 2
    Overload 17: object arg0 = Stuff 1, object arg1 = Stuff 2, object arg2 = Stu
ff 3
    Overload 18: object arg0 = Stuff 1, object arg1 = Stuff 2, object arg2 = Stu
ff 3, object arg3 = Stuff 4
Instance MessageInColor.Write() Tests End
Capture this screen, then press the ENTER key.
```

Figure 5 demonstrates all overloads of instance `MessageInColor.Write()`, using a new color scheme of `Black` text on a `White` background.

```
Command Prompt - "C:\Users\DAVE\Documents\Visual Studio 2013\Projects\WizardWrx_Libs\ConsoleAppAids2\TestStand\bin\Release\TestStand.exe"

Error reporting test 1 of 5 Begin

An System.NullReferenceException exception was caught in routine ExerciseColorEx
ceptionReporting:
                Message         = Object reference not set to an instance of an
object.
Exception Method = Void ExerciseColorExceptionReporting()
Exception Source = TestStand
There are no further exceptions to report.

Error reporting test 1 of 5 End


Press the ENTER key to display the next example.
```

Figure 6 is the first of 5 error reporting tests. This test and the next four use exceptions that are intentionally thrown, caught, and reported through the `AppExceptionLogger` class, and setting its `ErrorMessageColors` property to one of 5 carefully selected color pairs.

```
Command Prompt - "C:\Users\DAVE\Documents\Visual Studio 2013\Projects\WizardWrx_Libs\ConsoleAppAids2\TestStand\bin\Release\TestStand.exe"

Error reporting test 2 of 5 Begin

An System.Exception exception was caught in routine ExerciseColorExceptionReport
ing:
              Message           = This error message appears in White text on Re
d background.
Exception Method = Void ExerciseColorExceptionReporting()
Exception Source = TestStand
There are no further exceptions to report.

Error reporting test 2 of 5 End


Press the ENTER key to display the next example.
```

Figure 7 is the second of the 5 error message tests. The bright red background in this test is pretty intense, especially in contrast to the red text on a white background shown in **Figure 6**.


```
Command Prompt - "C:\Users\DAVE\Documents\Visual Studio 2013\Projects\WizardWrx_Libs\ConsoleAppAids2\TestStand\bin\Release\TestStand.exe"

Error reporting test 3 of 5 Begin

An System.Exception exception was caught in routine ExerciseColorExceptionReport
ing:
              Message           = This error message appears in Yellow text on R
ed background.
Exception Method = Void ExerciseColorExceptionReporting()
Exception Source = TestStand
There are no further exceptions to report.

Error reporting test 3 of 5 End


Press the ENTER key to display the next example.
```

Figure 8 is the third of 5 error message tests, rendered in Yellow text on a bright Red background. This color scheme is almost as intense as the one shown in **Figure 7**.

```
Command Prompt - "C:\Users\DAVE\Documents\Visual Studio 2013\Projects\WizardWrx_Libs\ConsoleAppAids2\TestStand\bin\Release\TestStand.exe"

Error reporting test 4 of 5 Begin

An System.Exception exception was caught in routine ExerciseColorExceptionReport
ing:
             Message           = This error message appears in White text on Da
rkRed background.
Exception Method = Void ExerciseColorExceptionReporting()
Exception Source = TestStand
There are no further exceptions to report.

Error reporting test 4 of 5 End


Press the ENTER key to display the next example.
```
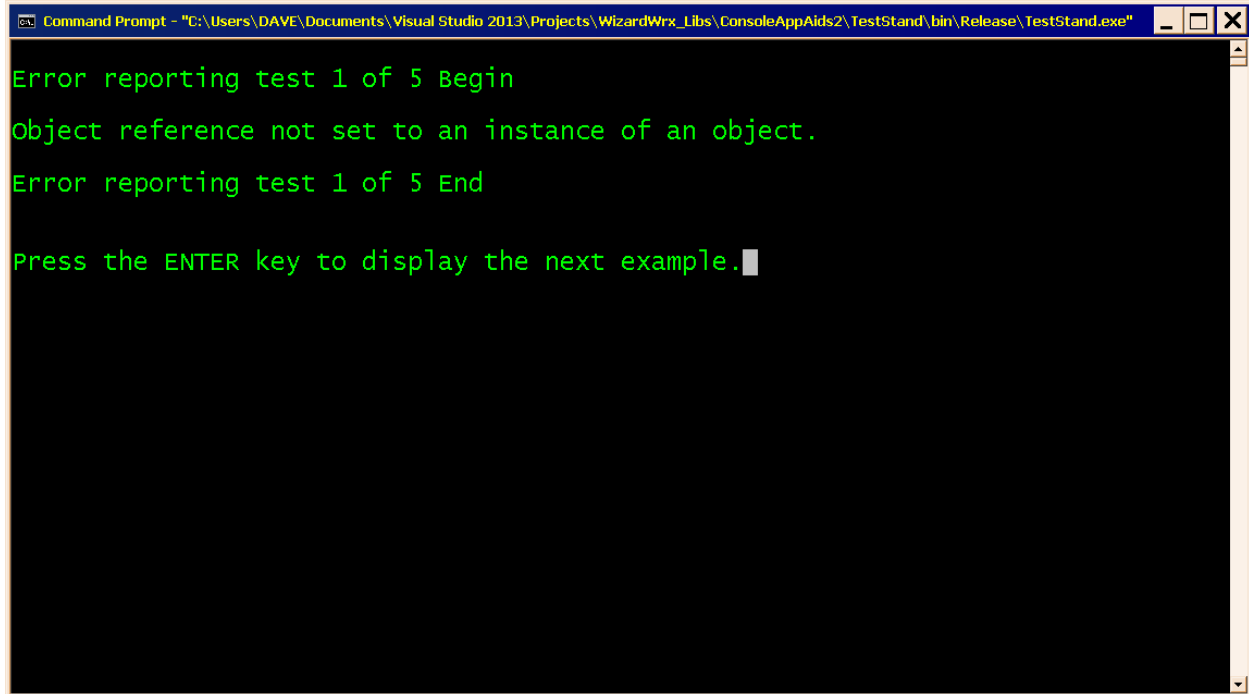
Figure 9 is the fourth of the 5 error message displays. The `DarkRed` background gets your attention without being nearly as overwhelming as the regular red used in the three previous examples. By a small margin, this color scheme is my favorite.
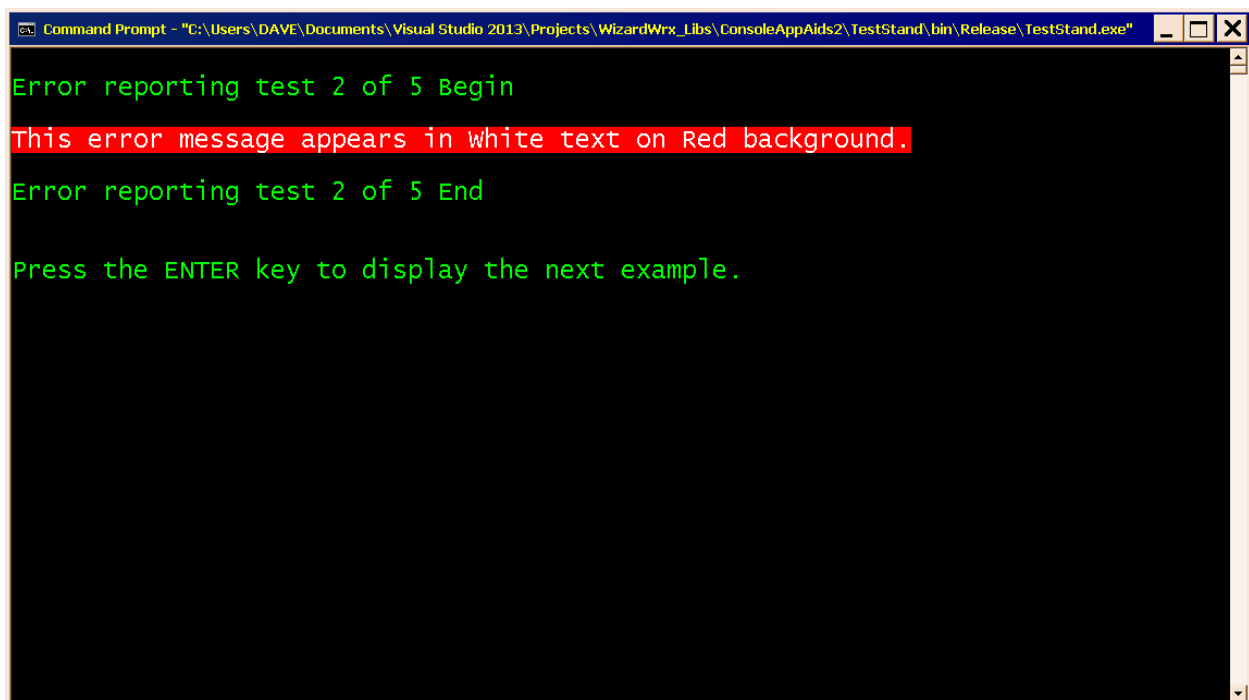
```
Command Prompt - "C:\Users\DAVE\Documents\Visual Studio 2013\Projects\WizardWrx_Libs\ConsoleAppAids2\TestStand\bin\Release\TestStand.exe"

Error reporting test 5 of 5 Begin

An System.Exception exception was caught in routine ExerciseColorExceptionReport
ing:
             Message           = This error message appears in Yellow text on D
arkRed background.
Exception Method = Void ExerciseColorExceptionReporting()
Exception Source = TestStand
There are no further exceptions to report.

Error reporting test 5 of 5 End


Press the ENTER key to display the next example.
```

Figure 10 is the fifth of the 5 color scheme demonstrations. This is my second preference, mostly because I think the yellow text sends a somewhat mixed signal; it is Stop or Caution?

Figure 11 is the first of 5 terse error message demonstrations. This baseline demonstration uses the default color scheme of my console window, which is **Green** text on a **Black** background. Your results will vary, depending on the colors set for your CMD.EXE window.

Figure 12 demonstrates the color scheme first shown in **Figure 7** with a terse message.

Figure 13 demonstrates the color scheme first shown in **Figure 8** with a terse message.

Figure 14 demonstrates the color scheme first shown in **Figure 9** with a terse message.

Figure 15 demonstrates the color scheme first shown in **Figure 10** with a terse message.

Figure 16 is the first of 3 demonstrations of the `FixedConsoleWriter` class. This is its most basic application, where the text starts at the left edge of the window.

```
Command Prompt - "C:\Users\DAVE\Documents\Visual Studio 2013\Projects\WizardWrx_Libs\ConsoleAppAids2\TestStand\bin\Release\TestStand.exe"

       Count    =      10 (Default =     10)
       Interval =  1,000 (Default =  1,000)

Test # 20 begin

    Test a FixedConsoleWriter with Black foreground color and
    White background color. Text is indented 0 spaces from the left.
Waiting for 1,000 milliseconds (1 seconds). This is pause 10 of 10.
Test # 20 end

Test # 21 begin

    Test a FixedConsoleWriter with Black foreground color and
    White background color. Text is indented 4 spaces from the left.

    Waiting for 1,000 milliseconds (1 seconds). This is pause 10 of 10.
Test # 21 end

Test # 22 begin

    Test a FixedConsoleWriter with Black foreground color and
    White background color. Text is indented 8 spaces from the left.

        Waiting for 1,000 milliseconds (1 seconds). This is pause 10 of 10.
```

Figure 17 is the second of 3 demonstrations of the `FixedConsoleWriter` class, showing resumption after the first test completed, followed by a demonstration in which the message is displayed at an indent of 4 spaces, then another set displayed at an indent of 8 spaces.

```
Command Prompt - "C:\Users\DAVE\Documents\Visual Studio 2013\Projects\WizardWrx_Libs\ConsoleAppAids2\TestStand\bin\Release\TestStand.exe"

    White background color. Text is indented 8 spaces from the left.

        Waiting for 1,000 milliseconds (1 seconds). This is pause 10 of 10.
Test # 22 end

Test # 23 begin

    Test a FixedConsoleWriter with Black foreground color and
    White background color. Text is indented 12 spaces from the left.

            Waiting for 1,000 milliseconds (1 seconds). This is pause 10 of 10.
Test # 23 end

Test # 24 begin

    Test a FixedConsoleWriter with Black foreground color and
    White background color. Text is indented 16 spaces from the left.

                Waiting for 1,000 milliseconds (1 seconds). This is pause 10 of
10.
Test # 24 end

Test # 24: This test displays a countdown of 30 seconds.

Test ending in 22 seconds
```

Figure 18 is the third demonstration, in which the text wraps around, followed by another countdown displayed in blue text on a yellow background.

```
     MessageForegroundColor      = Yellow
     EventLoggingState            = Enabled ALERT: This property is deprecated.
An System.Exception exception was caught in routine TimedExitTest:
             Message               = This is a test exception. This is only a test.

Exception Method = Void TimedExitTest(Int32 ByRef, WizardWrx.ApplicationHelpers2
.CmdLneArgsBasic)
Exception Source = TestStand
There are no further exceptions to report.
Test TimeedwaitTest 24:
An System.IO.DirectoryNotFoundException exception was caught in routine Timeedwa
itTest:
             Message               = Could not find a part of the path 'C:\Document
s and Settings\DAG\My Documents\Visual Studio 2010\Projects\wizardwrx_Libs\Conso
leAppAids2\NOTES\Test_Data\TimedwaitTests.TXT'.
Exception Method = Void WinIOError(Int32, System.String)
Exception Source = mscorlib
There are no further exceptions to report.
Test TimeedwaitTest 24 is over.

TestStand End
2015-06-06 02:08:17.367 (2015-06-06 07:08:17.367 UTC)
Elapsed time: 0:00:48:21.753

C:\Users\DAVE>
```

Figure 19 repeats the demonstration of my two preferred color schemes, coupled with interruptible timed waits, both of which were interrupted by keyboard inputs.