

Graphic bundle

Chapter 1: Let's Start Programming

```
13:11:57-root:/tmp$ who
ggarrelli console Jan 20 06:49
ggarrelli ttyS000 Jan 20 06:49
ggarrelli ttyS002 Jan 20 06:49
ggarrelli ttyS003 Jan 20 13:10
13:12:00-root:/tmp$ echo "Hello, how are you doing?" > /dev/ttyS003
13:12:03-root:/tmp$ Fine, thanks
█

x bash
#13:11:53-ggarrelli:~$ Hello, how are you doing?
#13:12:06-ggarrelli:~$ echo "Fine, thanks" > /dev/ttyS002
#13:12:08-ggarrelli:~$ █
```

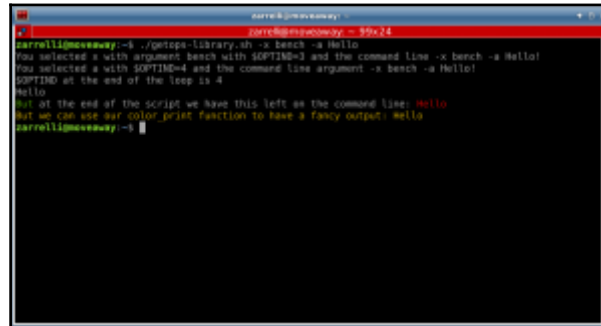
Chapter 4: Quoting and Escaping



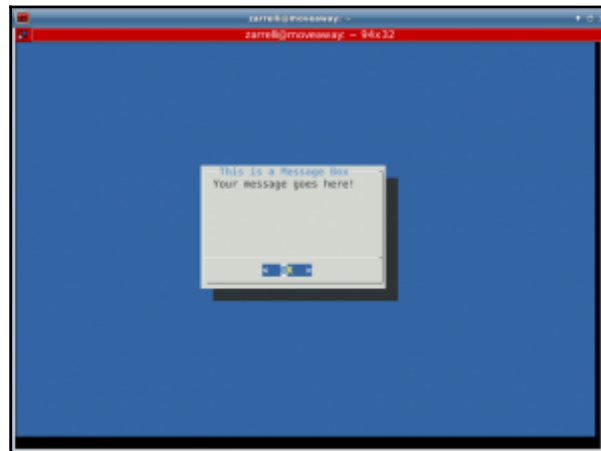
```
zarrelli@moveaway: ~  
zarrelli@moveaway: ~ ?ix8  
zarrelli@moveaway:~$ for i in (1..100) ; do echo $i ; sleep 1; done  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
zarrelli@moveaway:~$
```

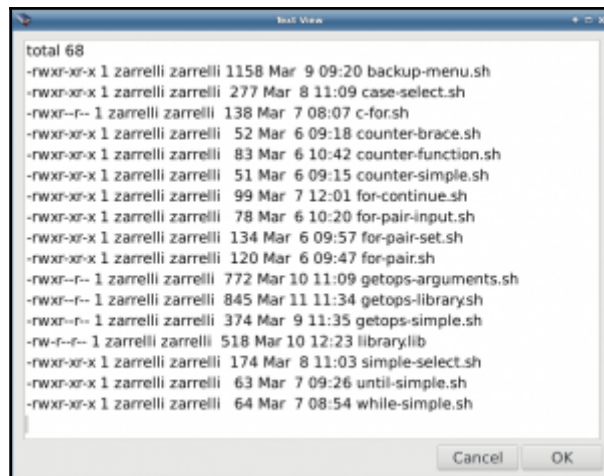
Chapter 6: Iterations

```
1) backup-menu.sh      6) counter-simple.sh  11) for-pair.sh
2) case-select.sh     7) EXIT               12) simple-select.sh
3) c-for.sh           8) for-continue.sh   13) until-simple.sh
4) counter-brace.sh   9) for-pair-input.sh  14) while-simple.sh
5) counter-function.sh 10) for-pair-set.sh
Which file do you want to backup? █
```

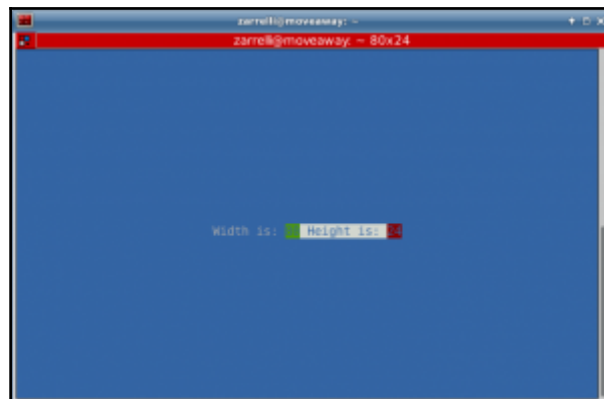


A terminal window titled 'zarrell@ironwasp' with a red title bar. The prompt is 'zarrell@ironwasp ~'. The user enters './getopts-library.sh -x bench -a Hello'. The output shows: 'You selected a with argument bench with SOPTIND=3 and the command line -x bench -a Hello!', 'You selected a with SOPTIND=4 and the command line argument -x bench -a Hello!', 'SOPTIND at the end of the loop is 4', 'Hello', and 'But at the end of the script we have this left as the command line: hello'. A second line of output says 'But we can use our color_print function to have a fancy output: hello' in red text. The prompt returns to 'zarrell@ironwasp ~'.





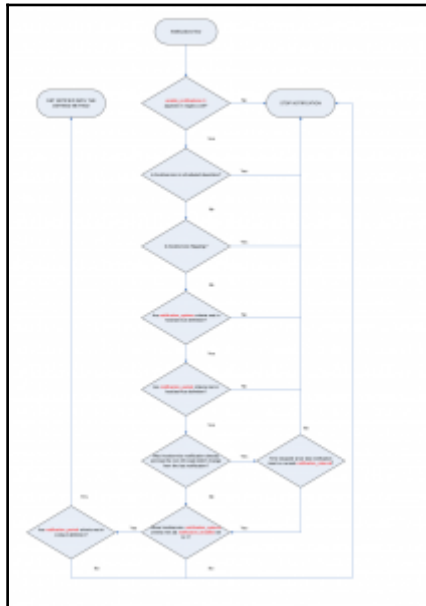
```
total 68
-rwxr-xr-x 1 zarrelli zarrelli 1158 Mar  9 09:20 backup-menu.sh
-rwxr-xr-x 1 zarrelli zarrelli  277 Mar  8 11:09 case-select.sh
-rwxr--r-- 1 zarrelli zarrelli  138 Mar  7 08:07 c-for.sh
-rwxr-xr-x 1 zarrelli zarrelli   52 Mar  6 09:18 counter-brace.sh
-rwxr-xr-x 1 zarrelli zarrelli   83 Mar  6 10:42 counter-function.sh
-rwxr-xr-x 1 zarrelli zarrelli   51 Mar  6 09:15 counter-simple.sh
-rwxr-xr-x 1 zarrelli zarrelli   99 Mar  7 12:01 for-continue.sh
-rwxr-xr-x 1 zarrelli zarrelli   78 Mar  6 10:20 for-pair-input.sh
-rwxr-xr-x 1 zarrelli zarrelli  134 Mar  6 09:57 for-pair-set.sh
-rwxr-xr-x 1 zarrelli zarrelli  120 Mar  6 09:47 for-pair.sh
-rwxr--r-- 1 zarrelli zarrelli  772 Mar 10 11:09 getops-arguments.sh
-rwxr--r-- 1 zarrelli zarrelli  845 Mar 11 11:34 getops-library.sh
-rwxr--r-- 1 zarrelli zarrelli  374 Mar  9 11:35 getops-simple.sh
-rw-r--r-- 1 zarrelli zarrelli  518 Mar 10 12:23 library.lib
-rwxr-xr-x 1 zarrelli zarrelli  174 Mar  8 11:03 simple-select.sh
-rwxr-xr-x 1 zarrelli zarrelli   63 Mar  7 09:26 until-simple.sh
-rwxr-xr-x 1 zarrelli zarrelli   64 Mar  7 08:54 while-simple.sh
```



```
zarrelli@moveaway: ~
zarrelli@moveaway: ~ 80x24

Width is: Height is:
```

Chapter 7: Plug into the Real World



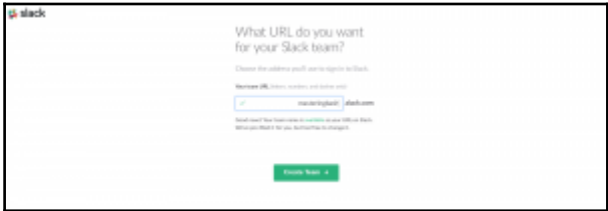
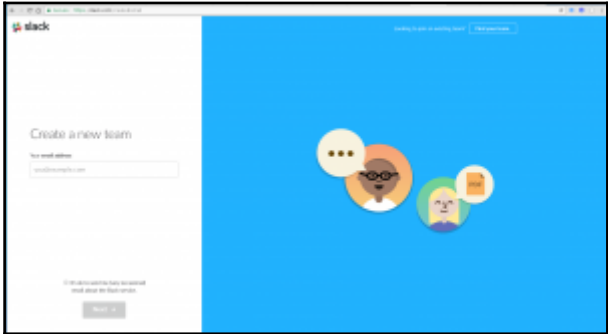
Service	State	Age	Health	Check Type	Check Latency	Check Duration
Service 1	OK	1m 30s	OK	ACTIVE	0.000s	0.000s
Service 2	OK	1m 30s	OK	ACTIVE	0.000s	0.000s
Service 3	OK	1m 30s	OK	ACTIVE	0.000s	0.000s
Service 4	OK	1m 30s	OK	ACTIVE	0.000s	0.000s
Service 5	OK	1m 30s	OK	ACTIVE	0.000s	0.000s

Service State Information

Current State: OK (for 16 0h 31m 15s)
State Information: CR - load average: 0.00, 0.01, 0.06
Performance Data: load1=0.00, 0.00, 0.00, 0.00, load5=0.00, 0.00, 0.00, 0.00, load15=0.00, 0.00, 0.00, 0.00, 0.00
Current Attempt: 1/4 (HAPD state)
Last Check Time: 2017-02-15 05:01:11
Check Type: ACTIVE
Check Latency / Duration: 0.000 / 0.000 seconds
Next Scheduled Check: 2017-02-15 05:06:11
Last State Change: 2017-02-14 04:21:17
Last Notification: HA (notification 0)
In This Service Flapping? NO (0.00% state change)
In Scheduled Downtime? NO
Last Updated: 2017-02-15 05:02:31 (0d 0h 0m 1s ago)

Active Checks: ENABLED
Passive Checks: ENABLED
Observing: ENABLED
Notification: ENABLED
Event Handler: ENABLED
Flap Detector: ENABLED

Chapter 8: We Want to Chat



[Browse apps](#) > [Custom Integrations](#) > [Incoming WebHooks](#) > [New configuration](#)

Incoming WebHooks

Send data into Slack in real-time.

Incoming WebHooks are a simple way to post messages from external sources into Slack. They make use of normal HTTP requests with a JSON payload, which includes the message and a few other optional details described later.

[Message Attachments](#) can also be used in Incoming WebHooks to display richly-formatted messages that stand out from regular chat messages.

⚠️ New to Slack Integrations?

Check out our [Getting Started](#) guide to familiarize yourself with the most common types of integrations, and tips to keep in mind while building your own. You can also [register as a developer](#) to let us know what you're working on, and to receive future updates to our APIs.

Post to Channel

Start by choosing a channel where your Incoming Webhook will post messages to.

Choose an option...

Choose a channel

Recently updated

Channels

#general

#random

#test

Direct Messages

@channel

[Browse apps](#) > [Custom Integrations](#) > [Incoming WebHooks](#) > [New configuration](#)

Incoming WebHooks

Added by gjojo on Mar 24th, 2017

[Duplicate](#) [Remove](#)

Incoming WebHooks are a simple way to post messages from external sources into Slack. They make use of normal HTTP requests with a JSON payload, which includes the message and a few other optional details described later.

[Message Attachments](#) can also be used in Incoming WebHooks to display richly-formatted messages that stand out from regular chat messages.

⚠️ New to Slack Integrations?

Check out our [Getting Started](#) guide to familiarize yourself with the most common types of integrations, and tips to keep in mind while building your own. You can also [register as a developer](#) to let us know what you're working on, and to receive future updates to our APIs.

Setup Instructions [View](#)

Will guide you through the steps necessary to configure an Incoming Webhook so you can start sending data to Slack.

Webhook URL [Copy](#)

<https://hooks.slack.com/services/T4FF9P9/444832041129443202/46469C11a>

Sending Messages

You have two options for sending data to the Webhook URL above:

- Send a JSON object as the `payload` parameter in a POST request
- Send a JSON string as the body of a POST request

For a simple message, your JSON payload could contain a `text` property of minimum. This is the text that will be posted to the channel.

A simple example:

```
payload={
  "text": "This is a line of text in a channel."
}
```

This will be displayed in the channel as:

Testing text 11:11 AM

This is a line of text in a channel.

And this is another line of text.

Message Formatting

Documentation

Message Builder

Here's a way for you to test your first Slack integration!

1. Enter your message as JSON

```
1 {  
2   "text": "This is a bold string and this a bold word."  
3 }
```

Examples: [Basic formatting](#) | [Attachments](#) | [Message buttons](#)

2. Preview your message

Clean Compact

 **Your App** APP 12:00 AM
This is a **bold string** and this a **bold** word.

Message Formatting

Documentation

Message Builder

Here's a way for you to test your first Slack integration!

1. Enter your message as JSON

```
1 {  
2   "text": "This is a bold string and this a bold word."  
3 }
```


Examples: [Basic formatting](#) | [Attachments](#) | [Message buttons](#)

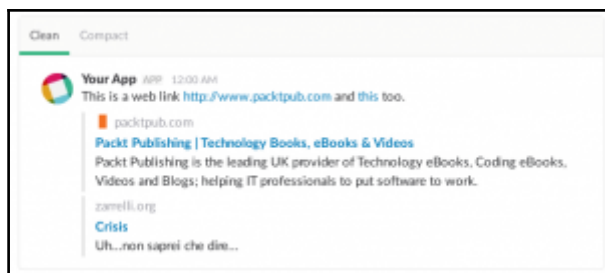
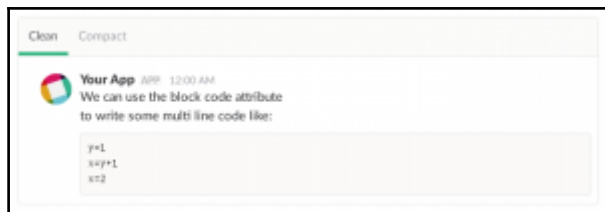
2. Preview your message

Clean Compact

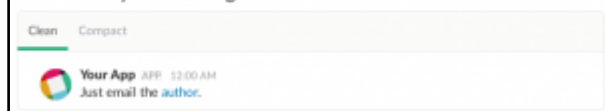
 **Your App** APP 12:00 AM
This is a **bold string** and this a **bold** word.

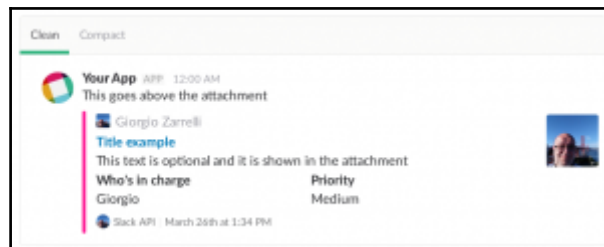
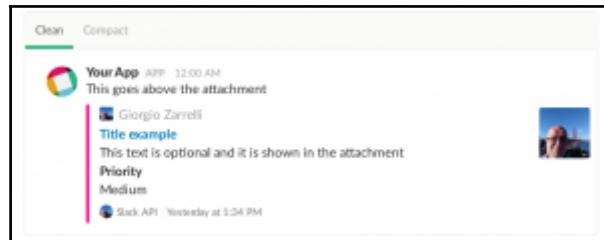
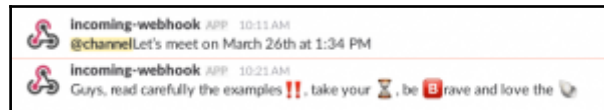
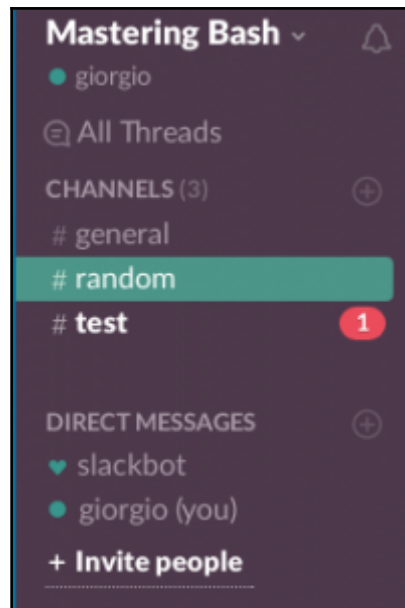
Clean Compact

 **Your App** APP 12:00 AM
We can use the code attribute
to write some code like:
`<code>xxx</code>`





2. Preview your message







This goes above the attachment

 Giorgio Zarrelli
Title example
 This text is optional and it is shown in the attachment
 Priority
 Medium


 Stack-API · March 26th at 1:34 PM


This goes above the attachment

 Giorgio Zarrelli
Title example
 This text is optional and it is shown in the attachment
 Priority
 Medium


 Stack-API · March 26th at 1:34 PM


This goes above the attachment


 Giorgio Zarrelli
Title example
 This text is optional and it is shown in the attachment
 Priority
 Medium

 Stack-API · March 26th at 1:34 PM


This goes above the attachment


 Giorgio Zarrelli
Title example
 This text is optional and it is shown in the attachment
 Priority
 Medium


 Stack-API · March 26th at 1:34 PM

 Incoming-webhook APP 11:15 AM


This goes above the attachment


 Giorgio Zarrelli
Title example
 This line of text is optional
 Priority
 Medium

 Stack-API · March 26th at 1:34 PM

 Incoming-webhook APP 11:48 AM

Announcement:

 Giorgio Zarrelli
New message
 This line of text is optional
 Priority
 Medium

 Mastering Bash · March 26th at 1:34 PM

Announcement:

 Me
My cli test
 This line of text is optional
 Priority
 Medium

 Mastering Bash · March 26th at 1:34 PM


```

Welcome
to our
restricted
sftp server

-----
[W|A|R|N|I|N|G]
-----
|T|h|i|s| |s|e|r|v|i|c|e| |i|s| |r|e|s|t|r|i|c|t|e|d| |t|o|
|a|u|t|h|o|r|i|z|e|d| |w|s|e|r|s| |o|n|l|y|. |A|l|l| |a|c|t|i|v|i|t|i|e|s|
|o|n| |t|h|i|s| |s|y|s|t|e|m| |a|r|e| |l|o|g|g|e|d|.
-----
```

```

File Edit View Terminal Tabs Help
root@spoton:/opt/sftp-jails# figlet -cf alligator test
::::::::::::::::::::::::::::::::::::::::::::::::::
|+|      |+|      |+|      |+|      |+|
+!+      +!+      +!+      +!+      +!+
+#+      +#+:++#      +#+:++#+      +#+
+#+      +#+      +#+      +#+      +#+
#+#      +##      +##      +##      +##
###      #####      ###
root@spoton:/opt/sftp-jails# █
```