

Narration	Time
اسپوکن ٹیوٹوریل میں خوش آمدید۔ Bash پر Command line arguments and Quoting کے	00:01
اس ٹیوٹوریل میں، ہم سیکھیں گے	00:08
'Command line Arguments' * اور	00:11
Quoting *	00:13
اس ٹیوٹوریل کو سمجھنے کیلئے آپ کو 'Linux Operating System' سے واقف ہونا چاہئے۔	00:15
اگر نہیں، تو براہ کرم متعلقہ ٹیوٹوریل کے لئے ہماری ویب سائٹ ملاحظہ کریں۔	00:20
اس ٹیوٹوریل کے لئے میں	00:26
Ubuntu Linux 12.04 OS اور	00:29
GNU Bash version 4.1.10 استعمال کر رہا ہوں	00:33
عمل کیلئے '4' 'GNU Bash' یا اس سے اعلیٰ ورژن کی صلاح دی جاتی ہے	00:37
'Shell Script' * کمانڈ لائن ' سے arguments لیتا ہے۔	00:43
ایک argument کو کال کئے جانے والے program کو پاس کیا جاتا ہے	00:46
کسی بھی تعداد میں arguments کو program میں پاس کیا جاسکتا ہے۔	00:52
اب اپنے ' کی بورڈ پر Ctrl Alt اور T کی زربا کر 'ٹرینٹل' کھولیں۔	00:57
میں نے کوڈ کو پہلے ہی 'arg.sh' فائل میں لکھا ہے۔	01:06
'ٹرینٹل' پر یہ 'ٹائپ کر کے، میں اس فائل کو کھولتا ہوں،	01:12
gedit space arg.sh space ampersand sign(&).	01:16
ہم ampersand کا استعمال پرومپٹ کو صاف کرنے کیلئے کرتے ہیں	01:23
اب اینٹر دباؤں۔	01:27
ٹیکسٹ ایڈیٹر کھلتا ہے۔	01:30
اب میں کوڈ سمجھاتا ہوں	01:33
یہ shebang لائن ہے۔	01:36
یہ لائن zeroth argument پر پرنٹ کرے گی	01:39
یہاں، \$0 (Dollar zero) شیل سکرپٹ کا نام پرنٹ کرے گا۔	01:43
اس کا یہ مطلب ہے کہ 'zeroth argument' خود 'پروگرام کا نام ہے۔	01:48
اب پروگرام کو ایک کیوٹ کریں اور دیکھیں۔	01:55

ٹرمینل پر جائیں	01:59
سب سے پہلے فائل کو ایکزیکوٹبل بنانے کیلئے ٹائپ کریں	02:01
chmod space plus x space arg.sh	02:05
اینٹر دبائیں۔	02:12
اب ٹائپ کریں dot slash arg.sh	02:14
اینٹر دبائیں۔ آؤٹ پٹ اس طرح لگ رہا ہے : Zeroth argument is arg.sh	02:18
اب واپس اپنے ایڈیٹر پر آئیں اور یہاں دکھائی گئی تین لائنیں لکھیں۔	02:26
،\$1 (Dollar one) ، command line سے program کو پاس کئے گئے argument کی نمائندگی کرتا ہے	02:33
، '\$2 (Dollar two) ، پروگرام کو پاس کئے گئے دوسرے argument کی نمائندگی کرتا ہے۔	02:40
اور ڈالر 3 تیسرے argument کی نمائندگی کرتا ہے۔	02:44
اب سیو پر کلک کریں۔	02:48
اب پروگرام کو ایکزیکوٹ کریں اور دیکھیں۔	02:49
اپ-ایرو کی کوڈ بائیں، اینٹر دبائیں۔	02:52
ہم دیکھتے ہیں کہ 'Zeroeth 'Argument' پرنٹ کیا گیا ہے۔	02:57
لیکن پہلا، دوسرا اور تیسرا argument خالی ہے۔	03:00
وجہ یہ ہے کہ 'Command line Arguments' کو ایکزیکوٹن کے وقت دیا جاتا ہے۔	03:05
تو اپ ایرو کی دبائیں اور ٹائپ کریں sunday monday اور tuesday	03:11
اینٹر دبائیں۔	03:18
آپ دیکھ سکتے ہیں کہ پہلا، دوسرا اور تیسرا argument sunday monday اور tuesday ہیں	03:21
اب اپنے ایڈیٹر پر واپس آئیں اینٹر دبائیں۔	03:28
اب یہاں دکھائی گئی لائن ٹائپ کریں۔	03:33
، \$12 (Dollar twelve) ، بارہویں argument کی نمائندگی کرتا ہے	03:37
9 سے زائد ایک argument لکھنے کے لئے، ہمیں کرلی بریکٹس 'استعمال کرنے ہوں گے	03:41
ورنہ 'Bash' صرف 'ten's کی جگہ 'انچر کے argument لے گا	03:46
اور آپ کو متوقع آؤٹ پٹ نہیں ملے گا۔	03:53
اب سیو کریں 'پر کلک کریں۔	03:57
اب پروگرام ایکزیکوٹ کریں	03:59

04:01	ٹرمینل 'کھولیں'۔
04:04	اب میں پرومپٹ صاف کرتا ہوں۔
04:07	اب پروگرام کو 12 یا 13 arguments دینے کی ضرورت ہے۔
04:12	تو ٹائپ کریں 'dot slash arg.sh space 1 to 13' اینٹر دبائیں۔
04:23	آپ دیکھ سکتے ہیں کہ بارہواں argument، 12 ہے۔
04:27	اپنے ایڈیٹر پر واپس آئیں
04:30	اور یہاں دکھائی گئی لائن ٹائپ کریں۔
04:34	'\$ #' (ڈالر بیش) ، arguments کی کل تعداد دیتا ہے جو پروگرام کو پاس کئے گئے ہوں
04:40	اب سیو پر کلک کریں۔
04:43	اب ایکزیکوٹ کریں
04:44	کھولیں ٹرمینل۔
04:46	اب ایکزیکوٹ کریں۔ اپ-ایرو کی دبائیں اور اینٹر دبائیں۔
04:52	ہم دیکھ سکتے ہیں کہ گل arguments '13' ہیں۔
04:57	اب ایڈیٹر پر جائیں۔
05:00	اینٹر دبائیں اور یہاں دکھائے جانے والی لائنیں لکھیں۔
05:04	'\$*' (Dollar asterisk) ، ایک ہی لائن میں سارے arguments پرنٹ کرے گا
05:10	ہم اسے ایک سادہ for loop کی مدد سے جانچیں گے
05:14	ہم ایکزیکوٹ کے وقت اس لوپ کا تجزیہ کریں گے۔
05:18	اب 'سیو' پر کلک کریں۔ ٹرمینل پر جائیں
05:22	اب میں پرومپٹ صاف کرتا ہوں۔
05:26	اب ٹائپ کریں 'dot slash arg.sh space sunday monday and tuesday'
05:35	اینٹر دبائیں۔
05:38	گل arguments '3' ہیں کیونکہ آپ دیکھ سکتے ہیں کہ ہم نے اپنے پروگرام میں 3 arguments پاس کیے ہیں۔
05:46	جیسا کہ پہلے ذکر کیا گیا ہے، * \$ ایک لائن میں arguments پرنٹ کرے گا
05:54	اور یہ for loop کے لئے آؤٹ پٹ ہے۔
05:57	ہم دیکھتے ہیں کہ ایک ہی لائن میں تمام arguments پرنٹ کئے گئے ہیں۔
06:02	اب اپنے پروگرام میں واپس جائیں اور یہاں دکھائی گئی لائنوں جیسے ہی ٹائپ کریں

@ (Dollar at) بھی سارے آرگیومنٹس پرنٹ کرے گا	06:09
تاہم، اس وقت ہر argument الگ الگ لائن پر پرنٹ ہوگا	06:13
یہ ایک اور for 'لوپ' ہے، جو ہر 'argument' کو مختلف لائنوں میں پرنٹ کرے گا	06:20
اب دیکھتے ہیں کہ کیسے۔ سیو پر کلک کریں۔	06:26
ٹرینل پر آئیں	06:29
up-arrow key دبائیں	06:32
اینٹر دبائیں، اب آپ فرق دیکھ سکتے ہیں۔	06:34
یہ '@ \$' کی طرف سے پرنٹ ہوئے arguments ہیں۔	06:39
ایک مختلف لائن میں '@ \$' ہر ایک 'argument' پرنٹ کرتا ہے۔	06:43
یہ دوسرے for loop کیلئے آؤٹ پٹ ہے	06:47
اب 'Bash' میں quoting پر جاتے ہیں	06:52
سلائیڈز پر جائیں	06:55
quotes کے تین اقسام ہیں۔	06:57
Double quote Single quote Backslash	06:59
Double quote ، variables اور commands کے ویلیوز میں تبدیلی کرتا ہے	07:02
مثلاً 'echo "Username is \$USER"'	07:09
یہ سسٹم کا username ظاہر کرتا ہے۔	07:13
'ٹرینل' پر جائیں۔	07:17
اب میں prompt صاف کرتا ہوں	07:20
اب ٹائپ کریں، echo space ڈبل کوٹس میں Username space is dollar کیپٹل حروف میں USER	07:23
اینٹر دبائیں۔ سسٹم کا یوزر نیم پرنٹ کیا جاتا ہے۔	07:34
آؤٹ پٹ آپ کے سسٹم کے مطابق بدل جائے گا۔	07:39
اب سلائیڈز پر واپس جائیں۔	07:42
'Single quotes' دی گئی string میں ہر کیریٹر کے لفظی معنی برقرار رکھتا ہے۔	07:46
یہ تمام حروف کے خاص معنی کو بند کرنے کے لئے استعمال کیا جاتا ہے۔	07:53
Terminal کھولیں۔	07:58
ٹائپ کریں: echo space سنگل کوٹس میں Username space is dollar کیپٹل حروف میں USER	08:01

08:10	اینٹرو بائیں.
08:12	آؤٹ پٹ Username is \$USER ہے.
08:16	اس مثال میں، یہ سنڈگل کوؤٹس میں موجود تمام حروف پرنٹ کرتا ہے.
08:23	یہ variable \$USER کی ویلیو نہیں بدلتا
08:28	اپنے سلائڈ پر واپس آئیں
08:31	Backslash ایک کیریٹر سے خاص معنی کو ہٹا دیتا ہے.
08:37	* یہ 'Bash' میں escape character کے طور پر استعمال کیا جاتا ہے.
08:42	اٹریبل 'پر جائیں.
08:44	اب ٹائپ کریں: echo space ڈبل کوؤٹس میں Username is backslash dollar کیپٹل حروف میں USER
08:55	چونکہ ہم نے ڈبل کوؤٹس لگائے ہیں، ہم امید کرتے ہیں کہ echo کمانڈ username دکھائے گی
09:02	اب اس کمانڈ کی کوشش کریں . تو اینٹرو بائیں.
09:06	آؤٹ پٹ ہے Username is \$USER
09:10	اس مثال میں backslash '\$' (Dollar) سائن کا خاص مطلب ہٹاتا ہے.
09:16	'\$ USER' کسی خاص فعالیت کے بغیر ایک string کے طور پر لیا جاتا ہے
09:22	اب یہ ٹیوٹوریل ختم ہوتا ہے
09:25	اپنے سلائڈ پر واپس آئیں
09:27	خلاصہ دیتے ہیں کہ ہم نے کیا سیکھا،
09:31	Command line arguments
09:33	ڈبل کوؤٹس، سنڈگل کوؤٹس اور بیک سلیش کی فنکشنلٹی یعنی فعالیت
09:39	ذیل لنک میں دستیاب ویڈیو دیکھیں.
09:42	یہ اسپوکن ٹیوٹوریل پروجیکٹ کا خلاصہ ہے.
09:45	اگر آپ کو اچھا بیڈ وڈ تھوڑا دستیاب نہیں تو، آپ اسے ڈاؤن لوڈ کر کے دیکھ سکتے ہیں.
09:51	اسپوکن ٹیوٹوریل پروجیکٹ ٹیم اسپوکن ٹیوٹوریلز کا استعمال کرتے ہوئے ورکشاپس منعقد کرتی ہے اور
09:56	آن لائن امتحان پاس کرنے والوں کو اسناد دیتے ہیں
10:00	مزید معلومات کے لئے، contact@spoken-tutorial.org پر لکھیں.
10:07	اسپوکن ٹیوٹوریل پروجیکٹ ٹاک تو اے ٹیچر پروجیکٹ کا ایک حصہ ہے.

یہ ہندوستانی حکومت کے آئی سی ٹی کے ذریعے قومی خواندگی مشن کی طرف سے حمایت شدہ ہے۔ اس مشن پر مزید معلومات http://spoken-tutorial.org/NMEICT-Intro پر دستیاب ہے۔	10:10
میں وجاحت احمد اب آپ سے رخصت لیتا ہوں ہمارے ساتھ شامل ہونے کیلئے شکریہ۔	10:30