

Narration	Time
Ruby میں Control Statements کے اسپوکن ٹیوٹوریل میں خوش آمدید۔	00:01
اس ٹیوٹوریل میں ہم مندرجہ ذیل استعمال کرنا سیکھیں گے	00:06
if statement	00:08
elsif statement	00:09
else	00:11
case statements	00:12
یہاں ہم	00:14
Ubuntu ورژن 12.04 اور	00:15
Ruby 1.9.3 استعمال کر رہے ہیں	00:18
اس ٹیوٹوریل کو سمجھنے کے لئے، آپ کے پاس انٹرنیٹ کنکیشن ہونا ضروری ہے۔	00:21
آپ کو Linux کمانڈس، Terminal اور ٹیکسٹ Text-editor کا علم بھی ہونا چاہئے۔	00:24
اگر نہیں، تو متعلقہ ٹیوٹوریل کے لئے، ہماری ویب سائٹ ملاحظہ کریں۔	00:30
شروع کرنے سے پہلے، یاد کریں کہ ہم نے پہلے ttt ڈائریکٹری بنائی تھی۔	00:34
اس ڈائریکٹری پر جاتے ہیں۔	00:38
اور پھر ruby hyphen tutorial، ruby hyphen statements control پر	00:41
اب ہم اس فولڈر میں ہیں، آگے بڑھتے ہیں۔	00:47
Ruby میں if statement کا سنٹیکس اس طرح ہوتا ہے:	00:52
if "condition"	00:56
'ruby' کوڈ	00:58
end	00:59
ایک مثال دیکھتے ہیں۔	01:01
gedit میں ایک نئی فائل بنائیں، جیسے شروعاتی Ruby ٹیوٹوریل میں دکھایا گیا ہے۔	01:03
اسے if hyphen statement dot rb نام دیں۔	01:08

01:12	میرے پاس if statement کی ایک فعال مثال ہے۔
01:15	اس کو کرتے وقت، ٹیوٹورل کو درمیان میں روک کر کوڈ ٹائپ کر سکتے ہیں۔
01:19	میں نے اس مثال میں ایک if statement ڈکلیئر کیا ہے۔
01:23	پہلے، میں ایک لوکل ویریبل my_num ڈکلیئر کرتا ہوں اور اس کو 2345 ویلیو دیتا ہوں۔
01:31	پھر ایک if statement ڈکلیئر کرتا ہوں۔
01:34	if سٹیٹمنٹ میں puts ڈکلیئر کیا ہوا میتھڈ یعنی طریقہ، آؤٹ پٹ ظاہر کرے گا۔
01:39	if سٹیٹمنٹ اسے جانچے گا کہ کیا my_num کی ویلیو 0 سے بڑی ہے
01:43	اگر ایسا ہے، تو یہ وہ مخصوص string پرنٹ کرے گا۔
01:47	اب، ٹرمینل پر جائیں اور ٹائپ کریں
01:51	ruby space if hyphen statement dot rb
01:57	The value of my_num is greater than 0. آؤٹ پٹ ظاہر ہوگا۔
02:02	یہ آؤٹ پٹ ثابت کرتا ہے کہ if کنڈیشن نے True رٹرن کیا ہے۔
02:07	اب آپ Ruby میں if سٹیٹمنٹ لکھنے کے قابل ہونے چاہئیں۔
02:12	اب آگے if-else سٹیٹمنٹ دیکھتے ہیں۔
02:16	else استعمال کرنے کے لئے سنٹیکس ہے:
02:18	if condition
02:19	ruby کوڈ
02:20	else
02:21	ruby کوڈ
02:22	end
02:24	ایک مثال کو دیکھتے ہیں۔
02:26	gedit میں ایک نئی فائل بنائیں، جیسے شروعاتی Ruby ٹیوٹوریل میں دکھائی گئی ہے۔
02:30	اسے if hyphen else hyphen statement dot rb نام دیں۔
02:37	میرے پاس if-else سٹیٹمنٹ کی ایک فعال مثال ہے۔

02:40	اس کو کرتے وقت، ٹیوٹورل کو درمیان میں روک کر کوڈ ٹائپ کر سکتے ہیں۔
02:44	میں نے اس مثال میں ایک if-else سٹیٹمنٹ ڈیکلئیر کیا ہے۔
02:48	پہلے، میں ایک لوکل ویریبل my_num ڈیکلئیر کرتا ہوں اور اس کو 1-ویلیو دیتا ہوں۔
02:55	پھر میں ایک if statement ڈیکلئیر کرتا ہوں۔
02:58	if سٹیٹمنٹ، my_num کی ویلیو اگر 0 سے بڑی ہے تو جانچے گا
03:03	اگر ایسا ہے، تو یہ وہ مخصوص string پرنٹ کرے گا۔
03:06	اگر نہیں، تو else سٹیٹمنٹ پر جائے گا
03:10	اور یہ اس string کو پرنٹ کرے گا جو وہاں مخصوص ہے۔
03:13	اب، ٹرمینل پر جاتے ہیں، اور ٹائپ کرتے ہیں
03:18	ruby space if hyphen else hyphen statement dot rb.
03:26	اور آؤٹ پٹ دیکھتے ہیں۔
03:27	The value of my_num is lesser than 0 آؤٹ پٹ ظاہر ہوگا
03:32	یہ، ایکریوٹ ہو else سٹیٹمنٹ ظاہر کرتا ہے۔
03:35	اب آپ Ruby میں if-else سٹیٹمنٹ لکھنے کے قابل ہونا چاہئیں۔
03:41	اب آگے if-elsif سٹیٹمنٹ دیکھتے ہیں۔
03:45	elsif استعمال کرنے کے لئے سنیکس ہے:
03:48	ruby، if condition کوڈ
03:50	ruby، elsif condition کوڈ
03:52	ruby، else کوڈ
03:54	end
03:55	ایک مثال دیکھتے ہیں۔
03:58	gedit میں ایک نئی فائل بنائیں، جیسے شروعاتی Ruby ٹیوٹوریل میں ظاہر ہے۔
04:01	اسے if hyphen elsif hyphen statement dot rb نام دیں۔
04:07	میرے پاس if-elsif سٹیٹمنٹ کی ایک فعال مثال ہے۔

04:10	اس کو کرتے وقت، ٹیوٹورل کو درمیان میں روک کر کوڈ ٹائپ کر سکتے ہیں۔
04:14	میں نے اس مثال میں ایک if-elsif سٹیٹمنٹ ڈیکلئیر کیا ہے۔
04:19	یہاں بھی، میں ایک لوکل ویریبل my_num ڈیکلئیر کرتا ہوں اور اس کو -1 ویلیو دیتا ہوں۔
04:25	پھر میں ایک if statement ڈیکلئیر کرتا ہوں۔
04:28	if سٹیٹمنٹ، my_num کی ویلیو اگر 0 سے بڑی ہے تو جانچے گا
04:32	اگر ایسا ہے، تو یہ وہ مخصوص string پرنٹ کرے گا۔
04:35	اگر یہ True نہیں ہے تو یہ elsif سیکشن میں جائے گا
04:39	اب، اگر my_num کی ویلیو -1 ہے، تو یہ جانچے گا۔
04:43	اگر یہ True ہے تو یہ اس string کی ویلیو پرنٹ کرے گا جو وہاں مخصوص ہے۔
04:46	my_num کی ویلیو نہ تو 0 سے بڑی ہے اور نہ ہی -1 کے برابر ہے، تو یہ else سیکشن میں جائے گا
04:54	چونکہ my_num کی ویلیو -1 ہے تو یہ else بلاک میں آگے نہیں بڑھے گا۔
05:00	اور یہ conditional سٹیٹمنٹ سے باہر جائے گا۔
05:03	اب ٹرمینل پر جائیں اور ٹائپ کریں۔
05:07	ruby space if hyphen elsif hyphen statement dot rb
05:15	اور آؤٹ پٹ دیکھیں۔
05:17	The value of my_num is -1 and is lesser than 0. آؤٹ پٹ ظاہر ہوگا۔
05:23	واپس اپنی فائل پر جاتے ہیں اور my_num کی ویلیو کو 5 سے بدلتے ہیں۔
05:29	کوڈ کو سیو کریں اور اسکو ٹرمینل پر ایکزیکوٹ کریں۔
05:35	تو اب یہ if کنڈیشن کو پورا کرتا ہے اور وہ مخصوص سٹرنگ پرنٹ ہوئی ہے۔
05:42	my_num کی ویلیو 0 سے بڑی ہے۔
05:45	اپنی فائل پر واپس آتے ہیں اور my_num کی ویلیو کو -5 سے بدلتے ہیں۔
05:50	کوڈ کو سیو کریں اور اسکو ٹرمینل پر ایکزیکوٹ کریں۔
05:55	اس کیس میں یہ else کنڈیشن کو پورا کرتا ہے اور else بلاک کے اندر puts سٹیٹمنٹ ایکزیکوٹ ہوتا ہے۔
06:03	اب آپ Ruby میں if-elsif سٹیٹمنٹ لکھنے کے قابل ہونے چاہئیں۔

06:08	اب، case کا اسٹیٹمنٹ دیکھتے ہیں۔
06:12	case کا اسٹیٹمنٹ ایک خاص سلیکشن پر مبنی control flow اسٹیٹمنٹ ہے۔
06:17	اس اسٹیٹمنٹ کو سمجھنے کے لئے ہم case اسٹیٹمنٹ کا سنٹیکس دیکھتے ہیں۔
06:22	case استعمال کرنے کے لئے سنٹیکس:
06:24	case variable
06:26	when value 1
06:28	ruby code
06:29	when value 2
06:30	ruby code
06:31	else
06:32	ruby code
06:34	end
06:35	اب ایک مثال دیکھتے ہیں۔
06:37	gedit میں ایک نئی فائل بنائیں جیسا Ruby کے ابتدائی ٹیوٹوریلز میں دکھایا گیا ہے۔
06:41	اس کو case hyphen statement dot rb نام دیں۔
06:44	میرے پاس case اسٹیٹمنٹ کی ایک فعال مثال ہے۔
06:48	اس کو کرتے وقت، ٹیوٹوریل کو درمیان میں روک کو ڈائپ کر سکتے ہیں
06:52	اس مثال میں، میں نے ایک case کا اسٹیٹمنٹ ڈکلیئر کیا ہے۔
06:55	یہاں میرے پاس ایک print اسٹیٹمنٹ ہے جو ٹرمینل پر ایک سوال پرنٹ کرے گا۔
07:01	پھر میں gets کو کال کرتا ہوں، جو اسٹنڈرڈ ان پٹ سے ڈیٹا کی ایک لائن کو قبول کرے گا۔
07:09	پھر میں.chomp استعمال کرتے ہوئے new line characters کے ان پٹ ڈیٹا کو کاٹتا ہوں۔
07:15	میں نیچے domain نامی ویریبیل کو دیتا ہوں۔
07:18	پھر میں ایک case کا اسٹیٹمنٹ ڈکلیئر کرتا ہوں۔
07:22	اس کے اندر میں ایک when اسٹیٹمنٹ ڈکلیئر کرتا ہوں۔

07:25	یہ جانچنا ہے کہ، وہ مخصوص سٹرنگ، domain کی ویلیو کی طرح ہے یا نہیں۔
07:30	پہلے، یہ جانچنا ہے کہ، domain کی ویلیو UP ہے یا نہیں۔
07:34	اگر ایسا ہے تو، یہ Uttar Pradesh پرنٹ کرے گا اور case کا اسٹیٹمنٹ چھوڑ دیگا
07:39	اگر domain UP نہیں ہے تو، یہ جانچنا ہے کہ domain کی ویلیو MP ہے یا نہیں۔
07:44	اگر ایسا ہے تو، یہ Madhya Pradesh پرنٹ کرے گا اور آگے ایسے ہی چلے گا۔
07:48	اگر دور تک کوئی مماثلت نہیں ہو تو یہ domain کی ویلیو جانچتا رہے گا۔
07:53	اس پوائنٹ پر یہ else اسٹیٹمنٹ کا سامنا کرے گا، اگر مندرجہ بالا میں سے کوئی بھی کنڈیشن True نہ ہو۔
07:59	یہ ruby code کو ایکز کیوٹ کرے گا جو else ڈیکلریشن کے بعد آتا ہے۔
08:03	ہماری مثال میں یہ Unknown پرنٹ کرے گا۔
08:07	اب سیو کریں، ٹرینل پر جائیں اور ٹائپ کریں
08:11	ruby space case hyphen statement dot rb.
08:18	Enter the state you live in ٹرینل پر ظاہر ہوگا
08:22	UP ٹائپ کریں، اور آؤٹ پٹ دیکھیں۔
08:25	آؤٹ پٹ ظاہر ہوگا Uttar Pradesh
08:28	اب، پہلے کی طرح ہی، Ruby فائل کو دوبارہ ایگز کیوٹ کریں۔
08:31	اس وقت پرومپٹ پر ٹائپ کریں KL اور آؤٹ پٹ دیکھیں۔
08:36	یہ kerala پرنٹ کرے گا۔
08:38	مزید، ایک بار پھر فائل کو ایکز کیوٹ کریں۔
08:41	اس وقت پرومپٹ پر ٹائپ کریں TN اور آؤٹ پٹ دیکھیں۔
08:47	یہ Unknown پرنٹ کرے گا۔
08:50	یہ اس لئے ہے کیونکہ کوئی بھی case مطمئن نہیں ہوا، لہذا ڈفالٹ طور پر else اسٹیٹمنٹ ایکز کیوٹ ہوا ہے۔
08:58	اب آپ Ruby میں اپنا case-statements لکھنے کے قابل ہونا چاہئیں۔
09:03	اب ہم اس ٹیوٹوریل کے اختتام پر آگئے ہیں۔
09:07	اس کا خلاصہ کرتے ہیں۔

اس ٹیوٹوریل میں ہم نے	09:08
if statement	09:10
else construct	09:12
اور if-elsif	09:13
case statements استعمال کرنا سیکھا	09:15
مشق کے طور پر:	09:17
Ruby پروگرام ٹائپ کریں	09:18
یہ یوزر کو ایک نمبر یعنی تعداد داخل کرنے کو کہتا ہے	09:20
پھر جانچنے کے لئے کہ آیا نمبر، 2 کا ملٹپل ہے، مناسب control-statement استعمال کریں	09:23
اگر ایسا ہے تو، پرنٹ کرے گا "The number entered is a multiple of 2"	09:29
اگر نہیں، تو جانچے گا کہ کیا یہ 3 کا ملٹپل ہے۔	09:35
اگر ایسا ہے تو، پرنٹ کرے گا "The number entered is a multiple of 3"	09:38
اگر نہیں، تو جانچے گا کہ کیا یہ 4 کا ملٹپل ہے۔	09:43
اگر ایسا ہے تو، پرنٹ کرے گا "The number entered is a multiple of 4"	09:47
اگر نہیں، تو یہ پرنٹ کرنا چاہئے "The number is not a multiple of 2, 3 or 4"	09:51
مندرجہ ذیل لنک پر دستیاب ویڈیو دیکھیں۔	09:56
یہ اسپوکن ٹیوٹوریل پروجیکٹ کا خلاصہ کرتا ہے۔	10:00
اچھی بینڈ ویڈیو نہ ملنے پر آپ اسکوڈ اوون لوڈ کر کے دیکھ سکتے ہیں۔	10:03
اسپوکن ٹیوٹوریل پروجیکٹ ٹیم:	10:07
اسپوکن ٹیوٹوریل کا استعمال کرتے ہوئے ورکشاپ چلاتی ہے۔	10:09
اور آن لائن ٹیسٹ پاس کرنے والوں کو ٹیٹیکٹ دیتے ہیں۔	10:13
مزید معلومات کے لئے، <a href="mailto:contact@spoken-tutorial.org">contact@spoken-tutorial.org</a> پر لکھیں۔	10:16
اسپوکن ٹیوٹوریل ٹاک ٹوائے ٹیچر پراجیکٹ کا حصہ ہے۔	10:21
یہ بھارتی حکومت کے ایم ایچ آر ڈی کے آئی سی ٹی کے ذریعے قومی خواندگی مشن کی طرف سے حمایت شدہ ہے۔	10:26

اس مشن پر مزید معلومات spoken hyphen tutorial dot org slash NMEICT hyphen Intro پر دستیاب ہے۔	10:32
اس اسکرپٹ کا ترجمہ اور صدا بندی میں نے یعنی وجاحت احمد نے کی ہے۔ شامل ہونے کیلئے آپ کا شکریہ۔	10:41