

00:02 جاوا میں Nested-If اور Ternary Operator کے سپوکن ٹیوٹوریل میں آپ کا استقبال ہے۔

00:07 اس ٹیوٹوریل کے کرنے سے آپ مندرجہ ذیل کو کرنے کے قابل ہوں گے:

Nested-If سٹیٹمیٹس اور Ternary آپریٹرز کی وضاحت کرنے میں۔

ایک جاوا پروگرام میں ان کا استعمال کرنے میں .

00:17 اس ٹیوٹوریل کے لئے ہم استعمال کریں گے:

،Ubuntu v 11.10

JDK 1.6، اور

EclipseIDE 3.7.0

00:27 اس ٹیوٹوریل کو سمجھنے کے لئے، آپ کو درج ذیل کا علم ہونا چاہئے،

00:29 relational اور logical آپریٹرز

00:33 اور if ... else کنٹرول سٹیٹمیٹس کی روانی کے استعمال کے بارے میں۔

00:36 اگر ایسا نہیں ہے، تو مناسب ٹیوٹوریل کے لئے ہماری مندرجہ ذیل ویب سائٹ پر جائیں۔

00:41 Nested if سٹیٹمیٹ، ایک If سٹیٹمیٹ کے اندر دوسرا if سٹیٹمیٹ، nested-if سٹیٹمیٹ کہلاتا ہے۔

00:49 اب Nested-If سٹیٹمیٹ لکھنے کے لئے سنٹیکس کو دیکھتے ہیں۔

00:53 اس صورتحال میں، اگر کنڈیشن true 1 ہے تو پروگرام کنڈیشن 2 کے لئے جانچ کرتا ہے۔

00:59 کنڈیشن 2 دوسرے If سٹیٹمیٹ کا استعمال کرتے ہوئے دی جاتی ہے۔

01:03 اگر کنڈیشن true 2 ہے تو پروگرام سٹیٹمیٹ یا بلاک 1 کو ایکزیکوٹ کرتا ہے۔

01:09 دوسری صورت میں، یہ سٹیٹمیٹ یا بلاک 2 کو ایکزیکوٹ کرتا ہے۔

01:13 اگر کنڈیشن 1 فالس ہے، تو پروگرام کنڈیشن 2 کی جانچ نہیں کرے گا۔

01:18 اس کے بجائے براہ راست else سٹیٹمیٹ یعنی بلاک 3 پر پہنچ جائے گا۔

01:24 اب اسے بہتر سمجھنے کے لئے ایک مثال لیتے ہیں

01:28 ہمارے پاس باقی کوڈ کے لئے ضروری eclipse IDE اور skeleton ہے۔

01:32 ہم نے ایک کلاس NestedIfDemo بنا لیا ہے اور اسے مین میتھڈ میں شامل کر دیا ہے۔

01:37 ہم جانچ کریں گے کہ دی گئی تعداد ایک اور تعداد ہے یا ایک اوڈ تعداد ہے۔

01:42 ہم ایک nested-if کا استعمال کرتے ہوئے نیگیٹو یعنی منفی اعداد کو بھی ہینڈل کریں گے۔

01:46 لہذا میں میتھڈ میں ٹائپ کریں

int n = minus 5; 01:49

01:54 ہم نے منفی تعداد کو سٹور کرنے کے لئے ویریبل n بنا لیا ہے۔

01:58 اب ہم if کنڈیشن لکھیں گے۔

02:01 اگلی لائن میں ٹائپ کریں

if (n < 0) 02:02

02:07 کر لی بریکٹ کھولیں۔ اینٹرڈ بائیں۔

02:10 System.out.println بریکٹس اور ڈبل کوٹس کے اندر ("Negative number")؛

02:22 ہم پہلے دیکھتے ہیں کہ کیا تعداد ایک منفی تعداد ہے۔

02:25 اگر ہاں تو ہم اوڈ اور اون کے لئے جانچ نہیں کریں گے۔

02:29 اگر تعداد منفی نہیں ہے، تو ہم اون اور اوڈ کے لئے جانچ کریں گے۔

02:34 اگلی لائن میں ٹائپ کریں۔

else[

] اینٹرڈ بائیں

02:42 اب اگر ایکزیکوشن else حصہ پر آ گیا ہے۔

02:45 اس کا مطلب ہے کہ تعداد غیر منفی ہے۔

02:48 لہذا ہم اس else حصہ کے اندر اوڈ یا اون کے لئے جانچ کریں گے۔

02:52 ٹائپ کریں

02:53 if (n modulus 2 double equal to 0) { ، اینٹرڈ بائیں

System.out.println("Even number"); 03:03

}

else {

اینٹرڈ بائیں

ٹائپ کریں

System.out.println("Odd number");

}

03:29 ہم اس بات کو یقینی بنائیں گے کہ اوڈ یا اون نمبرس یعنی اعداد کی جانچ کرنے کے لئے منفی نمبرس پر غور نہیں کیا جاتا ہے۔

03:34 اب ہم کوڈ کے عمل کو دیکھتے ہیں۔

03:37 فائل سیوا اوررن کریں . جیسا کہ ہم دیکھ سکتے ہیں، ہمیں آؤٹ پٹ کے طور پر "Negative Number" (منفی تعداد) حاصل ہوتی ہے۔

03:43 اب ایک پوزیٹو یعنی مثبت تعداد لیتے ہیں۔

03:46 $n = -5$ کو $n = 5$ سے تبدیل کریں۔

03:53 اب فائل کو سیوا اوررن کریں۔

03:57 جیسا کہ ہم دیکھ سکتے ہیں، آؤٹ پٹ توقع کے مطابق (تعداد odd number) ہے . ایک درمیانی تعداد (even number) لیتے ہیں۔

04:04 $n = 5$ کو $n = 10$ سے تبدیل کریں۔

04:09 اب فائل کو سیوا اوررن کریں۔

04:12 جیسا کہ ہم دیکھ سکتے ہیں، آؤٹ پٹ توقع کے مطابق even number ہے۔

04:17 ایک if سٹیٹمنٹ کے اندر دوسرے کو شامل کرنے کی یہ عمل، nested-if کہلاتی ہے۔

04:22 nesting کی تعداد کی کوئی حد نہیں ہے۔

04:25 لیکن nesting کے تین درجے سے آگے نہ بڑھنا ایک اچھی عمل ہے۔

04:31 اب ہم ternary operator کو دیکھیں گے۔

04:33 پہلے میں مین میتھڈ کو ہٹاتا ہوں۔

04:37 ہم ایک پروگرام لکھیں گے جو تعداد کو 2 سے تقسیم کرتا ہے۔

04:40 یہ ایک بہت چھوٹا پروگرام ہے لیکن اوڈنمبرس کو تقسیم کرنے میں مسئلہ میں آتا ہے۔

04:45 جب 7، 2 سے تقسیم کیا جاتا ہے، تو ہمیں 3 حاصل ہوتا ہے۔

04:48 لیکن کیا ہوتا ہے، جب ہم نتائج کو رائونڈ آف صورت میں چاہتے ہوں

04:50 جس کا مطلب ہے، جب 7 کو 2 سے تقسیم کریں تو ہمیں 4 حاصل ہوتا ہے اور 3 نہیں۔

04:56 آسان الفاظ میں، ہم اگلی تعداد لیتے ہیں۔

04:59 ہم دیکھتے ہیں کہ اس طرح کا پروگرام کیسے لکھیں؟

05:01 مین میتھڈ کے اندر دیکھیں اور ٹائپ کریں ; int n, nHalf

05:08 ہم n میں تعداد اور nHalf میں آدھی تعداد کو سٹور کریں گے۔

05:13 اگلی لائن میں ٹائپ کریں ; n = 5

05:18 اگلی لائن میں ٹائپ کریں { if (n % 2 == 0) ، اینٹرڈ بائیں

05:28 ٹائپ کریں

nHalf = n / 2;

}

else {

nHalf = (n + 1) / 2;

}

05:50 ہم جانچ کرتے ہیں کہ تعداد اوڈ ہے یا اون اور اس کے مطابق تقسیم کرتے ہیں۔

05:55 اب پروگرام کے عمل کو دیکھنے کے لئے ایک پرنٹ سٹیٹمنٹ جوڑتے ہیں۔

05:59 لہذا ٹائپ کریں `System.out.println(nHalf);`

06:11 اب فائل کو سیو اور رن کریں

06:14 جیسا کہ ہم دیکھ سکتے ہیں، ہمارا مقصد مکمل ہو جاتا ہے۔ ہمیں آؤٹ پٹ میں 3 حاصل ہوتا ہے اور 2 نہیں۔

06:21 لیکن اگر ہم توجہ دیں، تو جو کچھ جو ہم کرتے ہیں وہ ہے ایک کنڈیشن کی بنیاد پر ویریبل کی ویلیو مقرر کرنا ہے

06:27 ہمارے پروگرام میں لوچک یعنی منطق سے زیادہ سٹینیکس ہے۔

06:31 یہ تب ہوتا ہے جب ternary آپریٹر کوڈ کو سمپل یعنی سادہ بناتا ہے۔

06:35 Ternary آپریٹر ایک کنڈیشنل آپریٹر ہے جو nested-if کی طرح نتائج دیتا ہے۔

06:40 یہ ایک چھوٹا سٹینیکس فراہم کرتا ہے اور ایک کوپیشن مارک (سوالیہ نشان) کے طور پر ظاہر کیا جاتا ہے۔

06:45 یہ ایک بار میں تین آپرینڈس لیتا ہے۔

06:48 Ternary آپریٹر کے سٹینیکس کے بارے میں سیکھتے ہیں۔

06:53 ایکسپریشن وہ کنڈیشن ہے جس کی جانچ کی جانی ہے۔

06:56 اگر کنڈیشن true ہے تو آپرینڈ 1 ویریبل نتائج کی ویلیو ہے۔

07:03 اگر کنڈیشن فالس ہے تو آپرینڈ 2 ویلیو ہے۔

07:09 اب اسے اپنے پروگرام میں استعمال کرتے ہیں۔

07:12 پہلے ہم if-else سٹیٹمنٹ کو ہٹاتے ہیں۔

07:17 ٹائپ کریں `nHalf = n % 2 == 0 ? n / 2 : (n + 1) / 2` semi-colon

07:41 یہ سٹیٹمنٹ پڑھتا ہے،

07:43 اگر n اون ہے، nHalf n بائی 2، یا، یہ n پلس 1 بائی 2 ہے۔

07:50 اب اسے استعمال کر کے دیکھتے ہیں۔

07:52 فائل سیو اور رن کریں، Ctrl S اور Ctrl F11 کیز دبائیں۔

07:59 جیسا کہ ہم دیکھ سکتے ہیں، آؤٹ پٹ توقع کے مطابق ہے۔

08:02 اس طرح، ternary آپریٹر کوڈ میں بے ترتیبی کو کم کرتا ہے اور پڑھنے کی اہلیت کو بہتر کرتا ہے۔

08:09 اب ہم اس ٹیوٹورل کے اختتام پر پہنچ گئے ہیں۔

08:11 اس ٹیوٹورل میں ہم نے سیکھا:

08:13 Nested-If سٹیٹمنٹ اور Ternary آپریٹر کے بارے میں

- 08:15 ایک جاوا پروگرام میں Nested-If سٹیٹمنٹ اور Ternary آپریٹر کے استعمال کے بارے میں
- 08:22 اب اس پر مشق لیں.
- 08:23 Ternary اور Nested-If آپریٹر . مندرجہ ذیل پر جاوا پروگرام لکھیں.
- 08:28 nested-if کا استعمال کرتے ہوئے جانچ کریں کہ ایک تعداد اون اور 11 کا ملٹیپل ہے یا نہیں.
- 08:34 Ternary آپریٹر کا استعمال کرتے ہوئے دی گئی دو اعداد کے درمیان بڑی تعداد کی نشاندہی کریں
- 08:40 سپوکن ٹیوٹوریل پروجیکٹ کے بارے میں مزید جاننے کے لئے، مندرجہ ذیل لنک پر دستیاب ویڈیو دیکھیں.
- 08:45 یہ سپوکن ٹیوٹوریل پروجیکٹ کو خلاصہ کرتا ہے . اگر آپ کے پاس اچھی بینڈ وڈتھ نہیں ہے، تو آپ اسے ڈاؤن لوڈ کر کے دیکھ سکتے ہیں.
- 08:52 سپوکن ٹیوٹوریل پروجیکٹ ٹیم،
- 08:54 سپوکن ٹیوٹوریل کا استعمال کرتے ہوئے ورکشاپ چلاتی ہے اور
- 08:57 اور آن لائن ٹیسٹ پاس کرنے والوں کو سہارا دیتے ہیں . مزید معلومات کے لئے، براہ مہربانی contact AT spoken HYPHEN tutorial DOT org پر لکھیں.
- 09:07 سپوکن ٹیوٹوریل پروجیکٹ، ٹاک-ٹو-اے-ٹیچر پروجیکٹ کا حصہ ہے اور
- 09:11 اسے قومی خواندگی مشن نے ICT، ایمپارڈی حکومت ہند کے ذریعے حمایت کی ہے.
- 09:17 اس مشن پر مزید معلومات spoken HYPHEN tutorial DOT org SLASH NMEICT
- HYPHEN Intro پر دستیاب ہے
- 09:26 اس اسکرپٹ کا ترجمہ اور صدا بندی میں نے یعنی وجاہت احمد نے کی ہے، ہمارے ساتھ شریک ہونے کیلئے آپ کا شکریہ