

Narration	Time
java میں this کی وارڈ کے استعمال پر سپوکن ٹیوٹوریل میں آپ کا استقبال ہے۔	00:02
اس ٹیوٹوریل میں ہم سیکھیں گے	00:07
this کی وارڈ کا استعمال	00:09
fields کے ساتھ this کی وارڈ کا استعمال	00:11
اور کنسٹرکٹس کے تسلسل کے لئے this کی وارڈ استعمال کرنے کے بارے میں،	00:14
یہاں ہم استعمال کر رہے ہیں، ابنٹوورژن 11.10 jdk 1.6 اکلپس 3.7.0	00:17
اس ٹیوٹوریل کو سمجھنے کے لئے آپ کو معلوم ہونا چاہئے۔	00:28
اکلپس کا استعمال کرتے ہوئے جاوا میں کنسٹرکٹ کیسے بنائیں۔	00:30
اگر نہیں تو، متعلقہ ٹیوٹوریل کے لئے دکھائی گئی ہماری ویب سائٹ پر جائیں۔ (http://www.spoken-tutorial.org)	00:34
اب ہم this کی وارڈ کے بارے میں سیکھیں گے۔	00:40
کنسٹرکٹ میں، موجودہ آبجیکٹ کے لئے this ایک ریفرنس یعنی حوالہ ہے۔	00:44
ہم this کا استعمال کرتے ہوئے کنسٹرکٹ میں موجودہ آبجیکٹ کے کسی بھی رکن (ممبر) کو ریفر کر سکتے ہیں۔	00:48
اب ہم فیلڈ کے ساتھ this کی وارڈ کا استعمال دیکھیں گے۔	00:55
this کی وارڈ سے ہمیں نام کے ٹکراؤ کو دور کرنے میں مدد ملتی ہے۔	01:00
ہم یہاں اس طرح کی ایک مثال دیکھیں گے۔	01:07
اس کے لئے اکلپس کھولیں۔	01:10
Student کلاس کھولیں، جو ہم نے گزشتہ ٹیوٹوریل میں بنایا ہے۔	01:17
ڈفالٹ کنسٹرکٹ کو کمینٹ کریں، 1 پیرامیٹر کے ساتھ کنسٹرکٹ کمینٹ کریں۔	01:23
اور پہلے دو آبجیکٹس دکھانے کا کوڈ کو بھی کمینٹ کریں۔	01:40

اب، parameterized کنسٹرکٹر پر توجہ دیں۔	02:03
کنسٹرکٹر کے پاس کئے گئے آرگیومنٹس ہیں۔ the_name اور the_roll_number	02:11
name اور roll_number انسٹنس ویریبل ہے۔	02:20
اب، میں name اور roll_number آرگیومنٹ بدلتا ہوں۔	02:26
تو کنسٹرکٹر کے اندر ہمارے پاس	02:39
name equal to name اور roll_number equal to roll_number ہے	02:42
اب فائل سیوا اور رن کریں۔ لہذا Ctrl+S اور Ctrl+F11 دبائیں۔	02:55
ہمیں درج ذیل قسم کا آؤٹ پٹ ملتا ہے۔	03:04
I am a Parameterized Constructor 0 null	03:07
اب کوڈ پرواپس آئیں۔	03:12
ہم کوڈ میں 2 وارننگس یعنی انتباہ دیکھتے ہیں۔	03:17
اپنے ماؤس کو وارننگ سمبل پر منتقل کریں۔	03:20
ہم دیکھ سکتے ہیں۔ The assignment to the variable roll_number has no effect.	03:23
اور The assignment to the variable name has no effect.	03:29
ایسا اس لئے کیونکہ کنسٹرکٹر میں رول نمبر اور نام لوکل ویریبل ہیں۔	03:33
لوکل ویریبل وہ ویریبل ہیں، جو میٹھڈ یا بلاک میں رسائی کے قابل ہوتے ہیں۔	03:40
یہاں name اور roll_number، 11 اور Raju کے طور پر انشلائز ہوں گے۔	03:47
کیونکہ ہم نے کنسٹرکٹر کے طور پر ویلیوس 11 اور Raju پاس کئے ہیں۔	03:54
لیکن ایک بار وہ کنسٹرکٹر کے باہر آجاتے ہیں تو رسائی کے قابل نہیں ہوتے	04:01
ہم جانتے ہیں، کہ ہماری جان پہچان والے name اور roll_number انسٹنس ویریبلس ہیں۔	04:06
وہ پہلے ہی 0 اور null سے انشلائز ہو جاتے ہیں، جیسے ہی آ بھیکٹ بن جاتا ہے۔	04:13
لہذا ہمیں آؤٹ پٹ 0 اور null ملتا ہے۔	04:18

اب کنسٹرکٹر کے اندر ایک چھوٹی تبدیلی کریں۔	04:21
تو ٹائپ کریں this dot roll_number equal to roll_number.	04:29
اور this dot name equal to name.	04:37
اب فائل سیو اور رن کریں۔ لہذا Ctrl+S اور Ctrl+F11 کی زد بائیں۔	04:44
ہمیں اس طرح آؤٹ پٹ ملتا ہے۔	04:51
I am a Parameterized Constructor 11 اور Raju	04:53
ایسا اس لئے کیونکہ this dot roll_number اور this dot name اور instance اور ریبلنس roll_number اور name کو ریفر کرتے ہیں۔	04:58
اور یہاں roll_number اور name میتھڈ میں پاس کئے گئے آرگیومنٹس ہیں۔	05:12
لوکل اور instance اور ریبلنس کے بیچ تضاد دور کرنے کے لئے ہمارے پاس this کی وارڈ ہے۔	05:19
اب ہم constructor کی چیپنگ کے لئے this کی وارڈ کا استعمال دیکھیں گے۔	05:29
ہم this کی وارڈ کا استعمال constructor کے اندر سے کسی دوسرے کو کال کرنے کے لئے کر سکتے ہیں۔	05:34
constructors یکساں کلاس میں ہونے چاہئے۔	05:39
یہ explicit constructor invocation کہلاتا ہے۔	05:43
اب Student کلاس پر واپس جائیں، جسے ہم نے بنایا ہے۔	05:46
اب کمیٹس کو خارج کر دیں۔	05:53
اب پہلے دو کنسٹرکٹرز میں instance اور ریبلنس کی ویلیو کا تعین کرنے کے حصہ کو کمیٹ کر دیں۔	06:28
پھر اس حصہ کو کمیٹ کریں جو دوسرا اور تیسرا آجیکٹ بناتا ہے۔	06:52
اب پہلے کنسٹرکٹر پر پیرامیٹرز کے بغیر آئیں۔	07:08
کرلی بریکٹ کے بعد ٹائپ کریں، this، بریکٹس میں 11 اور سیمی کولن	07:16
دوسرے کنسٹرکٹر کے اندر ٹائپ کریں، this، بریکٹس میں 11 کو ماڈل کوٹس میں Raju سیمی کولن	07:28
اب فائل سیو اور رن کریں۔ لہذا Ctrl+S اور Ctrl+F11 کی زد بائیں۔	07:42

ہمیں اس طرح کا آؤٹ پٹ ملتا ہے،	07:49
I am a Parameterized Constructor	07:51
I am a constructor with a single parameter	07:54
I am a Default Constructor	07:57
11 اور Raju	
اب، میں آؤٹ پٹ سمجھتا ہوں۔	08:02
جب آجیکٹ بنتا ہے، متعلقہ کنسٹرکٹر کال ہوتا ہے۔	08:08
یہاں کنسٹرکٹر no argument constructor ہے۔	08:13
کنسٹرکٹر میں کنٹرول پہلی لائن پر آ جاتا ہے۔	08:20
اس کا سامنا ہوتا ہے، this، بریکٹس میں 11 اسٹیٹمنٹ کے ساتھ۔	08:24
اس لئے یہ ایسا کنسٹرکٹر کال کرتا ہے، جو سنگل انجڑ آرگومینٹ لیتا ہے۔	08:26
پھر کنٹرول، this بریکٹس میں 11 کوما Raju پر آ جاتا ہے۔	08:36
لہذا، یہ ایسا کنسٹرکٹر کال کرتا ہے، جو 1 انجڑ اور 1 سٹرنگ آرگومینٹ لیتا ہے۔	08:44
I am a Parameterized Constructor.	08:53
لہذا یہ کنسٹرکٹر ایکریوٹ ہوتا ہے اور ہمیں یہ آؤٹ پٹ ملتا ہے	
اب instance ویریبلس، 11 اور Raju کے لئے انشیلائز ہوں گے۔ جیسے ہم نے پاس کئے۔	09:02
اب کنٹرول، کالنگ کنسٹرکٹر پرواپس جاتا ہے۔	09:11
تو اب دوسرا کنسٹرکٹر ایکریوٹ ہوتا ہے۔	09:16
I am a constructor with a single parameter.	09:19
پھر، کنٹرول پہلے کنسٹرکٹر پر جاتا ہے اور اسے ایکریوٹ کرتا ہے۔	09:25
I am a default constructor.	09:30
پھر studentDetail میٹھڈ ایکریوٹ ہوتا ہے۔	09:36
لہذا ہمیں 11 اور Raju ملتا ہے۔	09:42

اب، ایک چھوٹی سا تبدیلی کریں۔	09:45
this اسٹیٹمنٹ کو کنسٹرکٹر کا آخری اسٹیٹمنٹ بنائیں۔	09:47
ہمیں کمپائلر ایرر ملتا ہے۔	10:00
ایرر سمبل پر ماؤس رکھیں۔	10:03
ہمیں ایرر ملتا ہے:	10:06
Constructor call must be the first statement in the constructor.	10:07
لہذا ہمیں اسے Constructor میں پہلی لائن میں رکھنا چاہئے۔	10:12
تو اسے Constructor کی پہلی لائن بنائیں۔	10:16
اب ہم دیکھ سکتے ہیں کہ ایرر ہٹ گیا ہے۔	10:27
تو اس ٹیوٹوریل میں ہم نے سیکھا۔	10:31
فیلڈس کے ساتھ this کی وارڈ کا استعمال کرنا،	10:35
کنسٹرکٹس کی چیکنگ کے لئے this کی وارڈ کا استعمال کرنا۔	10:38
کنسٹرکٹر میں this کی وارڈ کا استعمال کیسا ہونا چاہئے۔	10:41
خود تشخیص کے لئے، پہلے بنائے ہوئے Employee کلاس میں،	10:45
دو پیرامیٹرز کے ساتھ ایک کنسٹرکٹر بنائیں۔	10:49
instance ریبلنس کو انشیلانز کرنے کے لئے this کی وارڈ کا استعمال کریں۔	10:52
1 اور بغیر پیرامیٹرز کے ساتھ ایک کنسٹرکٹر بنائیں۔	10:57
ٹیوٹوریل میں دی گئی تفصیلات کے مطابق this کا استعمال کرتے ہوئے کنسٹرکٹس کی چیکنگ کرنے کی مشق کریں۔	11:01
سپوکن ٹیوٹوریل پروجیکٹ کے بارے میں مزید جاننے کے لئے،	11:07
مندرجہ ذیل لنکس پر دستیاب ویڈیو دیکھیں۔	11:09
http://spoken-tutorial.org/What_is_a_Spoken_Tutorial	
یہ سپوکن ٹیوٹوریل پروجیکٹ کو خلاصہ کرتا ہے۔	11:12
اگر آپ کے پاس اچھی بینڈ ویڈیو نہیں ہے، تو آپ اسے ڈاؤن لوڈ کر کے دیکھ سکتے ہیں۔	11:16
سپوکن ٹیوٹوریل پروجیکٹ ٹیم	11:19

11:23	سپوکن ٹیوٹوریل کا استعمال کرتے ہوئے ورکشاپ بھی چلاتی ہے۔
11:26	اور آن لائن ٹیسٹ پاس کرنے والوں کو سہولت بھی دیتے ہیں۔
11:30	مزید معلومات کے لئے contact@spoken-tutorial.org پر لکھیں۔
11:36	سپوکن ٹیوٹوریل پروجیکٹ ٹاک-ٹو-اے ٹیچر پروجیکٹ کا حصہ ہے۔
11:40	اسے قومی خواندگی مشن نے ICT، ایمپچارڈی حکومت ہند کے ذریعے حمایت کی ہے۔
11:46	اس مشن پر مزید معلومات درج ذیل لنک پر دستیاب ہیں۔
11:55	اب ہم اس ٹیوٹوریل کے آخر میں آ گئے ہیں۔
11:58	اس اسکرپٹ کا ترجمہ اور صدابندی میں نے یعنی وجاہت احمد نے کی ہے، ہمارے ساتھ شریک ہونے کیلئے آپ کا شکریہ