

Time	Narration
00:02	به Spoken Tutorial در Nested-If and Ternary Operator در java خوش آمدید.
00:07	در پایان این برنامه شما قادر خواهید بود که Nested-If Statement و Ternary operators و استفاده از آن‌ها در برنامه Java را توضیح دهید.
00:17	برای این برنامه ما از Ubuntu v 11.10, JDK 1.6 و Eclipse IDE 3.7.0 استفاده می‌کنیم.
00:27	برای این برنامه شما باید در مورد استفاده از relational و logical operators و if...else control
00:29	flow statements بدانید.
00:33	
00:36	اگر نه برای برنامه‌های مربوطه لطفاً به وب سایت نشان داده شده مراجعه کنید.
00:41	دستور Nested if: یک If statement در یک if statement دیگر را دستور nested-if می‌گویند.
00:49	حالا ساختار (syntax) برای نوشتن Nested-If statement را می‌بینیم.
00:53	در این صورت اگر condition1 درست باشد سپس برنامه condition2 را بررسی می‌کند.
00:59	Condition2 با استفاده از یک if statement دیگر داده شده است.
01:03	اگر condition2 درست باشد پس برنامه 1 Statement or block را اجرا می‌کند.
01:09	در غیر این صورت 2 Statement or block را اجرا می‌کند.
01:13	اگر condition1 اشتباه باشد برنامه condition2 را بررسی نمی‌کند.
01:18	و بجای آن مستقیم به else statement یعنی block 3 می‌رود.
01:24	با یک مثال این را بهتر می‌فهمیم.
01:28	ما eclipse IDE و اسکلت مورد نیاز برای بقیه کد را داریم.
01:32	ما یک class NestedIfDemo را ایجاد کرده‌ایم و main method را به آن اضافه کرده ایم.
01:37	ما بررسی می‌کنیم که عدد داده شده زوج است یا فرد
01:42	ما همچنین اعداد منفی را با استفاده از nested-if بررسی می‌کنیم.
01:46	پس در main method این را تایپ کنید.
01:49	int n = minus 5;
01:54	ما متغیر n را برای ذخیره عدد منفی ایجاد کرده ایم.

01:58	حالا if condition را می نویسیم.
02:01	خط بعد if (n < 0) را تایپ کنید.
02:07	آکولاد باز. Enter را فشار دهید.
02:10	System.out.println بین کروشه و double quotes Negative number;
02:22	ابتدا می بینیم که آیا عدد منفی است
02:25	اگر بله پس برای زوج و فرد بودن بررسی می کنیم.
02:29	اگر عدد منفی نیست ما برای زوج (even) و فرد (odd) بررسی می کنیم.
02:34	خط بعد else آکولاد باز را تایپ کنید. Enter را فشار دهید.
02:42	حالا اگر اجرا به قسمت else آمده است
02:45	یعنی عدد منفی نیست
02:48	پس در این قسمت else ما فرد یا زوج بودن را بررسی می کنیم.
02:52	حالا if بین کروشه <code>n modulus 2 double equal to 0</code> آکولاد باز را تایپ کنید. Enter را فشار دهید.
03:03	System.out.println بین کروشه و double quotes Even number
03:13	سپس قسمت else را تایپ کنید . else آکولاد باز Enter را فشار دهید.
03:18	System.out.println بین کروشه و double quotes Odd number semicolon را تایپ کنید.
03:29	پس مطمئن می شویم که اعداد منفی برای زوج و فرد بودن بررسی نمی شوند.
03:34	حالا کد در حال اجرا را می بینیم.
03:37	فایل را Save و run کنید. همانطور که می بینیم خروجی <code>negative number</code> را بدست می آوریم.
03:43	حالا عدد مثبت را بررسی می کنیم.
03:46	و <code>n = -5</code> را به <code>n = 5</code> تغییر دهید.
03:53	فایل را Save و Run کنید.
03:57	همانطور که می بینیم خروجی <code>Odd number</code> است. حالا یک عدد زوج را می بینیم.
04:04	و <code>n = 5</code> را به <code>n = 10</code> تغییر دهید.
04:09	فایل را Save و Run کنید.
04:12	همانطور که می بینیم خروجی <code>Even number</code> می باشد.

04:17	این مراحل شامل کردن if statement داخل دیگری nested-if نامیده می شود.
04:22	هیچ محدودیتی در مورد تعداد nesting نمی باشد.
04:25	ولی بهتر است که برای بیشتر از سه مرحله nesting نرویم.
04:31	حالا ternary operator را می بینیم.
04:33	main method را واضح می کنم.
04:37	یک برنامه که عدد را بر 2 تقسیم می کند می نویسیم.
04:40	این یک برنامه جزئی می باشد . اما مشکل زمان تقسیم اعداد فرد پیش می آید.
04:45	وقتی 7 تقسیم بر 2 می شود ما 3 را بدست می آوریم.
04:48	اما اگر ما نتیجه را round off نخواهیم, چه می شود.
04:50	یعنی وقتی 7 تقسیم بر 2 می شود ما 4 را بدست می آوریم نه 3
04:56	به زبان ساده ما به عدد بعدی احتیاج داریم.
04:59	حالا می بینیم چگونه چنین برنامه ای را بنویسیم . پس در main method شما <code>int n, nHalf</code> semicolon را تایپ کنید.
05:08	ما عدد را در n ذخیره می کنیم و نصف عدد را در nHalf
05:13	خط بعد <code>n = 5</code> semicolon را تایپ کنید.
05:18	خط بعد if بین گروه <code>n modulus 2 == 0</code> آکولاد باز را تایپ کنید. Enter را فشار دهید.
05:28	و <code>nHalf = n / 2</code> semicolon را تایپ کنید.
05:34	در قسمت else شما بین آکولاد <code>nHalf equal to within brackets n + 1 the whole devided by 2</code> semicolon را تایپ کنید.
05:50	ما بررسی می کنیم که عدد زوج و یا فرد است و تقسیم را بر این اساس انجام می دهیم.
05:55	حالا print statement را برای دیدن برنامه در حال اجرا اضافه می کنیم.
05:59	پس <code>System.out.println</code> بین گروه <code>nHalf semicolon</code> را تایپ کنید.
06:11	فایل را Save و Run کنید.
06:14	همانطور که می بینیم هدف ما برآورده شده است . ما خروجی 3 را بدست می آوریم نه 2
06:21	اما اگر توجه کنیم ما مقدار متغیر را طبق condition (وضعیت) تنظیم کرده ایم.

06:27	در برنامه ما syntax بیشتر از logic می باشد.
06:31	این وقتی است که ternary operator کد را ساده تر می کند.
06:35	Ternary operator یک conditional operator است که نتیجه مثل nested-if را می دهد.
06:40	این syntax کوتاه را ایجاد می کند و با علامت سؤال نشان داده می شود.
06:45	این سه operands را همزمان می گیرد.
06:48	حالا در مورد syntax of Ternary Operator یاد می گیریم.
06:53	Expression وضعیتی است که باید بررسی شود.
06:56	Operand 1 مقدار متغیر می باشد. Result اگر وضعیت درست باشد
07:03	Operand 2 مقداری است که اگر وضعیت اشتباه باشد.
07:09	حالا این را در برنامه خود استفاده می کنیم.
07:12	ابتدا if-else statement را حذف می کنیم.
07:17	و $n + 1$ بین کروهه n Half equal to n modulus 2 double equal to 0 question mark $n / 2$ colon the whole divided by 2 semicolon را تایپ کنید.
07:41	این statement این را می خواند:
07:43	اگر n زوج باشد n Half is n by 2 در غیر این صورت این n plus 1 by 2 می باشد.
07:50	حالا این را در حال اجرا می بینیم.
07:52	فایل را Save و Run کنید. Ctrl S و Ctrl F11 را فشار دهید.
07:59	خروجی همان است که انتظار داشتیم.
08:02	با این روش ternary operator در هم ریختگی را در کد کم می کند و خواندن آن را بهتر می کند.
08:09	به پایان برنامه رسیده ایم.
08:11	در این برنامه این موارد را یاد گرفتیم:
08:13	در مورد Ternary Operator و Nested-If Statement و
08:15	استفاده از Ternary Operator و Nested-If Statement در برنامه Java
08:22	حالا یک ارائه در مورد Ternary Operator و Nested-If Statement. یک برنامه Java برای اینها بنویسید.
08:28	با استفاده از nested-if بررسی کنید که آیا عدد زوج است و همچنین مضرب 11 است.

08:34	با استفاده از Ternary operator عدد بیشتر بین دو عدد داده شده را مشخص کنید.
08:40	برای دانستن بیشتر Spoken Tutorial Project ویدئو را در لینک زیر ببینید.
08:45	این خلاصه پروژه می باشد. اگر پهنای باند خوبی ندارید ابتدا دانلود و سپس مشاهده کنید.
08:52	تیم این پروژه کارگاه آموزشی استفاده از Spoken Tutorial ارائه می دهد. و به کسانی که آزمون آنلاین را قبول
08:54	شوند، گواهینامه می دهد. برای جزئیات بیشتر لطفاً contact AT spoken HYPHEN tutorial DOT org ایمیل
08:57	بفرستید
09:07	Spoken Tutorial Project بخشی از پروژه Talk To a Teacher می باشد. که تحت پشتیبانی National
09:11	Mission on Education از طریق ICT توسط MHRD دولت هند می باشد.
09:17	اطلاعات بیشتر در لینک spoken hyphen tutorial dot org Slash NMEICT hyphen Intro در دسترس می باشد:
09:26	ترجمه و صداگذاری شبنم اقبال خان. با تشکر از شما