

Time	Narration
00:01	به spoken tutorial در do-while Loop خوش آمدید.
00:06	در این برنامه شما do-while loop و چگونگی استفاده از آن را یاد می گیرید.
00:12	برای این برنامه ما از Ubuntu 11.10, JDK 1.6 و Eclipse 3.7 استفاده می کنیم.
00:20	برای این برنامه شما باید با while loop در java آشنایی داشته باشید.
00:25	اگر نه برای برنامه های مربوطه به وب سایت نشان داده شده مراجعه کنید.
00:32	اینجا structure برای do-while loop می باشد.
00:37	توجه کنید که این شبیه while loop می باشد.
00:40	این دو قسمت دارد:
00:42	اولی loop running condition می باشد و دومی loop variable است.
00:51	تنها تفاوت این است که وضعیت بعد از do block نوشته شده است.
00:58	پس وضعیت بعد از اجرای statement داخل do block اجرا می شود.
01:05	حالا یک مثال
01:07	به eclipse بروید.
01:11	در اینجا Eclipse IDE و اسکلت مورد نیاز برای بقیه کد را داریم.
01:17	ما class DoWhileDemo را ایجاد کرده ایم و main metod را به آن اضافه کرده ایم.
01:22	ما شماره از 1 تا 10 را با استفاده از حلقه do-while پرینت می کنیم.
01:27	و n مساوی 1 را تایپ کنید
01:29	
01:32	و n که loop variable می باشد.
01:36	سپس این را تایپ کنید: <b>do</b>
01:40	Braces باز و بسته
01:44	داخل braces که <code>System.out.println(n);</code>
01:55	ما مقدار n را پرینت می کنیم و سپس آن را افزایش می دهیم. و n مساوی n به اضافه 1;
02:05	و این را تا زمانی
02:08	که n کوچکتر و یا مساوی 10 شود انجام می دهیم.

02:10	پس خارج از braces شما while در پراتز n کمتر و یا مساوی 10 را تایپ کنید.
02:20	و do-while را با استفاده از semicolon ببینید.
02:25	حالا کد در حال اجرا را می بینیم.
02:28	Save کنید و Run
02:37	ما می بینیم که اعداد از 1 تا 10 پرینت شده اند.
02:42	حالا می فهمیم که چگونه کد اجرا می شود.
02:47	ابتدا مقدار 1 پرینت می شود و سپس n که 2 می شود.
02:52	و سپس وضعیت بررسی می شود.
02:55	چون این درست است دوباره 2 پرینت می شود و n نیز 3 می شود.
03:00	و تا زمانی که همه 10 شماره پرینت شوند و مقدار n که 11 شود ادامه پیدا می کند.
03:06	وقتی که $n=11$ وضعیت رد می شود و حلقه متوقف می شود.
03:11	حال اعداد از 50 تا 40 را به ترتیب نزولی پرینت می کنیم.
03:17	پس با 50 شروع می کنیم.
03:19	و $n=1$ را به $n=50$ تغییر دهید.
03:23	چون ما از عدد بزرگتر به عدد کوچکتر می رویم و loop variable را کاهش می دهیم.
03:29	پس $n=n+1$ را به $n=n-1$ تغییر دهید.
03:34	تا زمانی که n بیشتر و یا مساوی 40 است ما loop می کنیم.
03:40	پس وضعیت را به $n \geq 40$ تغییر دهید.
03:48	حالا خروجی را می بینیم.
03:50	Save کنید و Run
03:57	همانطور که می بینیم اعداد از 50 تا 40 پرینت می شوند.
04:02	حالا یک logic متفاوت برای do-while loop را امتحان می کنیم.
04:10	باید ببینیم که عدد داده شده مربع کامل است یا نه
04:15	ابتدا main method را واضح کنید.
04:19	سپس; $int n=25$ را تایپ کنید.
04:25	ما باید ببینیم که مقدار در n مربع کامل است یا نه
04:32	سپس; $int x=0$ را تایپ کنید.

04:37	ما از $x$ برای ذخیره کردن توان دوم عدد اگر مربع کامل باشد استفاده می کنیم.
04:44	سپس این را تایپ کنید: <code>do</code>
04:46	Braces باز و بسته
04:49	داخل braces $x$ مساوی $x$ به اضافه 1;
04:55	و خارج از braces
04:58	While در پراتز <code>x into x is less than n</code>
05:06	و با استفاده از semicolon شما <code>do-while</code> را ببندید.
05:10	تا زمانی که $x$ در $x$ کمتر از $n$ باشد ما مقدار $x$ را افزایش می دهیم.
05:16	پس وقتی که loop متوقف شود برعکس این وضعیت درست خواهد بود.
05:22	یعنی <code>x into x</code> باید مساوی $n$ باشد.
05:26	یا باید بیشتر از $n$ باشد.
05:28	اگر $x$ در $x$ مساوی $n$ باشد پس عدد مربع کامل می باشد.
05:32	اگر مساوی $n$ نباشد عدد مربع کامل نمی باشد.
05:37	در آخر ما وضعیت را پرینت می کنیم.
05:47	<code>System.out.println (x*x==n);</code>
05:55	حالا کد در حال اجرا را می بینیم.
05:59	Save کنید و Run همانطور که می بینیم خروجی <code>true</code> است.
06:07	حالا با یک مربع کامل دیگر امتحان می کنیم
06:10	و $n=25$ را به $n=49$ تغییر دهید.
06:15	Save کنید و Run
06:20	می بینیم که دوباره <code>true</code> را بدست می آوریم.
06:23	حالا با یک عدد که مربع کامل نیست امتحان می کنیم.
06:26	49 را به 23 تغییر دهید. Save کنید و Run
06:34	و ما <code>false</code> را همانطور که انتظار داشتیم بدست می آوریم.
06:37	حالا ببینیم اگر مقدار $n$ ما 0 باشد چه خواهد شد.
06:42	و $n=23$ را به $n=0$ تغییر دهید چون 0 یک عدد طبیعی نیست ما باید <code>false</code> را بدست آوریم.

06:52	حالا کد را run می کنیم.
06:54	Save کنید و Run
07:00	همانطور که انتظار داشتیم false را بدست می آوریم.
07:05	این اتفاق می افتد چون حتی قبل از اینکه وضعیت
07:08	X into x is less than n بررسی شود مقدار x افزایش می یابد و 1 می شود.
07:16	Loop condition رد می شود و حلقه run نمی شود.
07:20	به این روش با استفاده از do-while loop ما مطمئن می شویم که 0 بعنوان مربع کامل در نظر گرفته نمی شود.
07:26	به این طریق do-while loop برای حل کردن محدوده ای از مشکلات استفاده می شود.
07:31	مخصوصاً وقتی که loop باید حداقل یکبار run شود.
07:37	به پایان برنامه می رسم.
07:40	در این برنامه ما در مورد do-while loop و چگونگی استفاده از آن یاد گرفتیم.
07:42	
07:46	ارائه: این تمرینها را حل کنید.
07:50	یک عدد <b>binary</b> داده شده است معادل اعشاری آن را پیدا کنید. مثال: $11010 = > 26$
07:56	برای دانستن بیشتر در مورد Spoken Tutorial Project ویدئو در لینک زیر مشاهده کنید.
08:01	این خلاصه پروژه می باشد. اگر پهنای باند خوبی ندارید ابتدا دانلود و سپس مشاهده کنید.
08:06	تیم پروژه Spoken Tutorial
08:10	کارگاه آموزشی استفاده از Spoken Tutorial ارائه می دهد. و به کسانی که آزمون آنلاین را قبول شوند، گواهینامه می دهد.
08:16	برای جزئیات بیشتر به <a href="http://contact.at.spokenhyphen.tutorial.dot.org">contact AT spoken Hyphen tutorial DOT org</a> ایمیل کنید.
08:22	Spoken Tutorial Project بخشی از پروژه Talk To a Teacher می باشد. که تحت پشتیبانی National Mission on Education از طریق ICT توسط MHRD دولت هند می باشد.
08:32	اطلاعات بیشتر در لینک <a href="http://spoken-tutorial.org/NMEICT-Intro">http://spoken-tutorial.org/NMEICT-Intro</a> موجود می باشد.
08:36	ترجمه و صداگذاری شبیم اقبال خان. با تشکر از شما