

Time	Narration
00:01	به spoken tutorial در Method overloading در Java خوش آمدید.
00:06	در این برنامه یاد می‌گیریم که:
00:08	Method overloading چیست.
00:10	و method را overload کنیم.
00:13	در اینجا ما از Ubuntu version 11.10 OS java Development kit 1.6 و Eclipse 3.7.0 استفاده می‌کنیم.
00:24	برای این برنامه شما باید بدانید که
00:26	چگونه methods را ایجاد کنید و
00:29	Overload constructor در Java با استفاده از eclipse
00:32	اگر نه برای برنامه‌های مربوطه به وب سایت نشان داده شده مراجعه کنید. http://www.spoken-tutorial.org
00:39	Method overloading چیست؟
00:42	دو یا تعداد بیشتری methods را با یک نام در class تعریف کنید.
00:46	آن‌ها باید در تعداد و نوع پارامترها با هم متفاوت باشند.
00:50	به این متدها overloaded methods گفته می‌شود.
00:54	و به این مراحل method overloading گفته می‌شود.
00:57	حالا می‌بینیم که چگونه method را overload کنیم.
01:00	در eclipse من class Addition را دارم.
01:06	در این class ما دو متغیر عدد صحیح را اعلام می‌کنیم.
01:10	پس int a مساوی 10 و int b مساوی 5 را تایپ کنید.
01:19	حالا یک method برای جمع کردن این دو عدد صحیح ایجاد می‌کنیم.
01:23	پس void add پرانتزها را تایپ کنید.
01:30	بین آکولاد System dot out dot println(); را تایپ کنید.
01:40	بین پرانتز a+b
01:44	پس این متد به ما جمع دو متغیر عدد صحیح (int) را می‌دهد.
01:50	حالا یک method دیگر ایجاد می‌کنیم که دو پارامتر را می‌گیرد.
01:55	پس void addTwoNumbers() را تایپ کنید.

02:04	و بین پرانتز <code>int num1 comma int num2</code>
02:14	سپس بین آکولاد <code>System dot out dot println num1 plus num2</code> را تایپ کنید.
02:35	پس این متد به ما جمع دو مقدار که بعنوان <code>arguments</code> در این <code>method</code> پاس شده‌اند را می دهد.
02:44	حالا یک <code>object</code> از <code>class</code> را ایجاد می‌کنیم و <code>methods</code> را می خوانیم.
02:49	پس در <code>main method</code> شما <code>Addition</code> که نام <code>class</code> و <code>obj</code> مسادی <code>new Addition</code> پرانتزها <code>semicolon</code> را تایپ کنید.
03:13	سپس <code>Obj.add();</code>
03:18	و <code>Obj.addTwoNumbers();</code> و بین پرانتز
03:31	ما دو <code>arguments</code> را پاس می کنیم.
03:33	فرض کنید ما مقدار <code>floating point</code> (اعشاری) را پاس می کنیم.
03:37	پس 2.5 کاما و یک عدد صحیح 3 را تایپ کنید.
03:45	ما یک اشتباه می بینیم. The method addTwoNumbers int comma int of the class addition is not applicable for the arguments double comma int
03:57	پس ما در متد بجای <code>int</code> از <code>double</code> استفاده می کنیم.
04:06	پس <code>double</code> را جایگزین <code>int</code> کنید. فایل را <code>save</code> کنید.
04:12	می‌بینیم که اشتباه برطرف می شود.
04:17	ما همچنین می‌دانیم که <code>java</code> بطور خودکار یعنی <code>implicitly</code> که <code>int</code> را به <code>double</code> تبدیل می کند.
04:24	پس در اینجا می‌توانیم همچنین <code>integer arguments</code> را پاس کنیم.
04:28	برنامه را <code>Save</code> و <code>Run</code> کنید.
04:32	در خروجی ما جمع دو مقدار عدد صحیح را می بینیم.
04:37	همچنین جمع دو <code>numeric arguments</code> که ما پاس کرده ایم.
04:43	حالا می‌بینیم هر دو متد یک عمل را انجام می دهند.
04:50	تفاوت این است که متد اول پارامتر ندارد ولی متد دوم پارامتر دارد.
05:00	پس در این موارد <code>Java</code> به ما <code>method overloading</code> را می دهد.
05:05	پس این یک نام به هر دو متد می دهد.
05:09	پس <code>add()</code> را جایگزین <code>addTwoNumbers()</code> کنید. همچنین اینجا را تغییر دهید.
05:29	ما یک متد دیگر با همین عملیات را تعریف می کنیم.
05:33	پس <code>void add()</code> را تعریف کنید.

05:38	بین پرانتز int n1 comma int n2 comma int n3
05:51	پس ما سه پارامتر را داده ایم.
05:54	System dot out dot println()
06:03	بین پرانتز n1 به اضافه n2 به اضافه n3
06:11	پس این متد جمع سه عدد را می دهد.
06:17	حالا این method را می خوانیم.
06:19	پس 4 comma 5 comma 1 add() obj را تایپ کنید.
06:35	Run و Save
06:39	در خروجی می بینیم که جمع سه عدد 10 است.
06:47	پس کامپایلر Java متد مناسب را طبق پارامتر ها overloads می کند.
06:52	این اعداد و نوع پارامترهایی که پاس می شوند را بررسی می کند.
06:57	پس بعنوان برنامه نویسی ما نباید در مورد نام method نگران باشیم.
07:01	و همچنین در مورد نوع و تعداد arguments که پاس می کنیم.
07:05	ما همچنین یک method دیگر که strings را اضافه می کند را ایجاد می کنیم.
07:11	پس ما یک overload method دیگر را ایجاد می کنیم.
07:15	و void add String s1 comma sting s2 را تایپ کنید.
07:29	System dot out dot println()
07:41	بین پرانتز s1 به اضافه s2
07:45	سپس ما این method را call می کنیم.
07:50	پس obj dot add را تایپ کنید.
07:55	World double quotes در Hello comma شما double quotes در double quotes فاصله
08:07	حالا برنامه را run و save کنید.
08:12	پس در خروجی Hello فاصله World را می بینیم.
08:16	پس add method با دو string arguments و string (رشته) را اضافه می کند
08:21	فرض کنید ما add method با return type را اعلام می کنیم.
08:27	پس int add پرانتزها بدون پارامتر و آکولاد را تایپ کنید.
08:40	ما یک اشتباه می بینیم: Duplicate method add() in type Addition
08:48	چون ما از قبل method add() را بدون پارامتر اعلام کرده ایم.

08:54	پس به خاطر داشته باشید که برای overload کردن متد پارامترها باید متفاوت باشند.
08:58	داشتن return types مختلف method را overload نخواهد کرد.
09:03	پس این مقدار را حذف کنید و فایل را Save کنید.
09:09	و این چگونگی method overloading در Java می باشد.
09:16	پس در این برنامه این موارد را یاد گرفتیم:
09:18	در مورد method overloading
09:20	و overload method
09:22	و مزایای method overloading
09:25	برای ارزیابی خود method subtract که اعداد را از هم کم می کند را ایجاد کنید.
09:31	آن را overload کنید.
09:33	برای دانستن بیشتر Spoken Tutorial Project ویدئو را در لینک زیر ببینید.
09:36	http://spoken-tutorial.org/What_is_a_Spoken_Tutorial
09:42	این خلاصه پروژه می باشد. اگر بهنای باند خوبی ندارید ابتدا دانلود و سپس مشاهده کنید.
09:45	
09:48	
09:50	تیم این پروژه کارگاه آموزشی استفاده از Spoken Tutorial ارائه می دهد. و به کسانی که آزمون آنلاین را قبول شوند، گواهینامه می دهد .
09:52	
09:56	برای جزئیات بیشتر لطفاً به contact AT spoken HYPHEN tutorial DOT org ایمیل بفرستید
10:01	Spoken Tutorial Project بخشی از پروژه Talk To a Teacher می باشد. که تحت پشتیبانی National
10:05	Mission on Education از طریق ICT توسط MHRD دولت هند می باشد.
10:11	اطلاعات بیشتر در لینک spoken hyphen tutorial dot org Slash NMEICT hyphen Intro در دسترس می باشد:
10:19	به پایان برنامه رسیده ایم.
10:21	ترجمه و صداگذاری شبنم اقبال خان. با تشکر از شما