

Time	Narration
00:01	به برنامه آموزشی Sets in Python خوش آمدید.
00:06	در این برنامه یاد می‌گیرید که: sets از lists را ایجاد کنید.
00:12	union, intersection و symmetric difference operations را انجام دهید.
00:18	بررسی کنیم که اگر یک set از subset دیگریست و تشابهات مختلف با lists را بفهمیم.
00:27	برای ضبط این برنامه من از سیستم عامل Ubuntu Linux 16.04 Python 3.4.3 و IPython 5.1.0 استفاده می‌کنم.
00:35	برای تمرین این برنامه باید بدانید که چگونه فرمانهای پایه Python را در ipython console را استفاده کنید. و استفاده از lists
00:42	اگر نه برنامه آموزشی Python مربوطه در این وب سایت را ببینید.
00:53	ابتدا مروری در مورد sets می‌کنیم.
00:58	Sets یک مجموعه بدون ترتیب از elements منحصر به فرد می‌باشد.
01:02	خود set که mutable می‌باشد. ما می‌توانیم items را از آن حذف و یا اضافه کنیم.
01:07	ipython را شروع می‌کنیم.
01:14	terminal را باز کنید.
01:19	ipython3 را تایپ کنید و Enter را فشار دهید.
01:24	از اینجا به بعد ، به یاد داشته باشید که پس از تایپ هر فرمان در terminal ، کلید Enter را فشار دهید.
01:31	حالا می‌بینیم که چگونه sets را وارد کنیم.
01:35	a underscore list مساوی داخل 1 comma 2 comma 1 comma 4 comma 5 comma 6 comma 2 را تایپ کنید.

01:47	a مساوی set داخل براکت a underscore list را تایپ کنید.
01:54	سپس a را تایپ کنید.
01:57	می‌بینیم که تکراری‌ها حذف شده‌اند و set فقط شامل elements منحصر به فرد می‌باشد.
02:03	Sets را همچنین می‌توان مستقیماً به این صورت ایجاد کرد:
02:08	b مساوی داخل curly braces 1 comma 2 comma 1 comma 4 comma 5 comma 6 comma 2 را تایپ کنید.
02:19	b. را تایپ کنید. می‌بینیم که set فقط شامل elements منحصر به فرد می‌باشد.
02:27	برای ایجاد یک empty set شما c مساوی set open and close brackets را تایپ کنید.
02:36	توجه کنید که d مساوی open and close curly braces یک دیکشنری خالی را می‌دهد نه یک set خالی.
02:45	Sets می‌تواند شامل numbers, strings و tuples باشد. اما نمی‌تواند شامل mutable elements مثل lists یا dictionaries باشد.
02:56	b underscore list مساوی داخل square brackets 1 comma 2 comma 1 comma 6 comma 2 دوباره داخل square brackets 1 comma 2 comma 1 comma 6 comma 2 را تایپ کنید.
03:09	حالا b مساوی set داخل براکت b underscore list را تایپ کنید.
03:17	همانطور که می‌بینید این TypeError را می‌دهد.
03:21	حالا تعدادی operations روی sets انجام می‌دهیم و برای این ابتدا یک جفت sets را ایجاد می‌کنیم.
03:29	sets را همانطور که نشان داده شده تایپ کنید.
03:33	اینجا f یک set از اعداد fibonacci از 1 تا 10 می‌باشد.
03:39	و p یک set از اعداد prime از 1 تا 15 می‌باشد.
03:44	operations های مختلف را می‌توان روی sets انجام داد.
03:47	ابتدا ما یک element را به set با استفاده از add method اضافه می‌کنیم.
03:53	f dot add inside brackets 13 را تایپ کنید.
03:58	حالا f را تایپ کنید. همانطور که می‌بینید 13 به f set اضافه شده است.
04:06	add method هیچ تأثیری ندارد اگر element از قبل موجود باشد.
04:11	f dot add داخل براکت 13 را تایپ کنید.
04:16	f را تایپ کنید. همانطور که می‌بینید این دفعه 13 اضافه نشده است.
04:23	سپس یاد می‌گیریم که یک element را با استفاده از remove method حذف کنیم.
04:28	p dot remove داخل براکت 13 را تایپ کنید.
04:34	حالا p را تایپ کنید. همانطور که می‌بینید 13 از p set حذف شده است.

04:41	اگر element یک member نباشد این KeyError می دهد.
04:46	p dot remove داخل براکت 18 را تایپ کنید.
04:52	KeyError می دهد چون element 18 در p موجود نمی باشد.
04:57	pipe character که union را نشان می دهد.
05:01	f pipe p را تایپ کنید
05:06	یا f dot union داخل براکت p را تایپ کنید. این union از f و p را می دهد.
05:16	ampersand character که intersection را نشان می دهد.
05:20	f ampersand p را تایپ کنید. یا f dot intersection داخل براکت p را تایپ کنید.
05:30	این intersection از f و p را می دهد.
05:34	f minus p را تایپ کنید. یا f dot difference داخل براکت p را تایپ کنید.
05:44	این تمام elements که در f هستند اما در p نیستند را می دهد.
05:50	f caret p را تایپ کنید.
05:54	یا f dot symmetric underscore difference داخل براکت p را تایپ کنید.
06:01	این همه elements در f union p را می دهد اما نه در f intersection p
06:08	به زبان ریاضی این symmetric difference را می دهد.
06:13	Sets همچنین بررسی subsets را حمایت می کند.
06:17	a مساوی set داخل براکت داخل 1 comma 2 comma 3 comma 4 square brackets را تایپ کنید.
06:27	b مساوی set داخل براکت داخل 1 comma 2 square brackets را تایپ کنید.
06:35	حالا b کمتر یا مساوی a را تایپ کنید. یا b dot issubset a داخل براکت a را تایپ کنید.
06:47	این True را می دهد چون b یک subset از a می باشد.
06:52	Sets همچنین بررسی کردن supersets را حمایت می کند. b بیشتر یا مساوی a را تایپ کنید.
07:01	یا b dot issuperset a داخل براکت a را تایپ کنید.
07:08	ما False را بدست می آوریم چون b یک superset از a نمی باشد.
07:14	هر set یک subset و همچنین یک superset از خودش می باشد.
07:20	a کمتر از یا مساوی a a بیشتر از یا مساوی a را تایپ کنید.
07:28	این True را در هر دو مورد می دهد چون a یک superset و subset از خودش می باشد.

07:35	به Elements های یک set با استفاده از for loop دسترسی پیدا می کنیم.
07:39	for x in a colon print داخل براکت x را تایپ کنید.
07:47	items از set به ترتیب خاص ظاهر نمی شود.
07:52	بررسی length و containership در sets مشابه lists و tuples می باشد.
07:59	len داخل براکت a را تایپ کنید. این 4 را نشان می دهد.
08:07	in a 1 in a 7 را تایپ کنید.
08:13	این True و False را پرینت می کند.
08:18	Sets از indexing حمایت نمی کند. بنابراین slicing و striding در sets معتبر نمی باشند.
08:25	ویدیو را متوقف کنید. این تمرین را انجام دهید و سپس به ویدیو برگردید.
08:31	لیست نمرات همانطور که نشان داده شده را داریم. تمام اعداد تکراری را لیست کنید.
08:37	برای حل کردن به terminal بروید.
08:41	marks مساوی داخل square brackets 20 comma 23 comma 22 comma 23 comma 20 comma 21 comma 23 را تایپ کنید.
08:53	marks underscore set مساوی set داخل براکت marks
09:00	for num in marks underscore set colon marks dot remove داخل براکت num
09:10	remove method فقط اولین رخداد element از list را حذف می کند.
09:17	duplicates مساوی set داخل براکت marks را تایپ کنید.
09:24	حالا duplicates را تایپ کنید.
09:28	ما حالا نمرات تکراری در set duplicates داریم. بنابراین ما جواب مورد نظر را بدست آورده ایم.
09:37	به پایان این برنامه می رسیم. خلاصه می کنیم.
09:43	در این برنامه یاد گرفتیم که sets از lists درست کنیم یا با استفاده از curly braces ,
09:50	union, intersection و symmetric difference operations را انجام دهیم,
09:55	با استفاده از less than or equal to operator بررسی کنیم که آیا یک set یک subset از دیگریست
10:02	و تشابهات مختلف با lists مثل طول و containership را فهمیدیم.

10:08	و تمرین برای شما:
10:12	اول. اگر a به صورت زیر داده شده است, set از a چیست؟
10:18	دوم. odd و squares به این صورت داده شده است. چگونه symmetric difference از این دو sets را بدست می آورید.
10:27	سوم. اگر a یک set است چگونه بررسی میکنید که آیا متغیر b در a موجود است؟
10:34	و پاسخ ها: اول. set از a شامل همه elements مشترک در a list می باشد. که 1 5 3 comma 2 comma 8 می باشند.
10:46	دوم. برای پیدا کردن symmetric difference بین دو sets ما از operator caret استفاده می کنیم. پس odd caret squares را می‌توانیم تایپ کنیم.
10:56	سوم. برای بررسی containership ما b in a را تایپ می کنیم.
11:01	لطفاً سؤالات زمان بندی شده خود را به این انجمن بفرستید.
11:05	لطفاً سؤالات کلی خود در مورد Python را به این انجمن بفرستید.
11:10	تیم FOSSEE پروژه TBC را هماهنگ می کند.
11:15	بودجه پروژه Spoken Tutorial توسط NMEICT, MHRD, دولت هند تأمین می شود. به این سایت مراجعه کنید.
11:25	ترجمه و صدا گذاری شبنم اقبال از IIT Bombay. با تشکر از شما