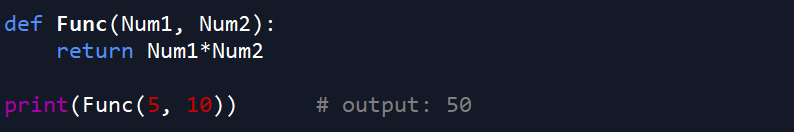
*Lambda*

* **تعریف لامبدا ( Lambda )**

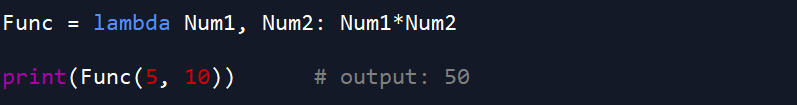
روشی ساده برای تعریف تابع ( Function ) در پایتون می باشد . این توابع معمولا به نام **توابع لامبدا** یا **عملگرهای لامبدا** نامیده می شوند .

* **طریقه نوشتن تابع با استفاده از کلمه کلیدی def**

پیش از بررسی توابع لامبدا ( Lambda ) یک تابع ساده با استفاده از کلمه کلیدی def می نویسیم .



اگر بخواهیم همین تابع را با عبارت لامبدا ( Lambda ) بنویسیم ، اینگونه است :



**نکته :** از lambda به جای def استفاده شده است .

**نکته :** کلماتی که بعد از کلید واژۀ lambda می آیند ، پارامترها هستند .

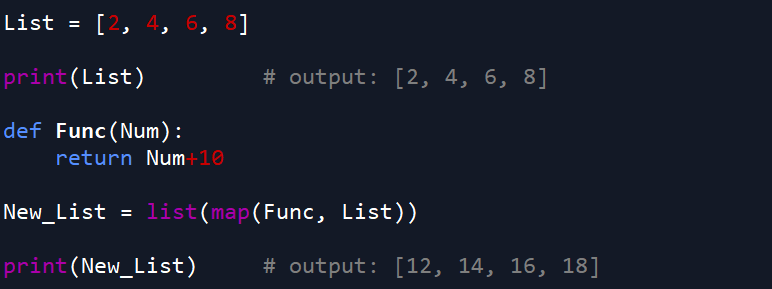
**نکته :** از علامت ":" برای جدا کردن عبارت و پارامترها استفاده می شود .

**نکته :** نیازی به کلید واژۀ return نیست ، چرا که lambda خود این عملکرد را به صورت خودکار انجام می دهد .

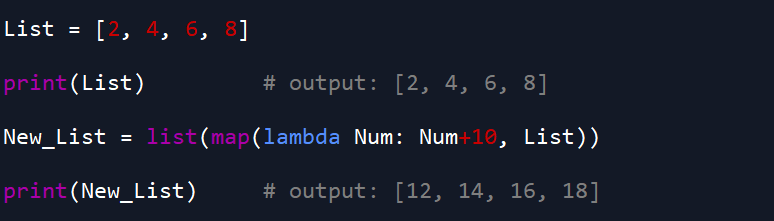
* **نگاشت ( Map )**

تابع نگاشت ( Map ) دو آرگومان ورودی می‌گیرد که یکی تابع ( Function ) و دیگری لیست ( List ) است . این تابع ، از تابع ورودی استفاده کرده و آن را روی لیست اجرا می‌کند و لیست اصلاح شده را به صورت یک شیء نگاشت ( Map ) باز می‌گرداند .

استفاده از نگاشت ( Map ) بدون لامبدا ( Lambda ) :



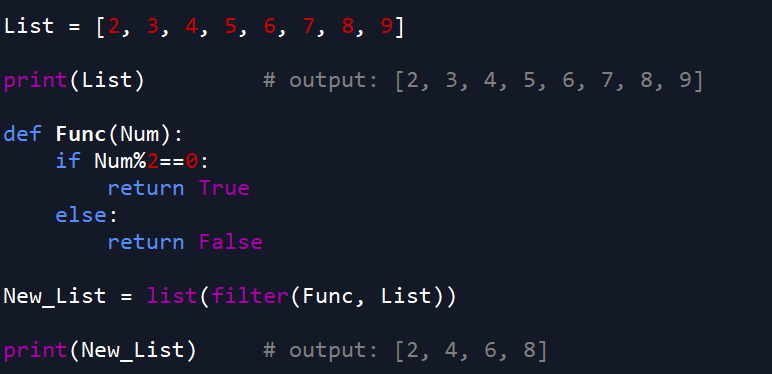
استفاده از نگاشت ( Map ) همراه لامبدا ( Lambda ) :



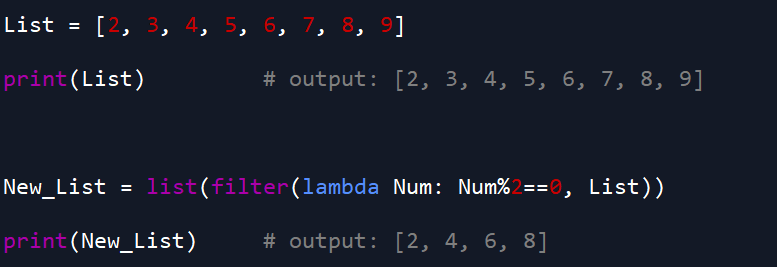
* **فیلتر ( Filter )**

Filter یک تابع را می‌گیرد و آن را بر روی همه عناصر یک فهرست اعمال می‌کند و لیست جدیدی ایجاد می‌کند .

استفاده از فیلتر ( Filter ) بدون لامبدا ( Lambda ) :



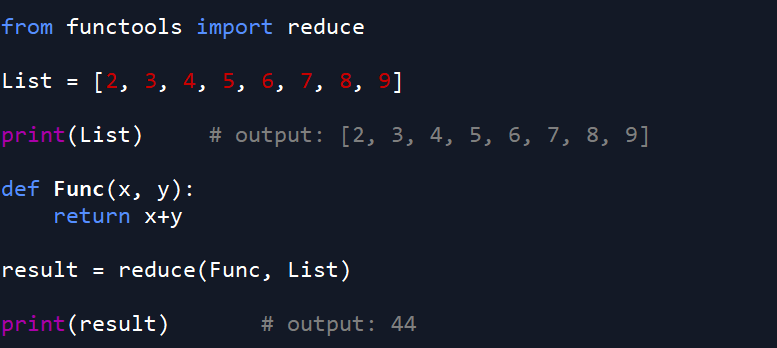
استفاده از فیلتر ( Filter ) همراه لامبدا ( Lambda ) :



* **کاهش ( Reduce )**

Reduce یک محاسبه چرخشی بر روی همه عناصر یک لیست انجام می‌دهد . از این تابع برای محاسبه جمع یا ضرب کردن همه اعداد لیست با هم می‌توان استفاده کرد .

استفاده از کاهش ( Reduce ) بدون لامبدا ( Lambda ) :



استفاده از کاهش ( Reduce ) همراه لامبدا ( Lambda ) :

