توابع در پایتون :

به هر بلوک از کد که به بار های متعدد قابل اجرا شدن هستن توابع گفته میشوند.

توابع میتوانند ورودی و خروجی داشته باشند

---------------------------------------------------------------------------------------

سینتکس توابع :

Def name (args):

Expressions

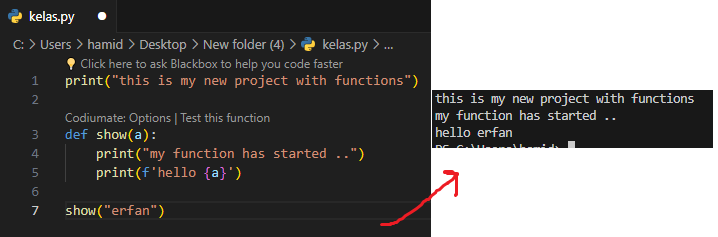
Return \*(optional)

name(args)

اگر ارگومانی قرار است تابع بگیرد موقع صدا زده شدن باید به آن پاس داده شوند

-----------------------------------------------------------------------------------------------

توابع برای اجرا شدن نیاز دارند که صدا زده شوند :



تابع نوشته شده که یک ارگومان a رو بهش دادیم و الان هرچی به تابع موقع صدا زدن بدیم داخل a ریخته میشود بعد داخل تابع یک سری کد هست که با اجرا شدن اجرا میشوند

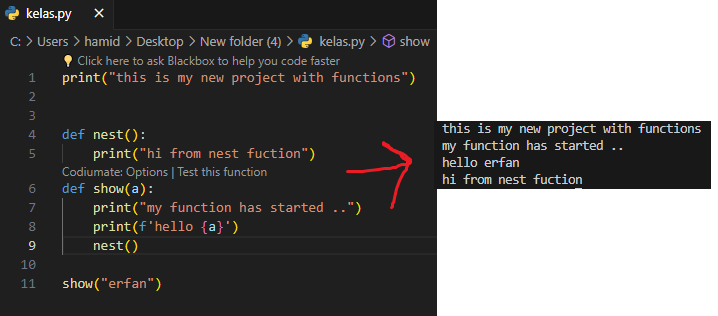
آخر هم تابع رو صدا زدیم و یک مقدار رو بهش به عنوان ارگومان دادیم تا داخل a ریخته بشود

---------------------------------------------------------------------------------------

مثل سی پلاس پلاس نیست که ما باید توابع رو بالا بنویسیم و بعد پایین اجرا کنیم توابع هر کجا که دوست داشته باشیم نوشته میشوند و هر کجا که دوست داشته باشیم اجرا میشوند

---------------------------------------------------------------------------------------

توابع هم میتوانند تو در تو بشوند :



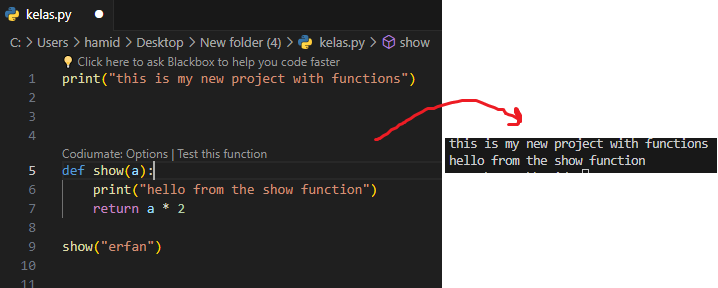
اینجا اومدیم تابع nest رو داخل تابع show صدا زدیم

---------------------------------------------------------------------------------------

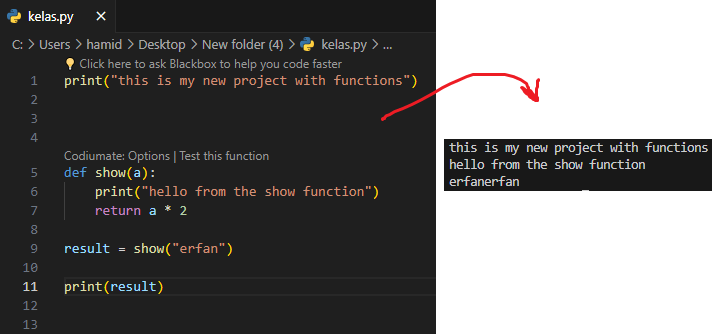
توابع return دار :

اگر بخواهیم تابع بعد از اتمام مقداری را برگرداند و جایی بفرستد یا ذخیره کند ما از return استفاده میکنیم

یه مثال براتون بزنم جا بیوفته : متد clear در توابع رو وقتی پرینت میکردیم None میداد چرا ؟ چون چیزی رو برنمیگردوند و فقط کد های داخلیش اجرا میشد ( لیست را خالی میکرد ) ولی مثلا تابع copy میاد و اون کپی از لیست رو return میکنه و ما مثلا میریزیمش داخل یه متغیر یا پرینت میکنیمش.

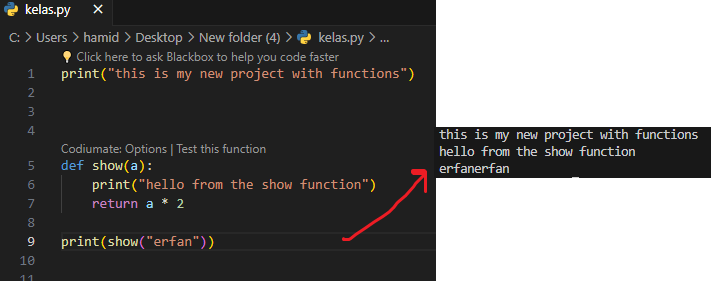


اینجا میبینید که a ضرب در 2 در خروجی چاپ نشده چرا > چون فقط برگردونده شده یعنی چاپ نشده یا در متغیری ریخته نشده :



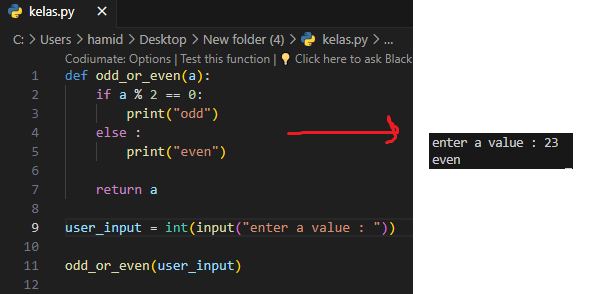
حالا اون چیزی که return میشه داخل متغیر result ذخیره میشه و ا اونو پرینت کردیم

میشه اون رو متسقیما هم پرینت کرد ( گفتم زمانی از متغیر استفاده میکنیم که بخواییم اون چیزی که return شده رو بعدا استفاده کنیم ) :



اینجا اون چیز return میشه داخل print و اونو ذخیره نکردیم و فقط مستیقیما چاپ کردیم

گفتیم که توابع وظیفه کوتاه و سازمان دهی کردن کد های مارو دارند :

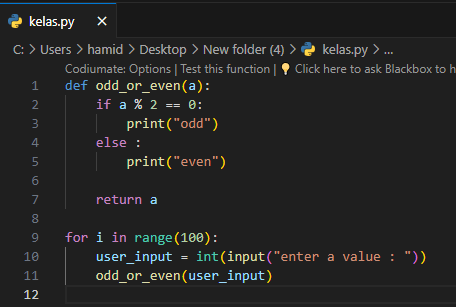


اینجا داخل تابع بحث زوج و فرد بودن رو برسی کردیم که اون ارگومان ورودی اگر باقیماندش بر 2 صفر بود چاپ کنه زوج وکرنه چاپ کنه فرد

پایین هم ورودی رو از کاربر گرفتیم و تبدیل ه عددش کردیم بعد تابع رو صدا زدیم و بهش پاسش دادیم

اون return هم به محض احتیاط گذاشتیم تا اگر برنامه نویس دیگه ای داشت با کد ما کار میکرد و دلش خواست مقدار a رو جایی ذخیره کنه ازش استفاده کنه

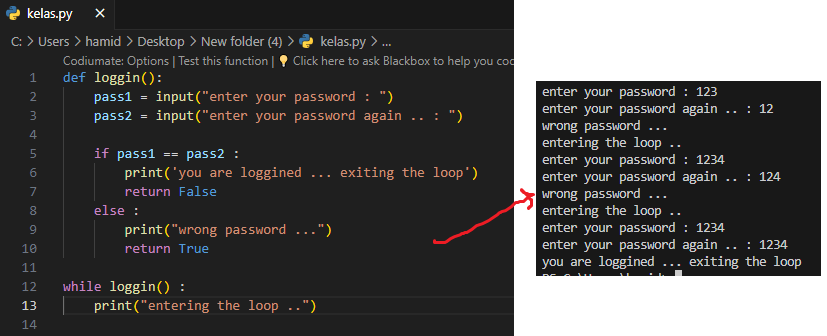
حالا 100 بار هم که شده نیازی نیست کد رو هعی بنویسیم بجاش تابع رو 100 بار صدا میزنیم :



الان 99 بار حلقه میخوره و 99 بار تابع اجرا میشه

---------------------------------------------------------------------------------------

یک مثال دیگه که جا بیوفته و همچنین کاربری در جنگو :



توی این مثال اومدیم کاری کردیم که اگر پسور 1 با 2 برابر بود False برگردد و اگر نبود True چرا ؟ اگر پسورد غلط بود True برگردونده میشه به شرط حلقه و حلقه ادامه پیدا میکنه و هر دفه تابع اجرا میشه . اگر پسورد درست بود False برمیگرده به شرط حلقه و حلقه بسه میشه

---------------------------------------------------------------------------------------