حالا، یکی از ویژگی هایی که همراه کاتلین میاد اینه که توابعمون میتونن مقادیر پیشفرض داشته باشند. مثلا تابع sum رو نوشیتم که یادتون میاد.خب حالا فرض کنین من میخوام یک مقدار z هم در نظر بگیرم، که البته بعضی جاها نیاز به 2 پارامتر و بعضی جاها نیاز به 3 پارامتر داریم! یعنی بعضی جاها پارامتر سوممون به صورت دیفالت صفر باید درنظر گرفته بشه.

fun sum**(**x:Int, y:Int, z:Int=0**)** = x+y+z  
  
fun main**(**args: Array<String>**)** {  
 sum**(**1,2,3**)** sum**(**1,2**)**}

نگاه کنین، کامپایلر میاد نگاه میکنه،اگه پارامتر سوم فراهم شده بود که همون مقدار رو استفاده میکنه، اگه پارامتر سوم فراهم نشده بود از مقدار دیفالت استفاده میکنه! و این خوبه، چراکه بهمون این قابلیت رو میده که یک تابع با چندین پارامتر دیفالت داشته باشیم و از تکرار بی مورد(overload) توابع جلوگیری کنیم. مثلا توی جاوا اگر میخواستیم یک تابع داشته باشیم که در بعضی مواقع دو ورودی میگرفت و در بعضی مواقع 3 ورودی، این کار امکان پذیر نبود و برای عملی شدن آن باید 2 تابع مینوشتیمف یکی با 2 ورودی و یکی با 3 ورودی که خب این مشکل در کاتلین رفع شده است.

خب بیاین یک تابع دیگه رو مثال بزنیم،مثلا

fun printDetails**(**name: String,email: String="",phone: String**)**{  
 println**(**"name : $name - email : $email - phone : $phone"**)**}

اومدم یک تابع درست کردم که مقدار اول و سوم رو درخواست میکنه ولی اگه مقدار دوم وارد نشد به صورت دیفالت یک مقدار جایگزینش میشه. حالا کد main رو هم مینویسم

printDetails**(**"Sina Darvishi","555 87 345"**)**

برای مقدار name نام خودم رو دادم و برای مقدار شماره هم شماره خودم رو وارد کردم. ولی وقتی اجراش کنیم متوجه میشیم که ارور میده، دلیلش هم اینه که مقدار شماره تلفن رو که من به عنوان دومین ورودی وارد کردم رو به عنوان email میگیره! در واقع میمونه که شما دارین پارامتر phone رو پاس میدین یا پارامتر email . برای رفع این مشکل هم راه حل خیلی ساده ای داریم، کاتلین این قابلیت رو توی کد نویسی به توسعه دهنده میده که وقتی میخواین به یک تابع مقدار بدین، دقیق مشخص کنین که به کدوم پارامتر میخواین مقدار بدین، پس کد درست بالا میشه:

printDetails**(**"Sina Darvishi",phone = "555 87 345"**)**

در ضمن این خاصیت این قابلیت رو به ما میده که ما خارج از ترتیبی که تابع رو تعریف و پارامتر های اونو درخواست کردیم به تابع ورودی بدیم.

در ضمن این قابلیت در واقع اونجایی خوبه که ما تعداد زیادی پارامتر ورودی میخوایم و در واقع گیج میشیم که کدوم ورودی برای کدوم پارامتره و ما با نام دهی به مقادیری که میخوایم به تابع پاس بدیم از این گیج شدن جلوگیری میکنیم.