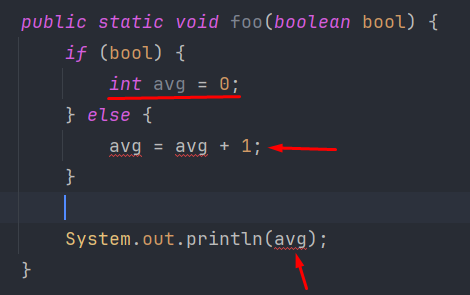
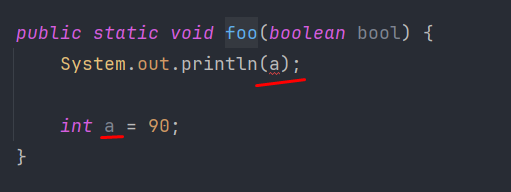
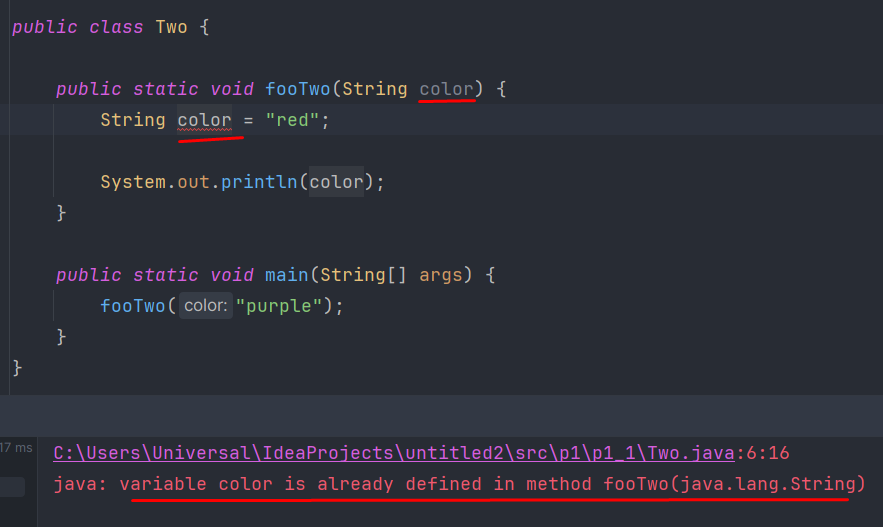
Java da **if-else** block ham **scope** hisoblanadi, yo **if** ni yo **else** scope ni ichida e’lon qilingan variable bir-birida ko’rinmaydi. Pastda xuddi shu narsa ko’rsatilgan. **If** blockni ichida e’lon qilingan **avg** variable **else** blockda ko’rinmaydi. Demak **avg** block faqatgina **if** ni ichida yaroqli va ko’rinadi:



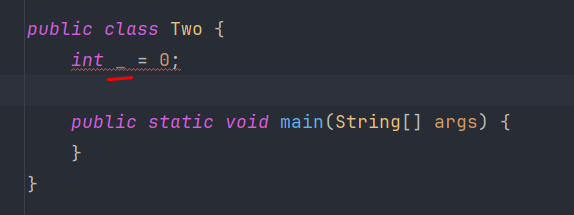
Javada methodda o’zgaruvchi e’lon qilinishidan oldin, unga murojaat qilib bo’lmaydi aks holda compile xatolik beradi:



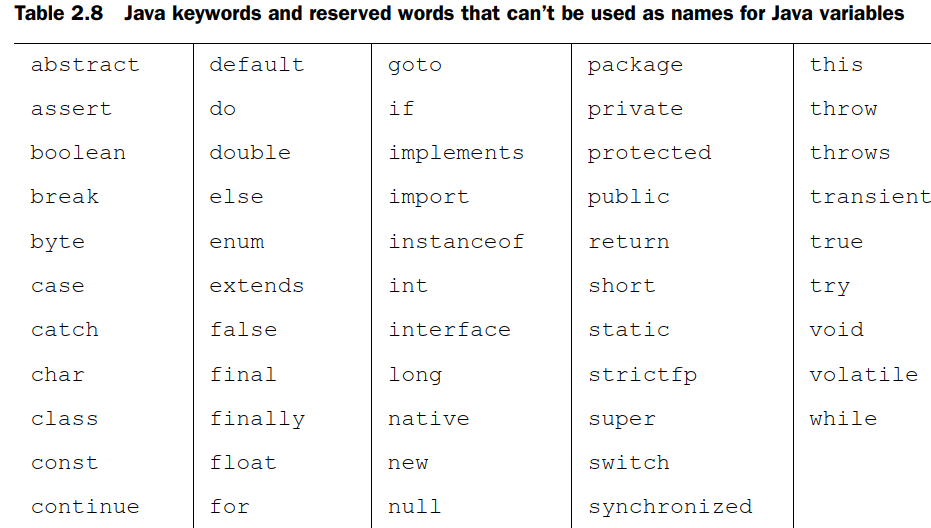
Javada methodni parametrida yozilgan o’zgaruvchini nomini yana qaytadan shu methodni ichida e’lon qila olmaymiz xatolik beradi. Pastda **fooTwo** methodni parametrida **color** nomli o’zgaruvchi yozilgan, shu **color** ni yana qaytadan shu methodni ichida e’lon qila olmaymiz. Chunki bu o’zgaruvchi allaqachon methodni parametrida e’lon qilingan:



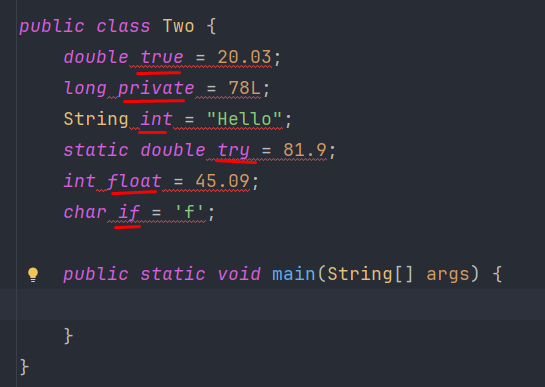
Javada **\_ (underscore)**ni o’zini o’zgaruvchini nomini yozishda ishlata olmaymiz, xatolik beradi compileda:



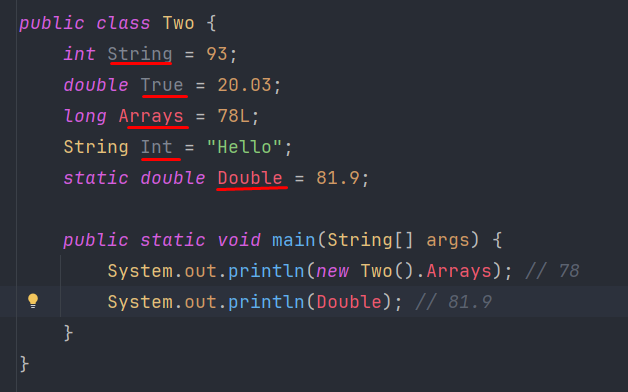
Javada o’zgaruvchilarni nomini e’lon qilishda e’tiborli bo’lish kerak. Reserved wordlarni ishlatmaslik kerak o’zgaruvchilarni nomini yaratishda.



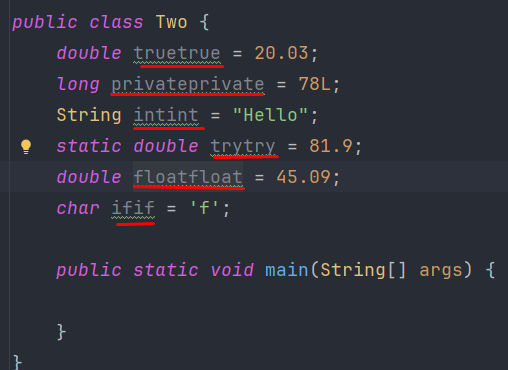
Aks holda xatolik beradi:



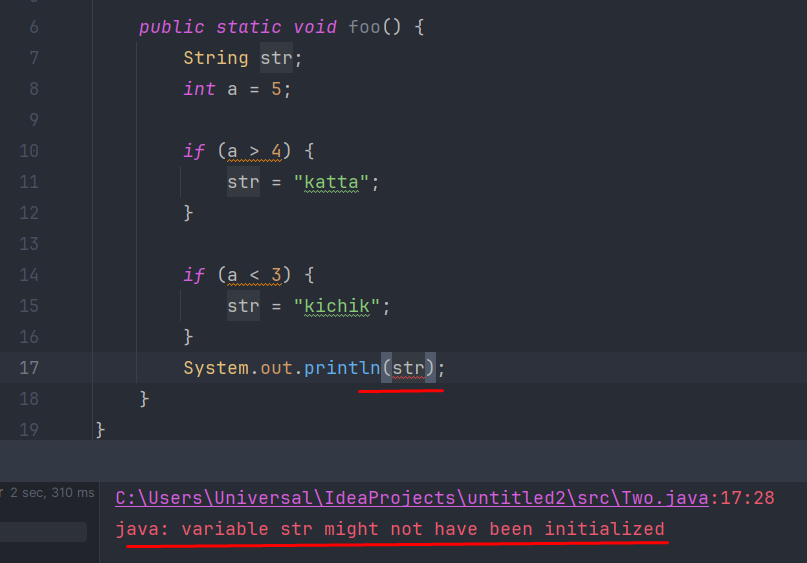
Lekin yuqoridagi ro’yxatdan tashqari boshqa javada classlarni nomini masalan **Arrays**, **String** yoki o’sha reserved wordlarni katta harflar bilan yozsak, masalan **int** => **Int**, **long** => **Long**,… kabi yozsak xato bo’lmaydi:



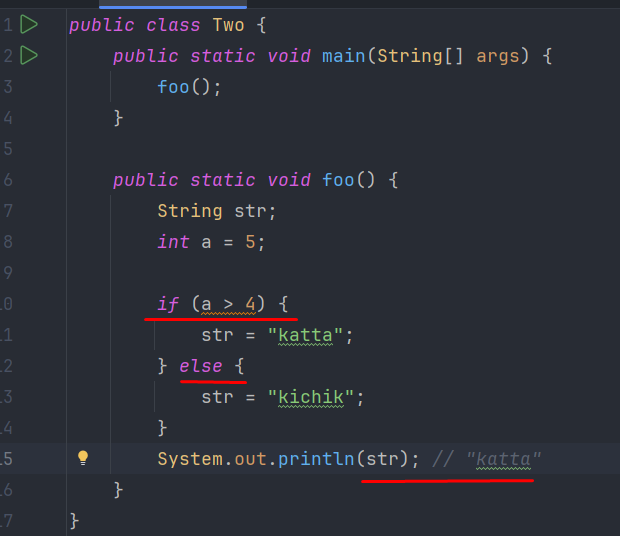
Yoki reserved wordlarni qo’shib yozsak ham xato bo’lmaydi:



Javada local o’zgaruvchini e’lon qilib, lekin unga boshlang’ich qiymat bermasak, u holda bu o’zgaruvchiga if shart operatorini ichida, shartga tekshirib qiymat beradigan bo’lsak, xatolik beradi compile timeda. Sababi if ni ichiga tushmay qoladigan bo’lsa, u holda bu local o’zgaruvchiga boshlang’ich qiymat berib bo’lmaydi, shuning uchun xatolik beradi:



Lekin to’g’ridan to’g’ri if-else ga solsak, xatolik bermaydi. Sababi, istalgan if shart qanoatlantirmasa, baribir else qanoatlantiradi. Bu yerda nima bo’lgan taqdirda ham, **str** local o’zgaruvchiga qiymat o’zlashtiriladi:



Pastdagi misolda qiziq bir holat ko’rsatilgan. **6**-qatorda **pro** nomli object yaratildi. Bu obyektni ichida **y** o’zgaruvchisini qiymati **y=33** ga tengdir, buni **7-**qatorda ham ko’rish mumkin. Undan keyin **9-**qatorda **foo()** nomli methodni chaqiryapmiz. Bu methodni ichida **18**-qatorda **y=12** qilib, **pro** objectni ichidagi **y** ni qiymatini **33** dan **12** ga o’zgartirdik. Endi **pro** objectni ichida **y** ni qiymati, **foo()** nomli methodni chaqirganimizdan keyin **12** ga o’zgardi, shuning uchun eski **y=33** ni rangini sariq bilan bo’yab qo’ydik, bu degani **y=33** dan **y=12** ga o’zgardi degani. Bu yerda e’tiborli joyi **foo()** methodni chaqirishdan oldin va keyin instance **foo()** method instance **y** o’zgaruvchisini o’zgartiryapti:

