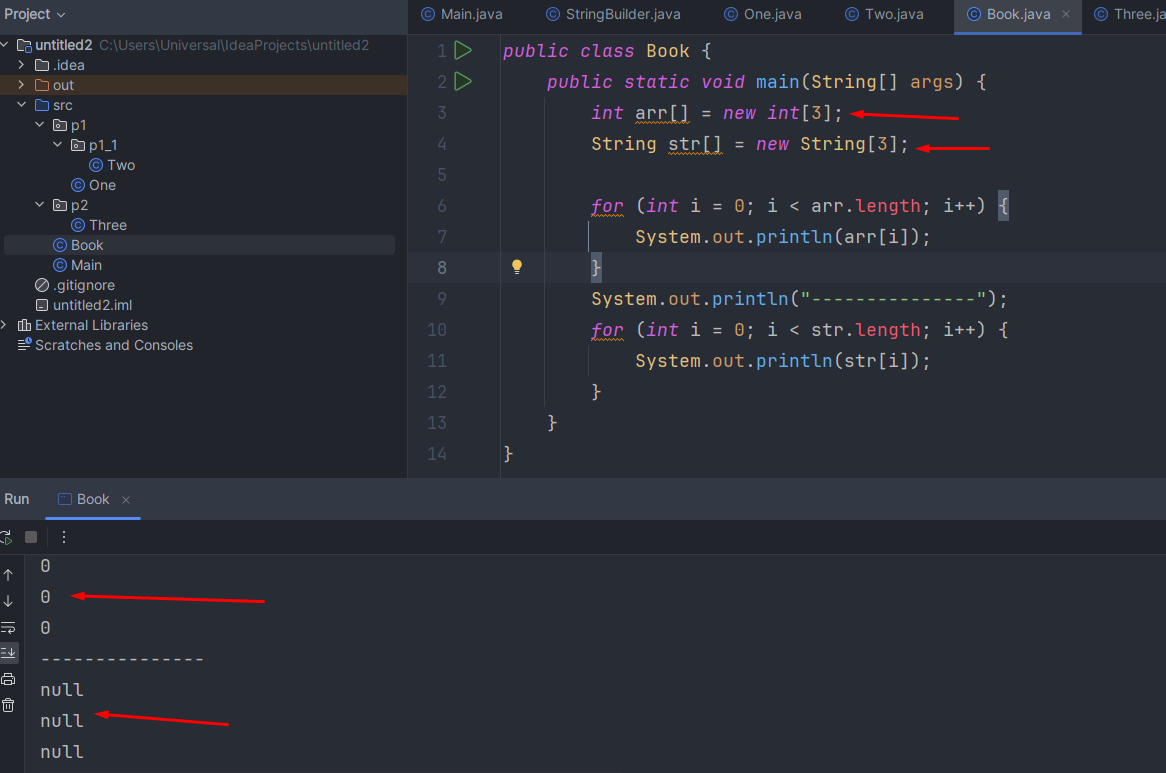
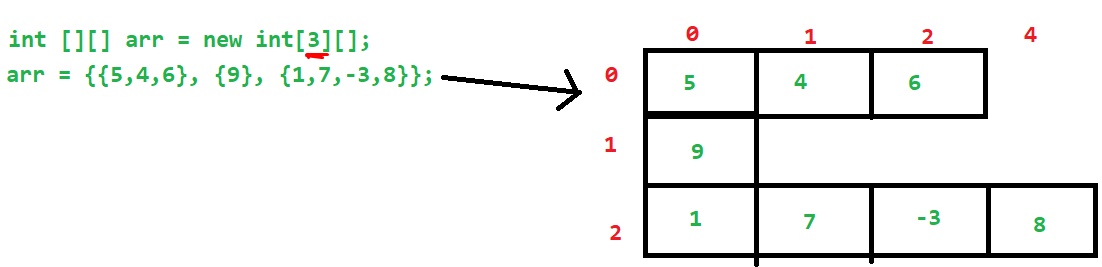
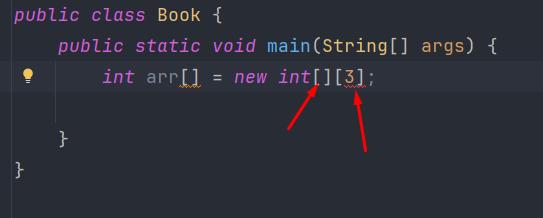
Agar array e’lon qilinsa, lekin unga boshlang’ich qiymat berilmasa, u holda unga o’zini type ga mos boshlang’ich qiymat beriladi. Masalan, massivni type int typeda bo’lsa, unga 0, double type da bo’lsa 0.0, Boolean type da bo’lsa false, reference type bo’lsa null. Pastda ham xuddi shunday arr massivni type int bo’lgani uchun, boshlang’ich qiymati 0, str massivni type String bo’lgani uchun, boshlang’ich qiymati null dir:



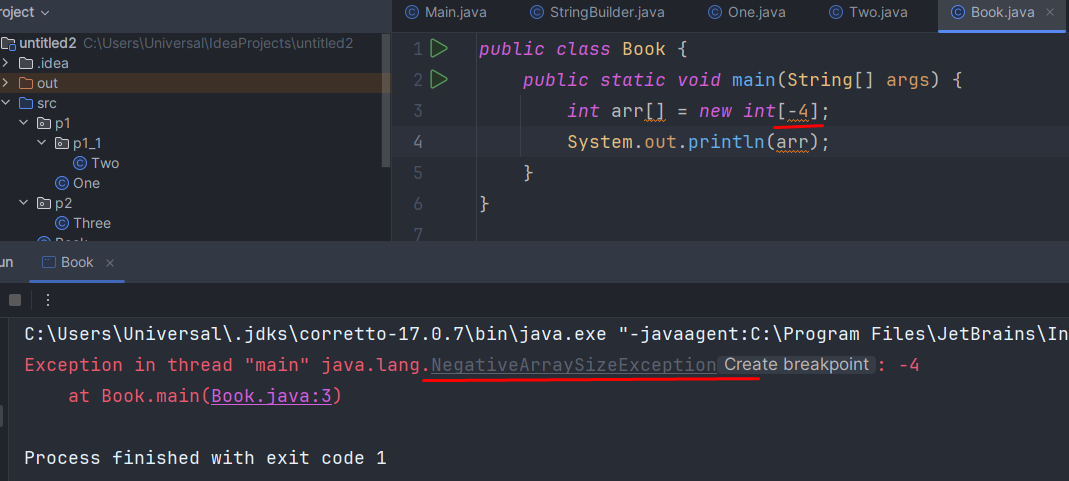
Agar massivda har xil o’lchamdan tashkil topgan 2 o’lchamli massiv berilgan bo’lsa, u holda bu massivni o’lchamini yozishda har doim row(i) ni uzunligi birinchi yoziladi va column(j)ni uzunligini bermaymiz. Agar teskarisini qilib row ni uzunligini bermay, columnni uzunligini beradigan bo’lsak, u holda compile error beradi. Pastdagi misolda ham massiv 3 ta row dan tashkil topgan, columnni uzunligi har xil, masalan 1-rowda 3 ta column(5,4,6) bo’lsa, 2-rowda 1 ta column(9), 3-rowda esa 4 ta column(1,7,-3,8) bor.



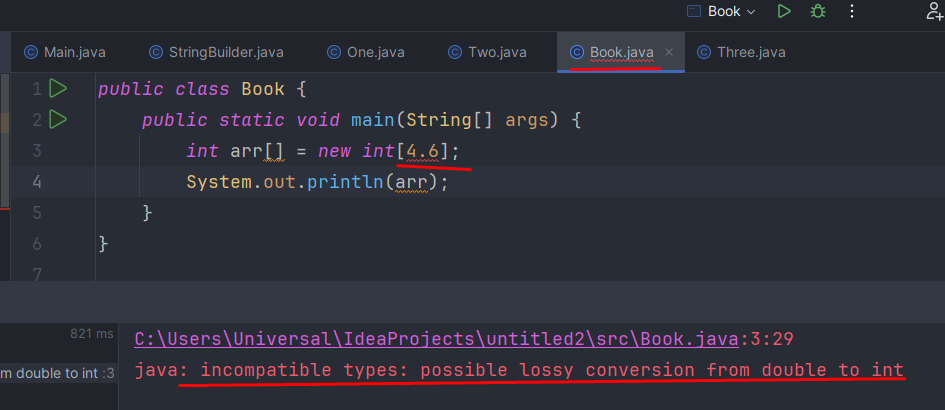
Boya yuqorida ko’rganimizdek, row ni uzunligini tashlab ketib, columnni yozsak compile error beradi. Shuning uchun har doim row yozilishi shart, lekin column ixtiyoriydir:



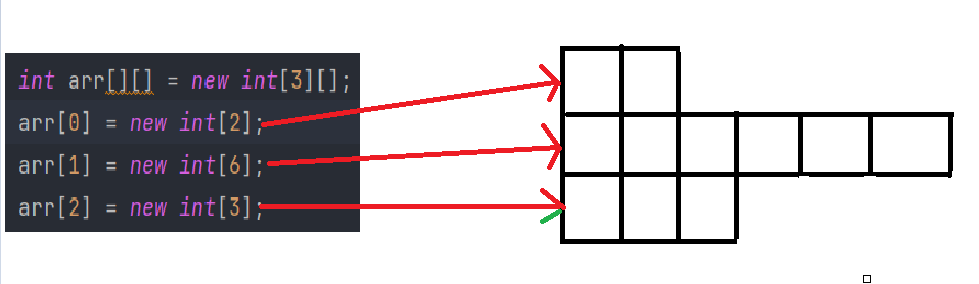
Masssivni uzunligi har doim **int** typedagi son bo’lishi kerak. Massivni uzunligiga istasak **manfiy butun son** berishimiz mumkin, chunki manfiy butun son ham **int** typedadir. Lekin Compile da xatolik bermaydi, lekin **runtime** da xatolik beradi. Shuning uchun berish mumkin emas:



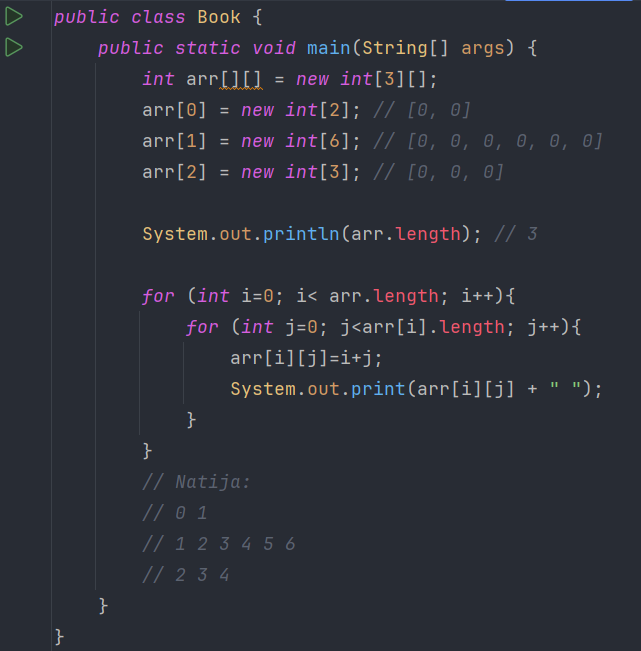
Massivni uzunligiga **float** yoki **double** kabi sonlar berish mumkin emas. Chunki faqat **int** type qabul qiladi, agar bersak compileda xatolik beradi:



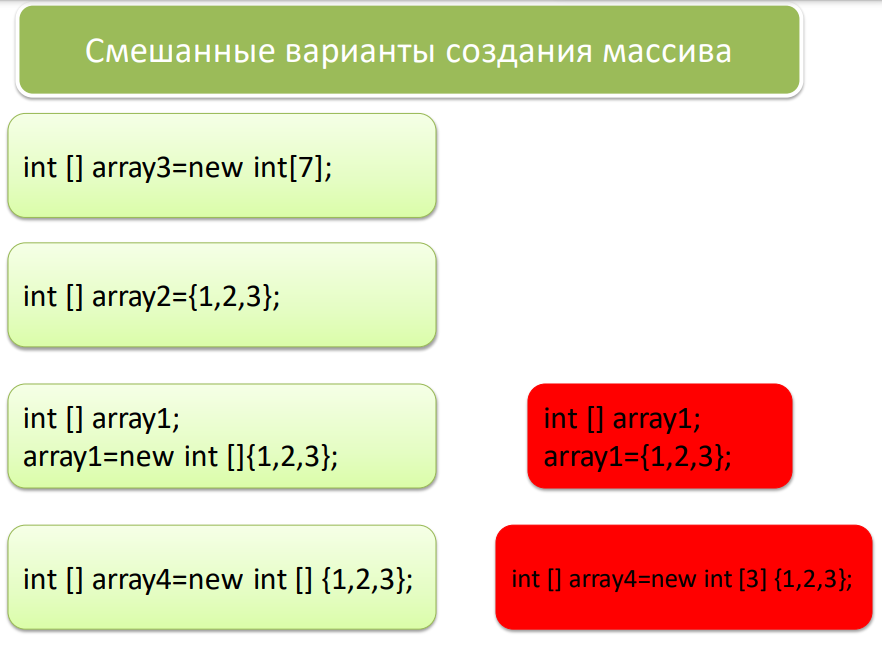
2 o’lchamli massivni yaratish: 3 ta row dan iborat, columlari esa har xil bo’lgan:



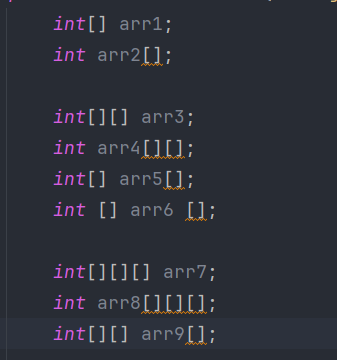
Yuqoridagi misolni dynamic initialization usulidan foydalanib, massivga qiymat beramiz. Buning uchun for loopdan foydalanamiz:



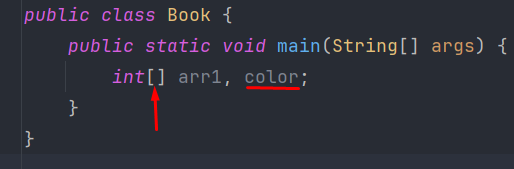
Pastda qizil rang bilan ko’rsatilgan usul bilan massivni e’lon qilish mumkin emas, xatolik beradi. Shu joylarini yaxshilab eslab qolish kerak. imtihonda tushushi mumkin:



Pastdagi holatda **[]** belgini o’zgaruvchidan oldin yoki keyin qo’yish xato emas. Bundan tashqari agar massivimiz ko’p o’lchamli bo’lsa, u holda **[]** belgini bir nechtasini o’zgaruvchidan oldin yoki keyin ham yozish mumkin. Hammasi to’g’ri hisoblanadi:



Pastdagi misolda **color** variable massiv hisoblanadi. Chunki **[]** belgigacha **arr1** va **color** o’zgaruvchilari e’lon qilib bo’lingan:



Bunday yozsak, u holda **color** o’zgaruvchi shunchaki **primitive** type bo’lib qoladi:



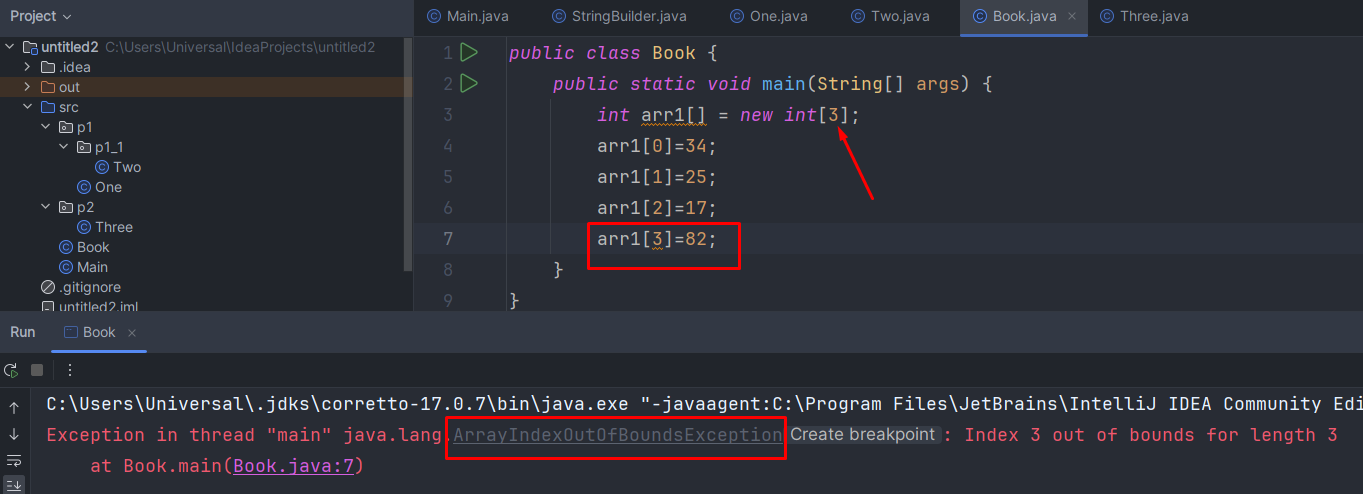
Bunday yozsak, u holda arr1 bir o’lchamli massiv, color esa 3 o’lchamli massiv bo’ladi:



Bunday yozsak, u holda color 2 o’lchamli massiv bo’ladi:



Agar massivni uzunligidan ortiqcha son elemnt qo’shmoqchi yoki olmoqchi bo’lsak u holda **ArrayIndexOutOfBoundsException** beradi:



Agar **2** o’lchamli massiv yaratsak va unga row larini sonini bersak, lekin shu rowlarda nechtadan element bo’lishini kiritmasak, u holda shu row ni ichidagi elementlarga murojaat qilmoqchi bo’lsak, **NullPointerException** tashlaydi. Pastdagi misolda ham **3** ta rowdan tashkil topishini aytdik massivimizni, lekin shu rowda nechtadan qiymat bo’lishini kiritmadik. Shuning uchun hali kiritlmagan elementni olmoqchi bo’lsak, shu exceptionni beradi. Chunki bu element hali mavjud emas:

