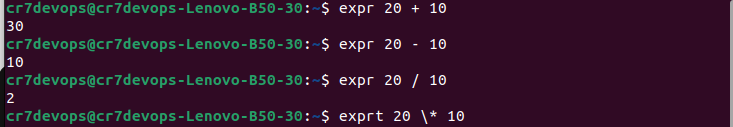
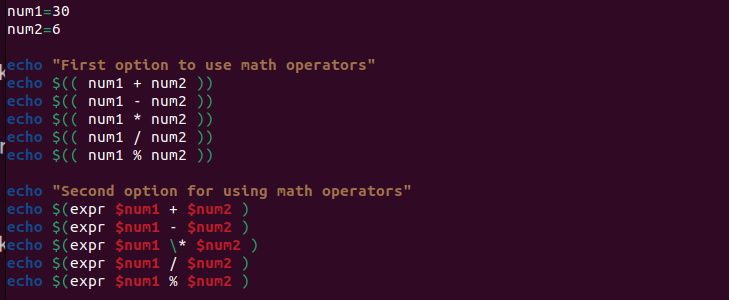
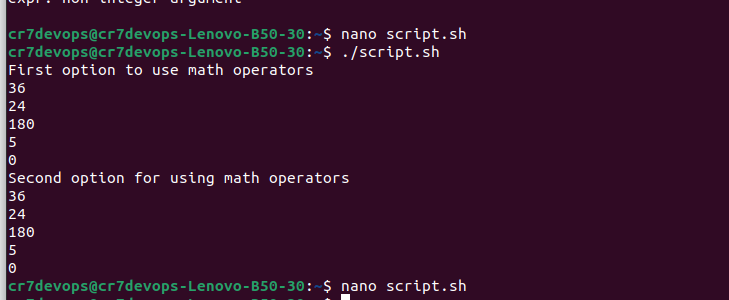
**Mavzu : Basic math, if statements**

Biz linux command line da biron matematik amal bajarmoqci bo’lsak *expr* commandni birinci yozib k amallarni bajarsak bo’ladi.Faqat ko’paytirish(\*) belgsida oldiniga *\(back slash)* qo’yishimiz kerak.

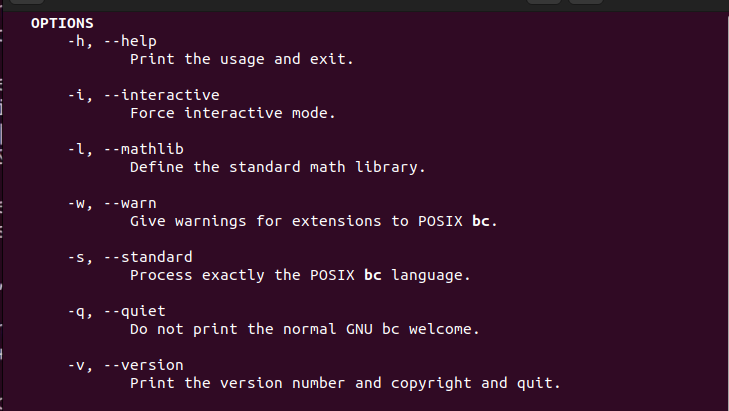






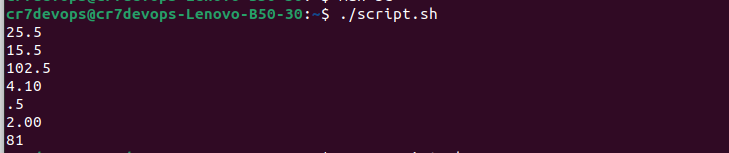
Qoldiqli amallarni bajarsak **bc** commanddan foydalanishimiz kerak.

bc →arbitary precision calculator language, buni o’z kutubxonalari mavjud



example:



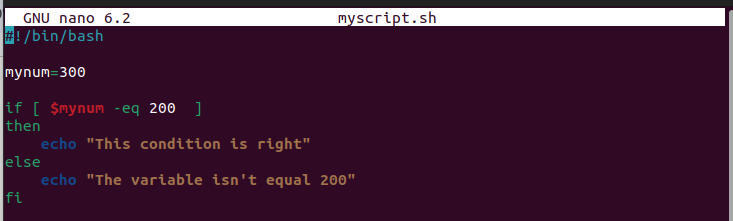


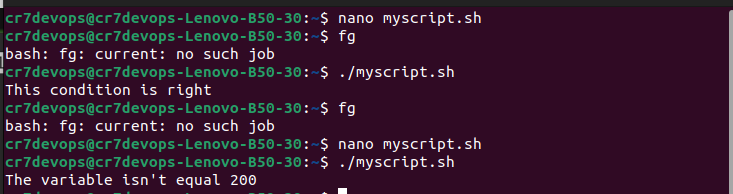
scale →verguldan keyingi nechta qiymat bo’lsa shularni kirgizish uchun ishlatiladi.

sqrt yoki pow →uxhun standart lin dan foydalanamiz.

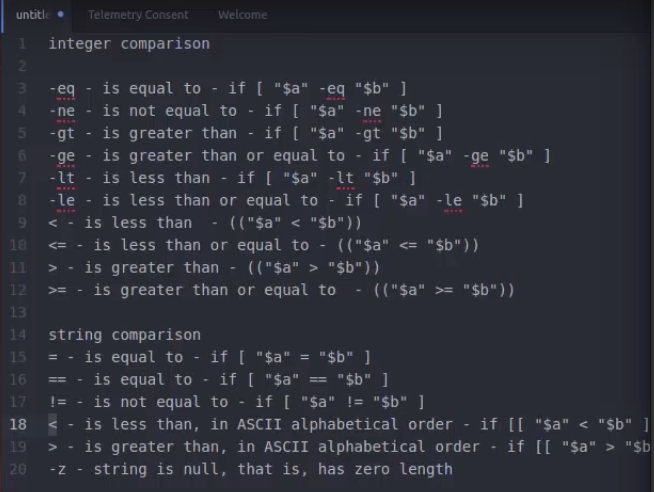
**If statements**

Bu shart amali Ubuntu Linux terminalida boshqacharoq amalga oshiriladi.Bash script fayl ichiga kerakli variables lar yaratib olinadi , if sharti yozilganda qavs dan keyin albatta bitta probel tashlanishi kerak. *“=”* belgisi *-eq* deb yoziladi.





* eq →equal
* ne →not equal
* ! eq →not equal
* gt →greater than
* -f →checking files(regular)
* -d →checking directory
* -e →checking files
* -s →file empty or not



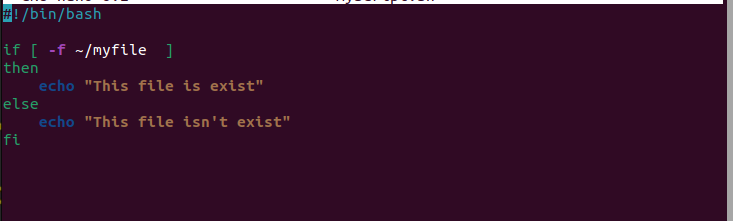
File ga yozish uchun

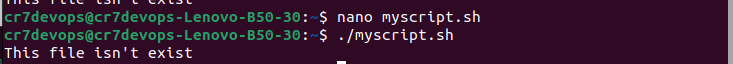
***cat > file\_name***

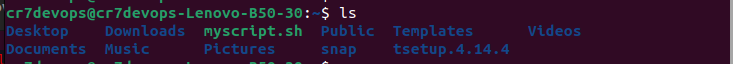
I’ll write something to this file

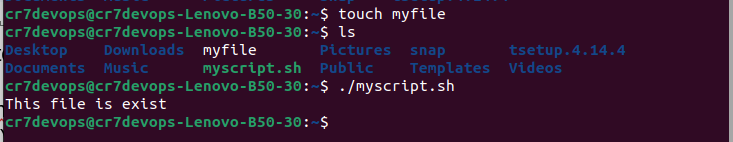
For exiting ***ctrl+d***

File mavjudligini tekshirishimiz uchun quyidagi ishni bajarsak bo’ladi.~(home) file dan myfile degan faylni qidirish jarayonda faylning bor yoqligi tekshirildi.









String comparison

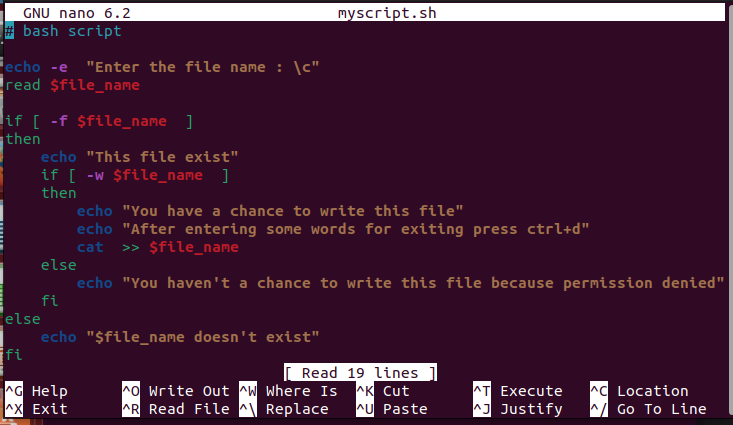


Bunda “b” var $word vardan ASCII kodi boyicha elif tekshirilishda true ekanligi ko’rilyapdi.

*echo -e “...... : \c”* →bitta line da yozish uchun ishlatiladi.

example:

File ni birinchi mavjudlikka tekshiramiz, undan so’ng yozish ruxsati borligiga tekshiramiz shu orada file dan write permission ni olib qo’yamiz va yana qaytarib beramiz.



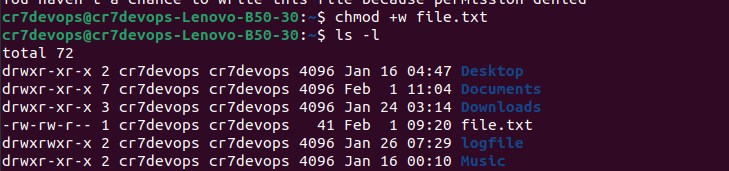
Bu code ni run qilsak filemizda write uchun permission yo’qligini ko’rishimiz mumkin.



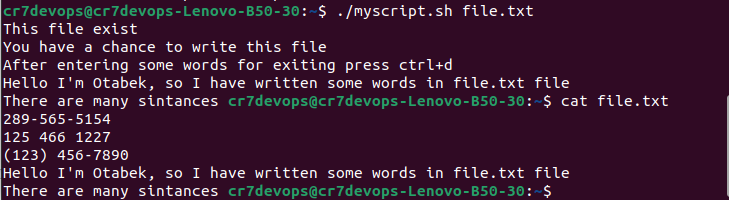
chunki file quyidagicha permissionga ega



endi bu fayga biz write, read uchun permission berib olamiz.



Endi script file mizni qaytadan run qilib olamiz.



Demak ishlarimiz successful bo’ldi.

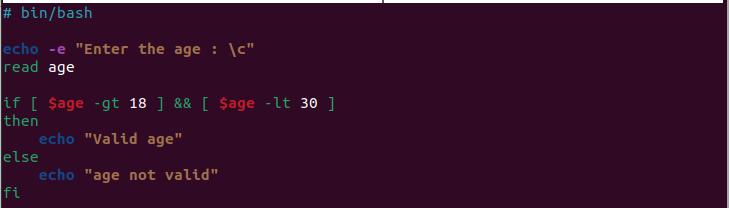
2-example: **AND(-a, &&)** operatori

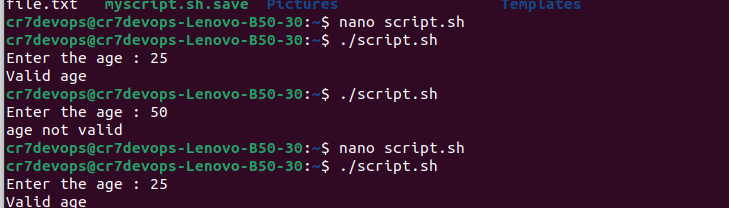
And operatori bir vaqtning o’zida bir nechta shart tekshirilishda foydalaniladi.

Kiritilgan o’zgaruvchini 18 dan katta 30 dan kichik ekanligini 3 xil (sintaksis) yo’l bilan tekshiramiz.

1-yo’l:

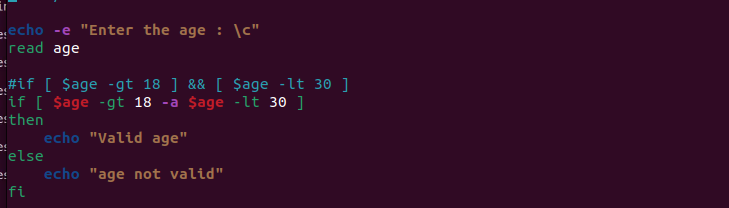
if [$age -gt 18 ] && [ $age -lt 30]

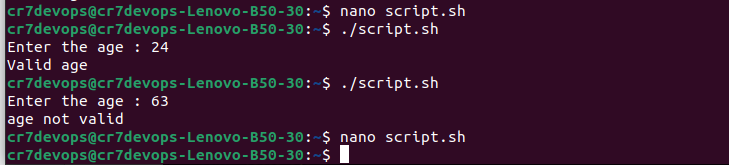




2-yo’l. -a

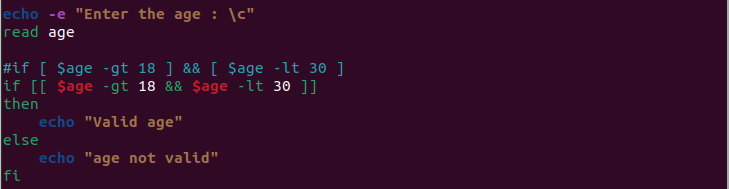
if [$age -gt 18 -a $age -lt 30]

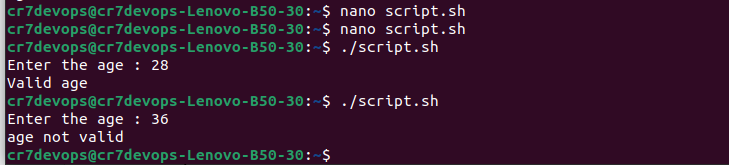




3-yo’l.

if [[ $age -gt 18 && $age -lt 30 ]]



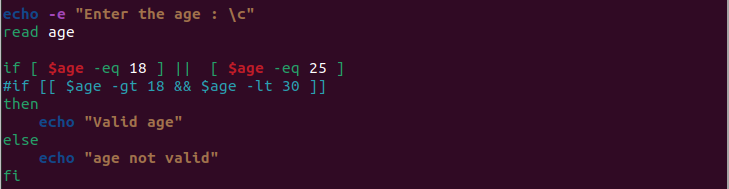


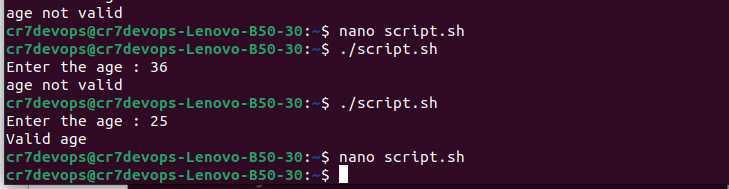
3-example

OR operatori qachonki shartlar ichida eng kamida bittasi bajarilsa bu shart bajariladi.Uni 3 xil ko’rinishini hozir amalda ko’ramiz.

1-yo’l.

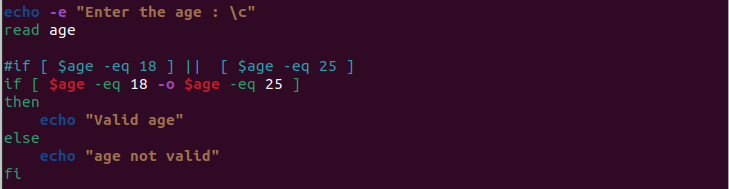
if [ $age -eq 18 ] || [ $age - eq 25 ]

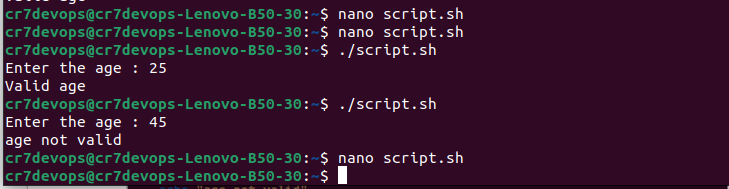




2-yo’l.

if [ $age -eq 18 -o $age -eq 25 ]





3-yo’l.

if [[ $age -eq 18 || $age -eq 25 ]]

