file handiling in c language

as we already know that , when we run a program stored from the hard drive , then the program load in ram and what ever be the variable in the program which allocate some memory and as the program end , the variable destroyed and all the memory were released and the data will be vanished out , pr agar hum yeh chahate hai ki program end hone ek bad bhi yeh data vanished na ho or hame jab dobara yeh program run kare or jan sake ki pahare result kya aye the toh es ke leye yeh jaruri hai ki program end hone se pahale jab wo data kahi permanent store ho jaye jo bhi ek tarah ki file me hoti hai

toh yahi file me data rakhana or file se data nikalana kahalata hai data handling i.e file writing and reading

i.e agar hame jab bar bar ekhi data input karane ki jarurat padati ho toh hum ek hi bar user se data ek file me store karwa lege toh use bar bar enter nhi karana padega

jaise hame ek program ka data jo ki ek variable hai us ki value file me store karani hai toh pahale us file ko ram me load karate hai i.e called as file ko open karana ,

agar hame es file me kuch likhana hai toh file ko access kaise kare , filen koe variable toh hai nhi yeh toh ek hub sari memory ya khub sari bytes ab har ek byte ka kuch address hoga so ek pointer ki help se us file ki pahali bytes ka address store kar lete hai

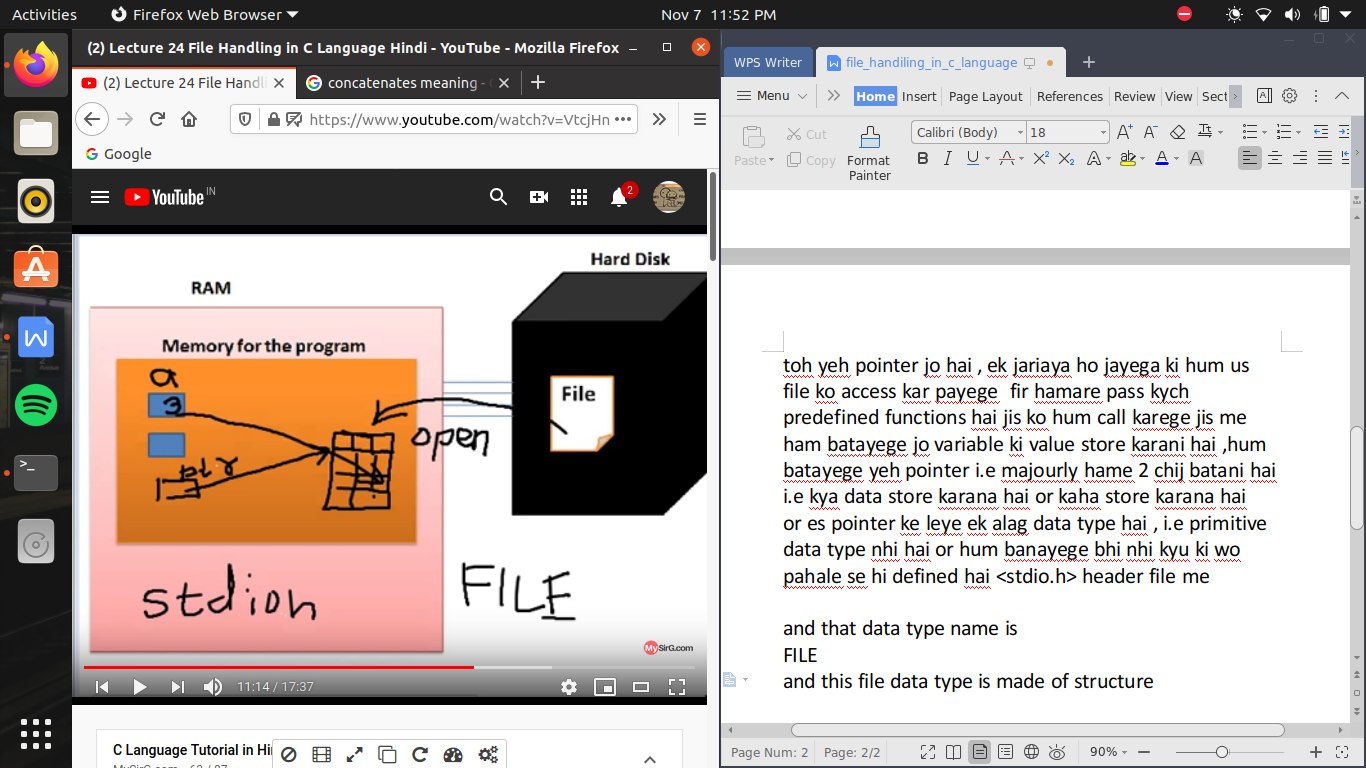
toh yeh pointer jo hai , ek jariaya ho jayega ki hum us file ko access kar payege fir hamare pass kych predefined functions hai jis ko hum call karege jis me ham batayege jo variable ki value store karani hai ,hum batayege yeh pointer i.e majourly hame 2 chij batani hai i.e kya data store karana hai or kaha store karana hai

or es pointer ke leye ek alag data type hai , i.e primitive data type nhi hai or hum banayege bhi nhi kyu ki wo pahale se hi defined hai <stdio.h> header file me

and that data type name is

FILE

and this file data type is made of structure



but yeh file type ke variable ka kuch nam nhi hai agr hum yeh file type ka variable khud na banaye toh ,

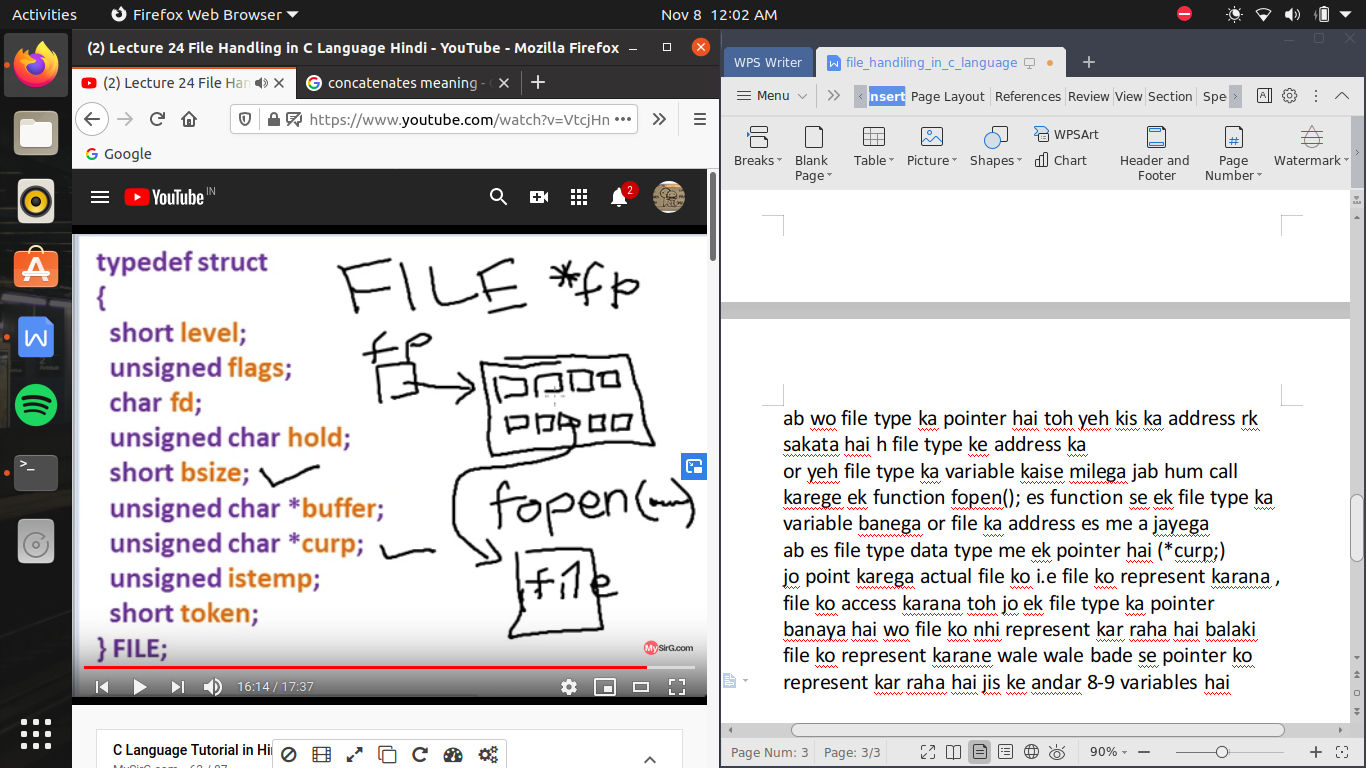
toh hum file type ka banayege ek pointer ,

ab wo file type ka pointer hai toh yeh kis ka address rk sakata hai h file type ke address ka

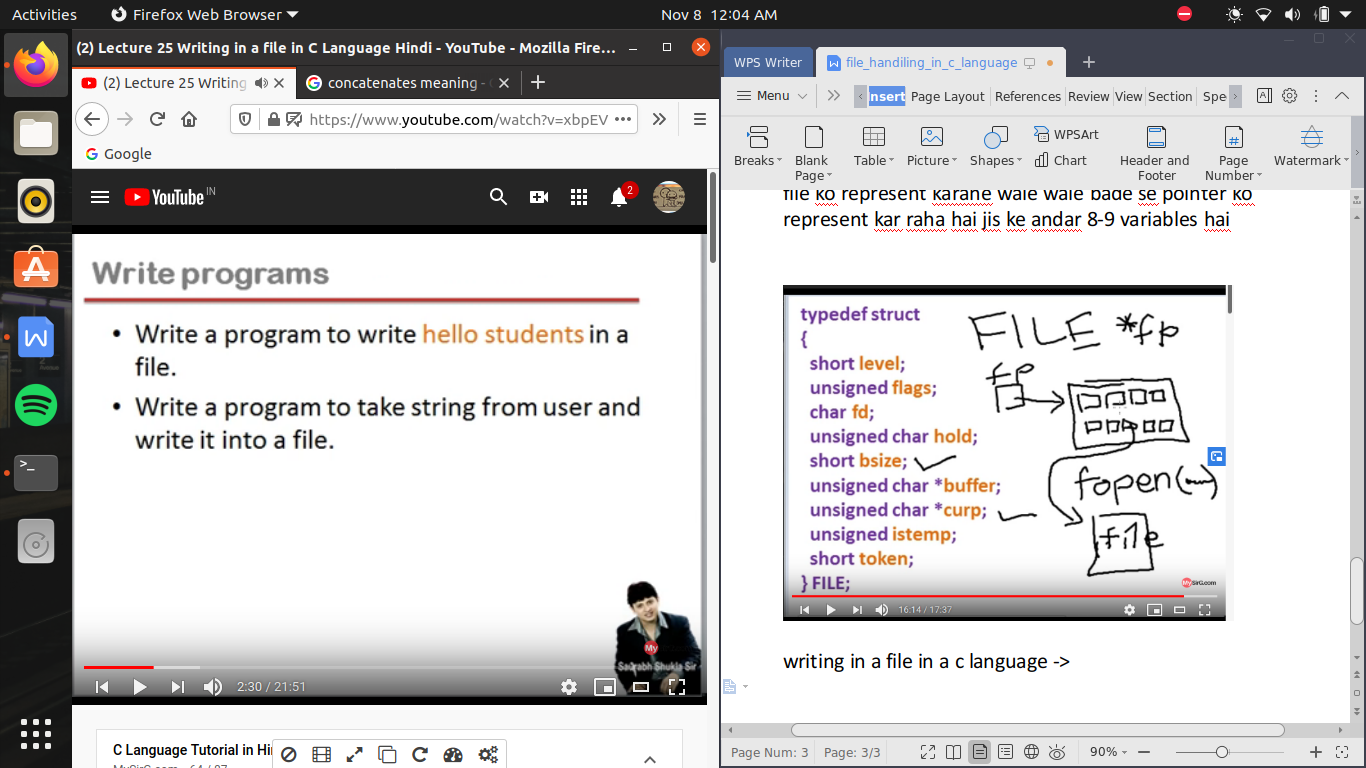
or yeh file type ka variable kaise milega jab hum call karege ek function fopen(); es function se ek file type ka variable banega or file ka address es me a jayega

ab es file type data type me ek pointer hai (\*curp;)

jo point karega actual file ko i.e file ko represent karana , file ko access karana toh jo ek file type ka pointer banaya hai wo file ko nhi represent kar raha hai balaki file ko represent karane wale wale bade se pointer ko represent kar raha hai jis ke andar 8-9 variables hai



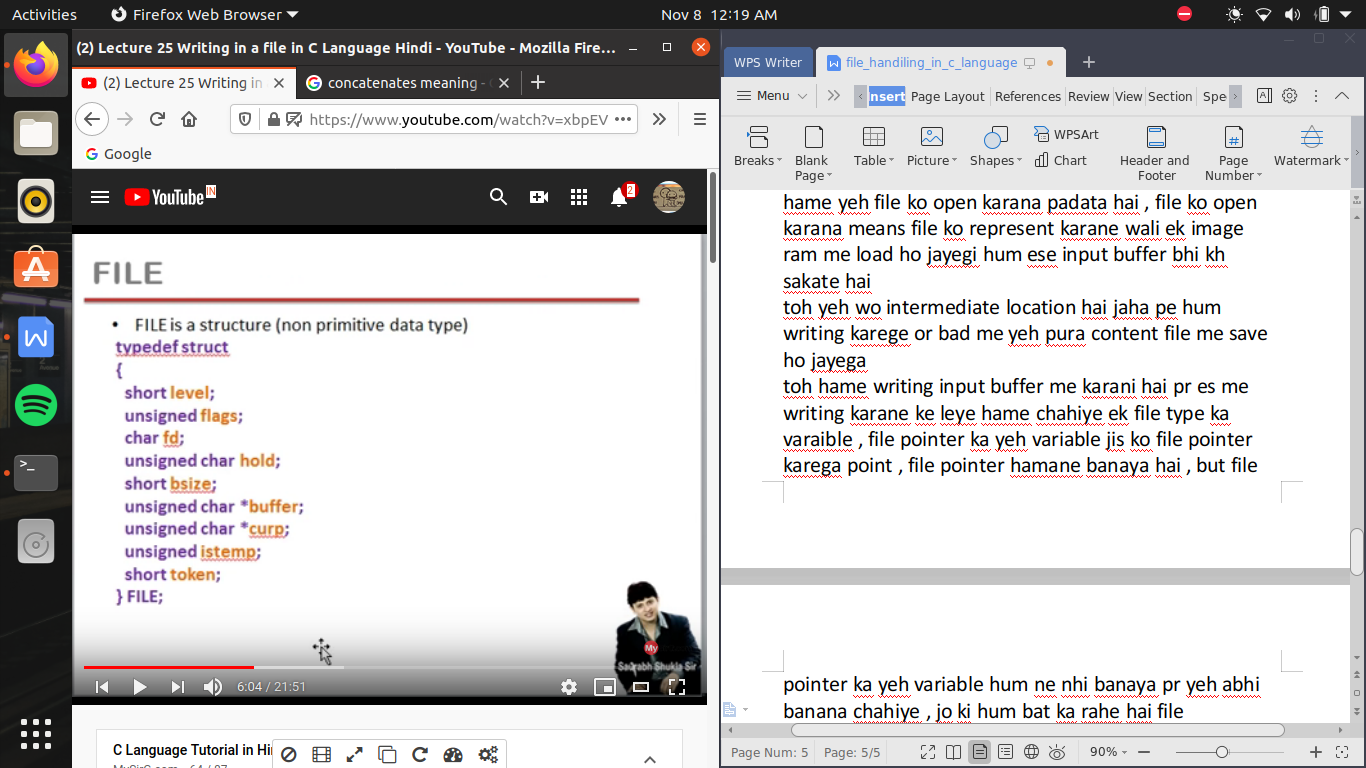
writing in a file in a c language ->



now , jab hum program run karege toh ram me kuch memory lelega ab us memory ek andar hum ek pointer varaible banayege let say \*fp; yeh ek file type ka pointer hai toh ab jarurat hai file type ke ek variable ke bananae ki , let say hard disk me ek file hai f1.txt , hame kya krana hai es file me data write karana hai es ke leye hame yeh file ko open karana padata hai , file ko open karana means file ko represent karane wali ek image ram me load ho jayegi hum ese input buffer bhi kh sakate hai

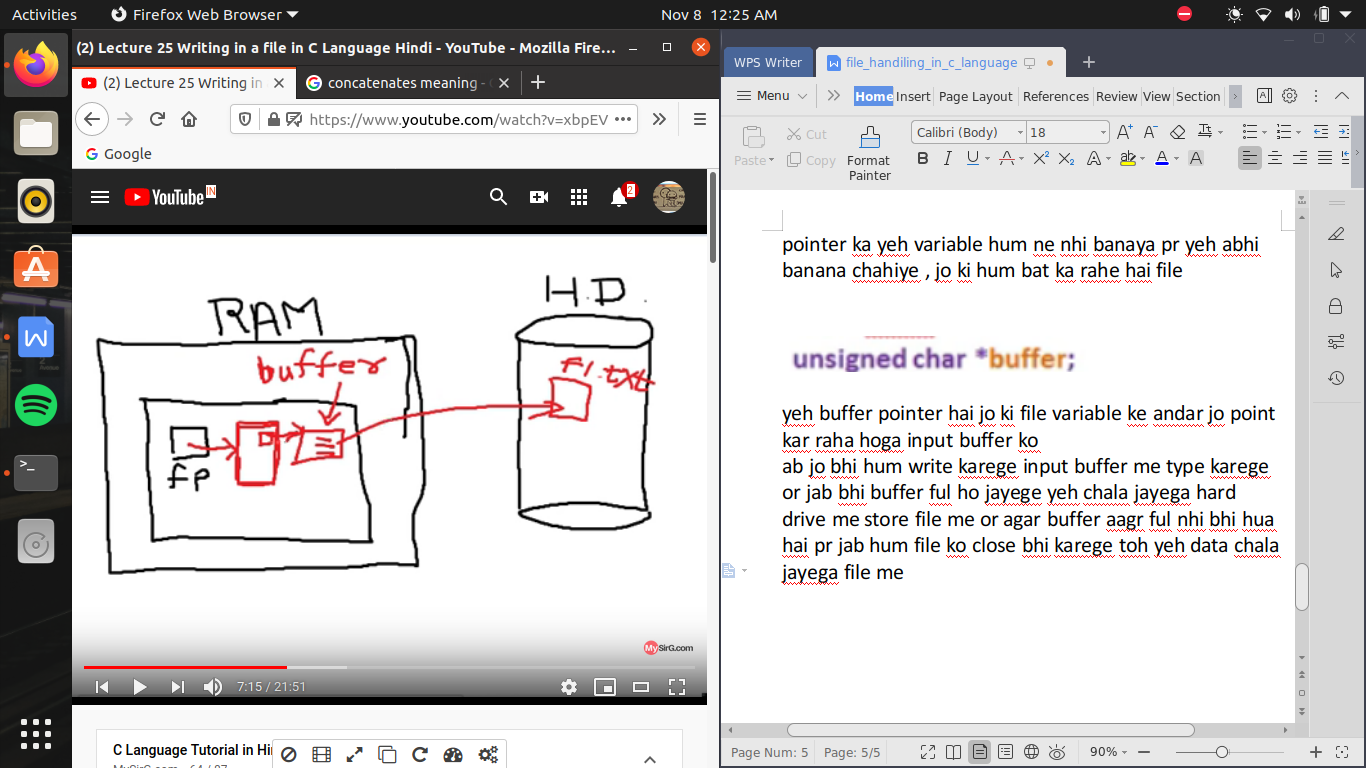
toh yeh wo intermediate location hai jaha pe hum writing karege or bad me yeh pura content file me save ho jayega

toh hame writing input buffer me karani hai pr es me writing karane ke leye hame chahiye ek file type ka varaible , file pointer ka yeh variable jis ko file pointer karega point , file pointer hamane banaya hai , but file pointer ka yeh variable hum ne nhi banaya pr yeh abhi banana chahiye , jo ki hum bat ka rahe hai file



yeh buffer pointer hai jo ki file variable ke andar jo point kar raha hoga input buffer ko

ab jo bhi hum write karege input buffer me type karege or jab bhi buffer ful ho jayege yeh chala jayega hard drive me store file me or agar buffer aagr ful nhi bhi hua hai pr jab hum file ko close bhi karege toh yeh data chala jayega file me



pr hame etana sab kuch headache lene ki jarurat nhi hai ham bs yeh karana hai ek fopen(); nam ka function banana hai or es file me pahale argument ke nam pr file nam pass dege yani jis file ko hum open karana chahte hai or dusare me file ka opening mode pass kr dijiye

opening mode age padege

(sare file openning mode ke string ke rup me represnt hota hai )

jaisa yaha “w” likha hai yeh right mode kahalata hai

/\* file\_handling \*/

#include<stdio.h>

#include<string.h> //to consider strlen(); command

#include<stdlib.h> //to consider exit(); command

int main()

{

int i;

char s[]="hello students";

FILE \*fp;

fp=fopen("f1.txt","w");

//abhi fopen me na keval file ko represnt krane wale input buffer ko banaya balaki file type ke variable ko banaya jis ke andar buffer pointer ne input buffer ko point bhi kiya yani sara headache wala kam kr diya fopen() function ne or badale me kya return kiya address kis ka , yeh file type ke address return kiya fopen function ne toh yeh ab return kr raha hai toh es address ko kisi pointer me rakahan padega or as file type ka address pass kar raha hai tohn ese file type pointer me hi rakhana padeaga

//now file pointer ke through sab kuch access ho jayega

//or agar f1.txt nam ki file exit nhi karati hai toh nayi file create ho jayegi , or agar exist karati hai toh wohi file open ho jayegi

//lekin us ke andar ka pura content pahale erase ho jayega or yeh kyu hota hai us ke bare me file opening mode me bataya jayega

//so hame check karana hai ki file open ho payi hai ya nhi , toh us ke leye agar file open nhi ho payi hai kisi bhi reason se toh fp<pointer>

//me a gaya hoga null

//so agar hum check kar le pointer me null hai ya nhi toh pata chl jayega ki file open hue hai ya nhi

if(fp==NULL) // NULL should be written in capital letters only

{

printf("null =\n");

exit(1);

}

for(i=0;i<strlen(s);i++)

fputc(s[i],fp); //now yeh fputc ka headache hai ki kaise pointer ko handle kr ke kaise input buffer me writing perform karega

//hame bs yeh karana hai ki bs fputc ko wo charecter dena hai jo buffer me write karwana hai

//ab loop ki help se har ek charecter pass hoga or input buffer me write ho jayega

fclose(fp);

//fclose ke andar bhi pointer pass karwana padega

//as we know earlier jiase hi input buffer full ho jata hai , wo data file me move kr deta hai or fir se khali ho kr dobara write ke leye teyar ho jata hain

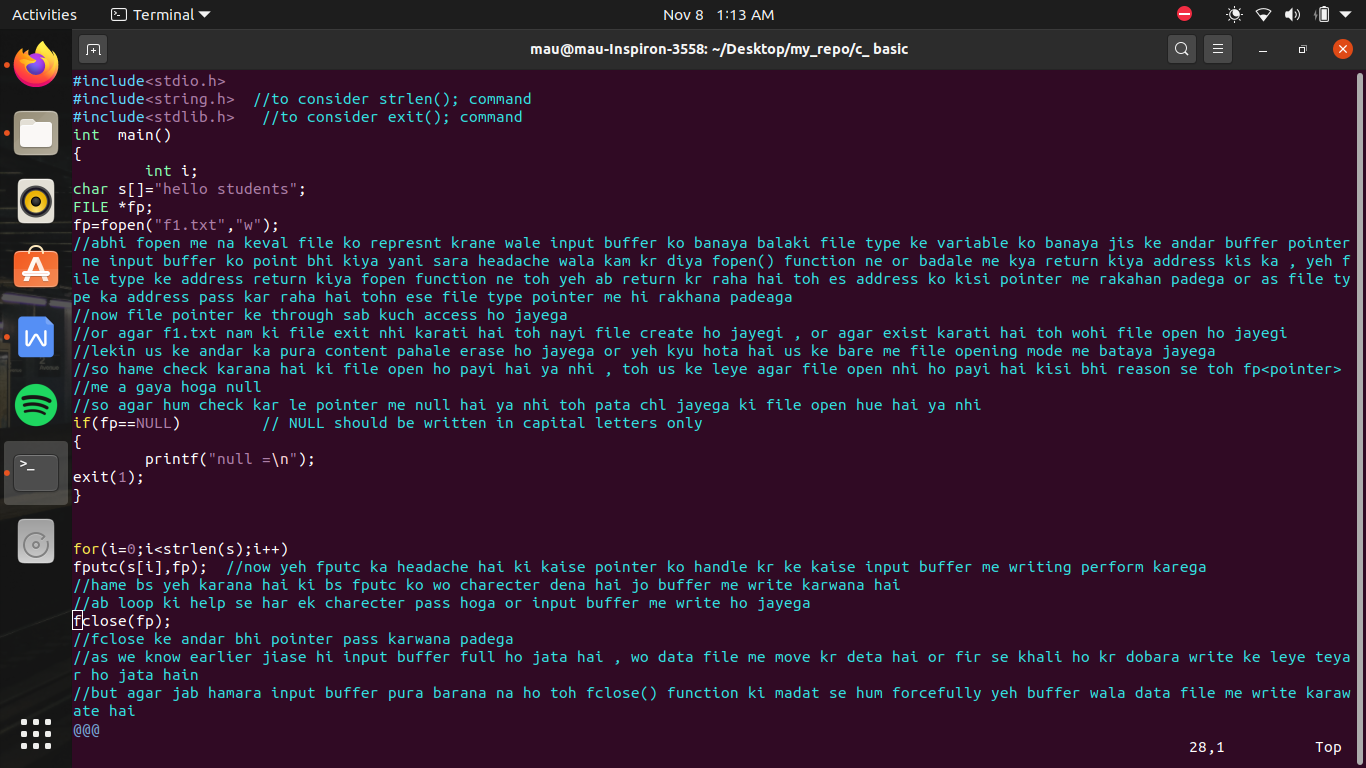
//but agar jab hamara input buffer pura barana na ho toh fclose() function ki madat se hum forcefully yeh buffer wala data file me write karawate hai

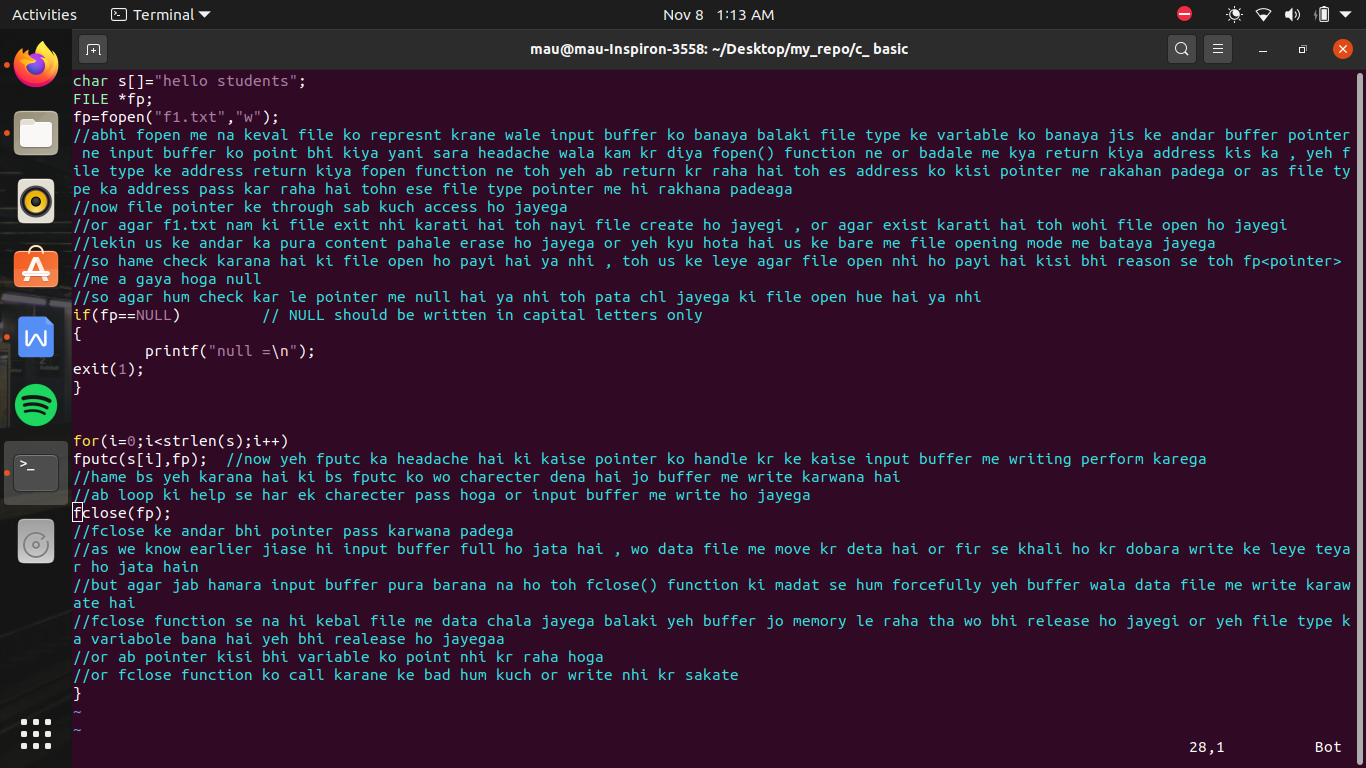
//fclose function se na hi kebal file me data chala jayega balaki yeh buffer jo memory le raha tha wo bhi release ho jayegi or yeh file type ka variabole bana hai yeh bhi realease ho jayegaa

//or ab pointer kisi bhi variable ko point nhi kr raha hoga

//or fclose function ko call karane ke bad hum kuch or write nhi kr sakate

}





now if we have to enter and store data which is entered by user

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

int main()

{

FILE \*fp;

char s[100];

int i;

fp=fopen ("f2.txt","w");

if (fp==NULL)

{

printf("file is not open \n");

exit(1);

}

printf("enter the string \n");

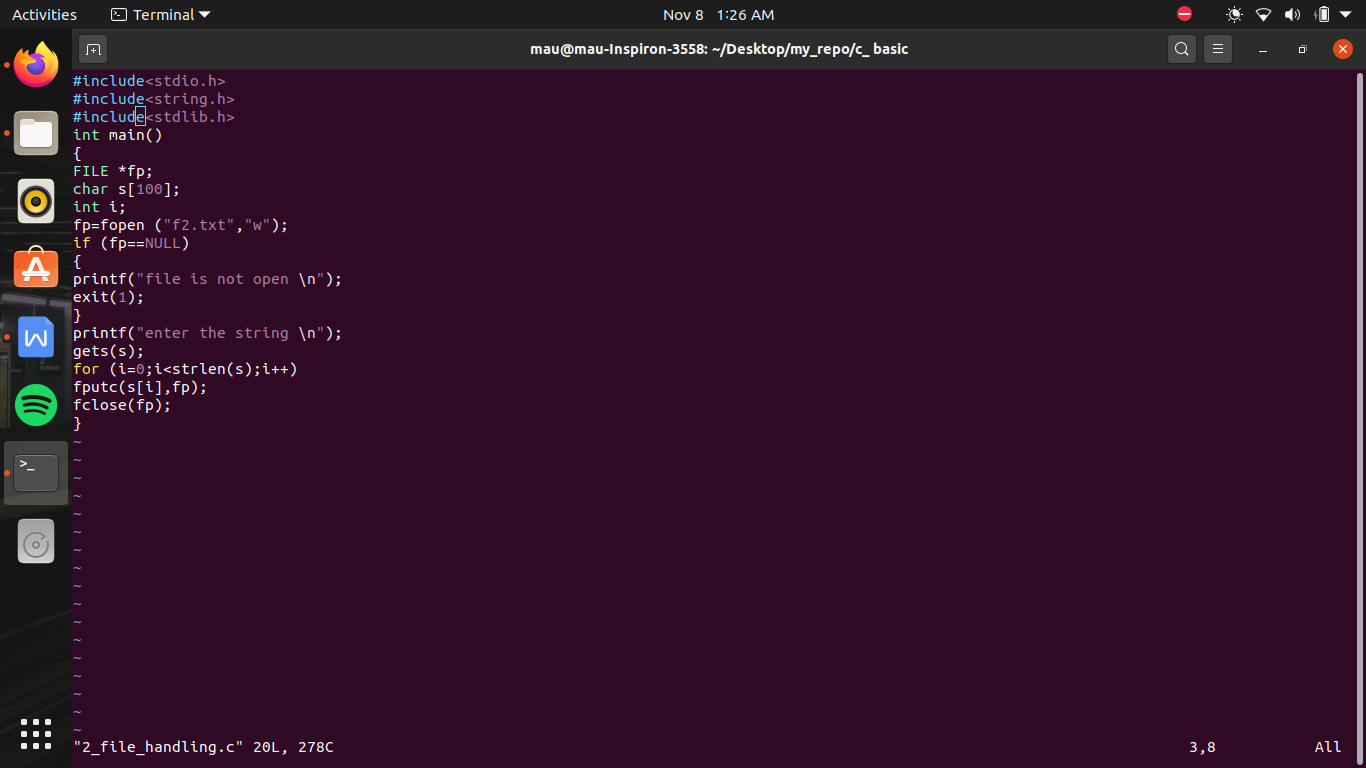
gets(s);

for (i=0;i<strlen(s);i++)

fputc(s[i],fp);

fclose(fp);

}

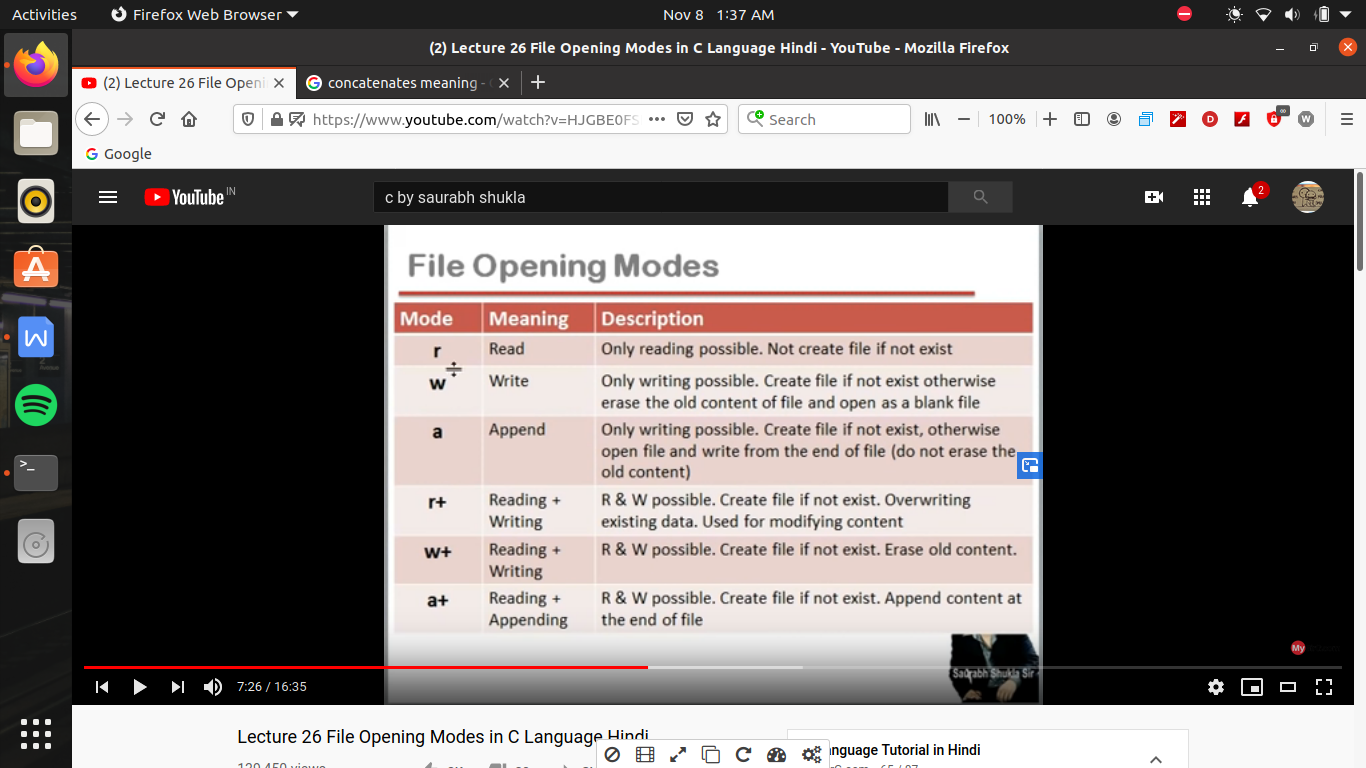


file opening mode in c language ->]

fopen(“<filename>”,”mode”);

modes tell that hum ne file ko kis intention se open kiya hai

we have total 6 possiblilities



“r” -> read only pre-excisted file, nayi file nhi banegi

“r+” -> main intension reading and also write can be done on pre-existing file only, nayi file nahi banegi

or agar “r” or “r+” ko chod de toh baki sab me agar file exsist nhi karati hai toh nayi file ban jayegi

mostly use when we have to update or modify old content

“w” -> only writing not reading and if file exist earlier toh file ka data pura erase ho jayega or file ek emply file ke roop me open hogi taki jo bhi hum write kare naye sire se write kare

“w+” ->both reading and writing but delete old content

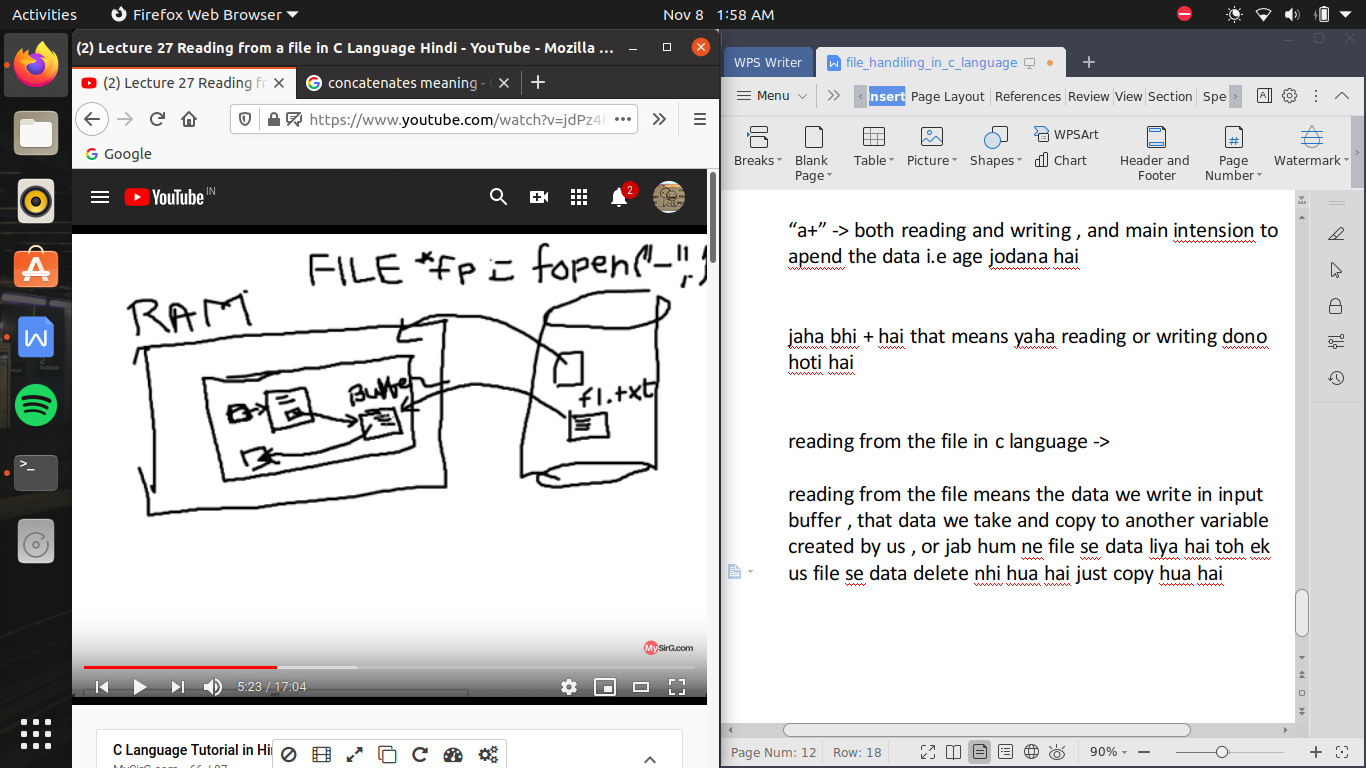
“a” -> only writing , not reading , but if file exsisted toh purane content ke age se write karega , pahale wala kuch delete nhi hoga

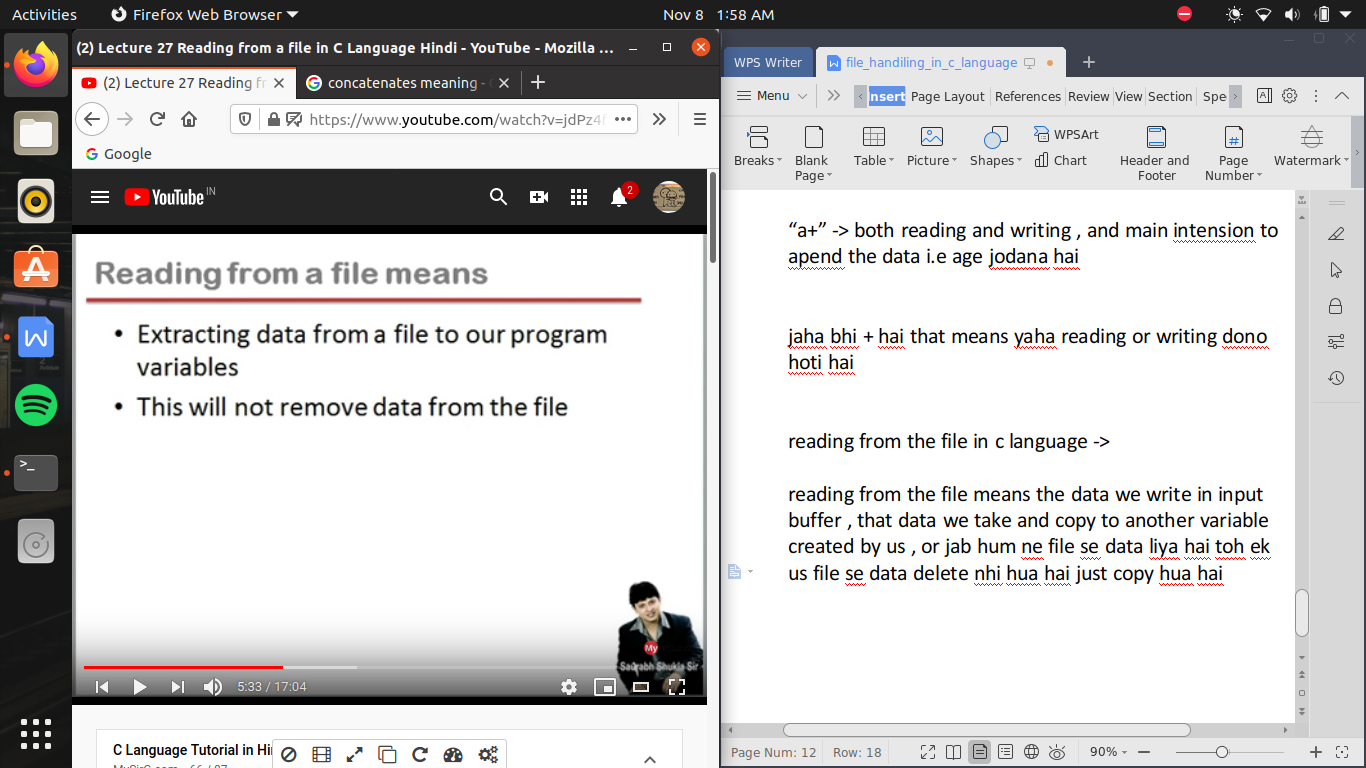
“a+” -> both reading and writing , and main intension to apend the data i.e age jodana hai

jaha bhi + hai that means yaha reading or writing dono hoti hai

reading from the file in c language ->

reading from the file means the data we write in input buffer , that data we take and copy to another variable created by us , or jab hum ne file se data liya hai toh ek us file se data delete nhi hua hai just copy hua hai

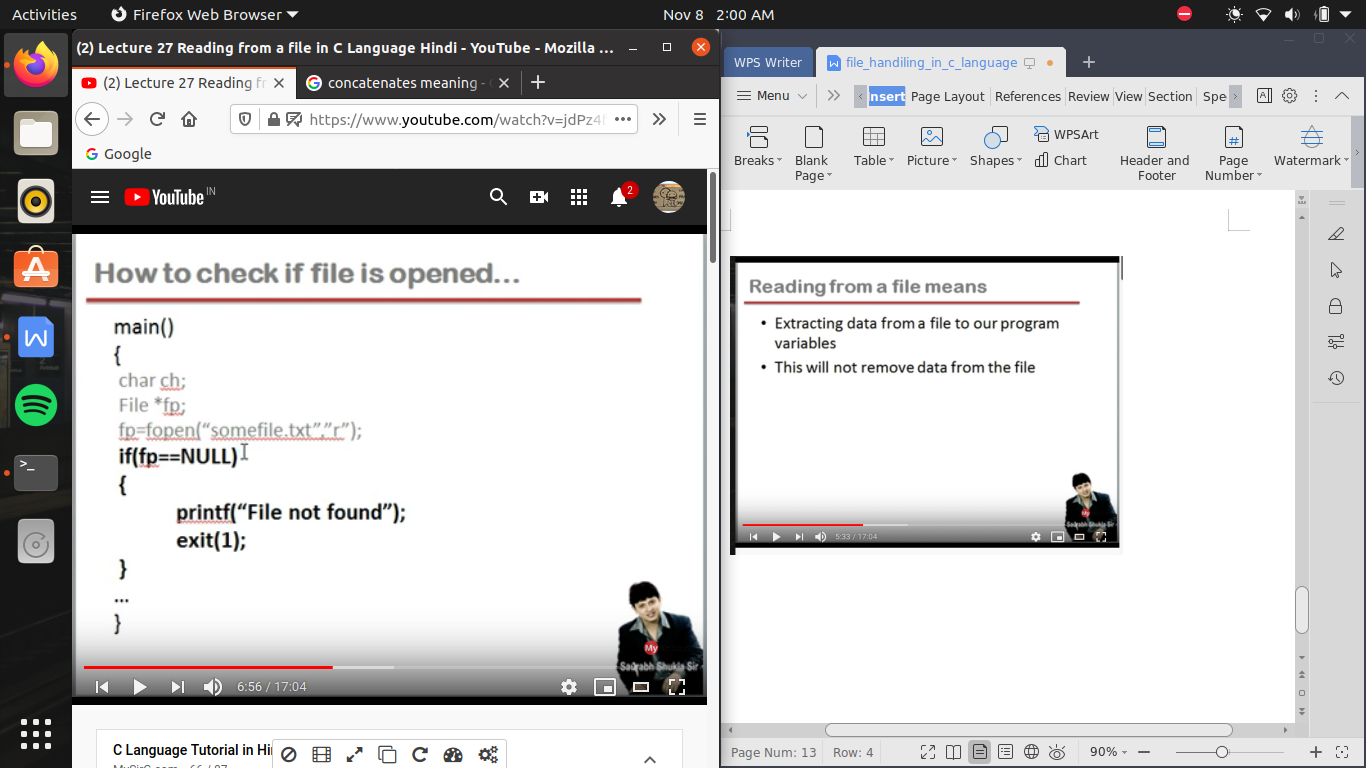




now here we created a variable char for the read of tha data form the file so , so that data will be copy in char data type and also one by one

so it is important to create a loop

how to check if file is opened



ex- write a program to read content from a file and display on the screen

/\* in this program we try to read from the pre-exsisting file \*/

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

int main()

{

char s;

FILE \*fp;

fp=fopen("f2.txt","r");

if (fp==NULL)

{

printf("file not found \n");

exit(1);

}

s=fgetc(fp);

//here we use the function fgetc(); to read the file, yeh function return karata hai ek character jo ki file ki pahali byte me raklha hua hai

//fgetc() ko yeh batane ke leye ki kis file se character pass karana hai toh use ke leye hum es me ke argument pass kr dege us pointer ka jonki input buffer ke pahale byte ho point kar raha hai

//ab ek ek chrater ko pass karege toh hame eseme loop lagana padega pr loop kaha tak lagaye yeh kaise pata chalega

//es le leye ek predefined function hai feof() i.e end of file, yeh function return karata hai true ya false , i.e 1 or 0 mtlb agar 1 return kr raha hai yani file ka end a chuka hai

//ab agr file ka end a gya hai toh ese rukanana chahiye toh hum not laga dete hai

while(!feof(fp))

{ printf("%c",s);

s=fgetc(fp);

}

fclose(fp);

}

