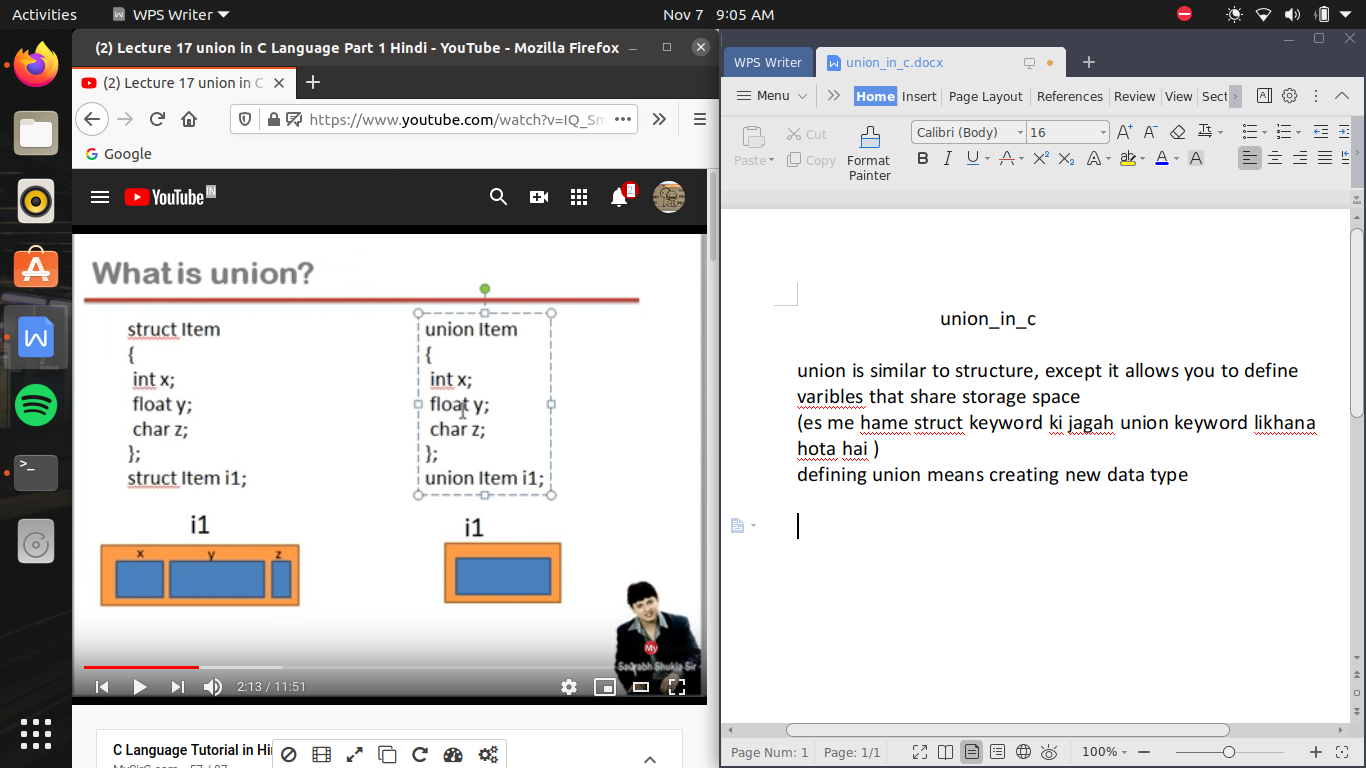
union\_in\_c

union is similar to structure, except it allows you to define varibles that share storage space

(es me hame struct keyword ki jagah union keyword likhana hota hai )

defining union means creating new data type



struct me , ek data type banega jis me x,y,z yani 2,4,1 bytes alag alag memory allocate hogi , yani yeh pura data type 7 bytes memory lega

but union me jo variable sab se jayada memory lega wo memory us data type ko allocate ho jati hai

union memory effecient hai pr union har jagah kam nhi ayega agar hum ne 3 variable banaye or hame tino sath me jarurat hai toh hum structure hi banayege but fir ek bar me ak ki hi jarurat padegi tab union kam a sakata hai

#include<stdio.h>

union item

{

int x; float y; char z;

};

void main()

{

union item b;

b.x=89;

printf("%d\n",b.x); //here we do is as x value is print and function ends then the memory allocated to next function

//so that is why we use one by one of the function because union call max space as 4 bytes and if we printf all the variable simultaneouslt then the result yield wrong value because data type does't carry enougth memory to allocate all of the variable siomultaneously

b.y=23.4; //yaha x ki value overwrite ho gayi hai y ki value se toh agar if we want to print x value after that we can't be able to that because x variable hi overwrite ho gaya hai y variable se

printf("%f\n",b.y);

b.z='a';

printf("%c\n",b.z);

}

/\*

union ka jayada tar use low level programming me hoti hai jab ki structure ka use high level programming me kiya jata hai

\*/

