**USER DEFINED DATA TYPES**

1. Structures
2. Unions
3. Enums
4. **Structures**

Struct tagName

{

Members of structures

**};**

**…….**

**Struct square**

**{**

**Int len;**

**Int breadth;**

**};**

Struct chair

{

Int no legs;

Char make[20];

Char material[20];

Char clr[20];

Float price;

Char DOM[20];

Char placeManu[20];

}

**Int** a1,a2, ;

Struct Square s1,s2,s3,sq[10];

Struct Square

{

Int len;

Int breadth;

}s4,s5,s6,s[10];

**Typede**f struct Square SQR; /sqr --- it is like defining or jst giving name to the struct to reduce the call of code with long name

SQR s7,s8,s9

Typedef struct square

{

Int len; int breadth; }SQR1

From the above SQR1 s1 is var ;

We can access the struct variables by

1. . => static variable == nameVar.memberName
2. -> => ptr var 🡺nameVar->memberName

**SQR! \*ptr;**

**…………………..in structure we cant declare c…………………………..**

**STRUCTURE PADDING :** all char define at begin or numeric at end -<🡪numeric to characters

**Srting tockenize**

**Code on toknizer**

#include <stdio.h>

#include <string.h>

int main()

{

char Lines[] = "101|Amit Kumar|M|8888|10000";

char Names[10][20];

int row=0,col=0,i;

char \*ptr=NULL;

int flag = 0;

ptr = strtok(Lines,"|");

do{

strcpy(Names[row],ptr);

// puts(Names[row]);

ptr = strtok(NULL,"|");

row++;

}while(ptr != NULL);

for(i=0;i<row;i++)

printf("\n%s",Names[i]);

printf("\n\n");

return 0;

}

…………………………

**UNION**S

**File Handling : file pointers**

**The**re are two different type of files

1. Sequential file
2. Random access files

It use spl data structure **FILE**

FILE \*fp

FILE \*fopen(char\*name,char\*mode);

Fp = fopen(“file name ,”mode”)

……fopen will return the address…….

R – read

R+ -- read & write

W – write

W+ -- if file not present it creats the fiel with the given and we can write If the file is existing then it truncates the content the new page wil open in the same file for to write

a—append – open for appending (if th file not existimg it create new file and u can use for read and writr

file pointer says where the pointer is located

--- ftell -🡪 say the position where the pointer or cursorwill be

Fseek 🡪moving of the bytes ,seek \_set begin of the file(move 10 byte forward),,

Fseek🡪seek\_END – is alo used for to get size of the data present in filer’

Seek\_cur—if we want to move backward we (-)off set

Fseek(fp,-13,SEEK\_CUR)

Fseek(fp,13,SEEK\_SET)

(Void) fseek(stream,0L,SEEK\_SET)

Seek seek\_cur,seek\_end ,seek\_data--- int(only)(0,1,2,3)

Fread , fwrite 🡪

Fread(void \*ptr,size\_t size\_t size, size\_t nmemb);

Fwrite(const void\*ptr,size\_t,size\_t nmemb,file\*stream)

Return value will if there is no contents for to read and to write