

# CS & IT ENGINEERING

Programming in C

Structure and Union  
Lec- 01

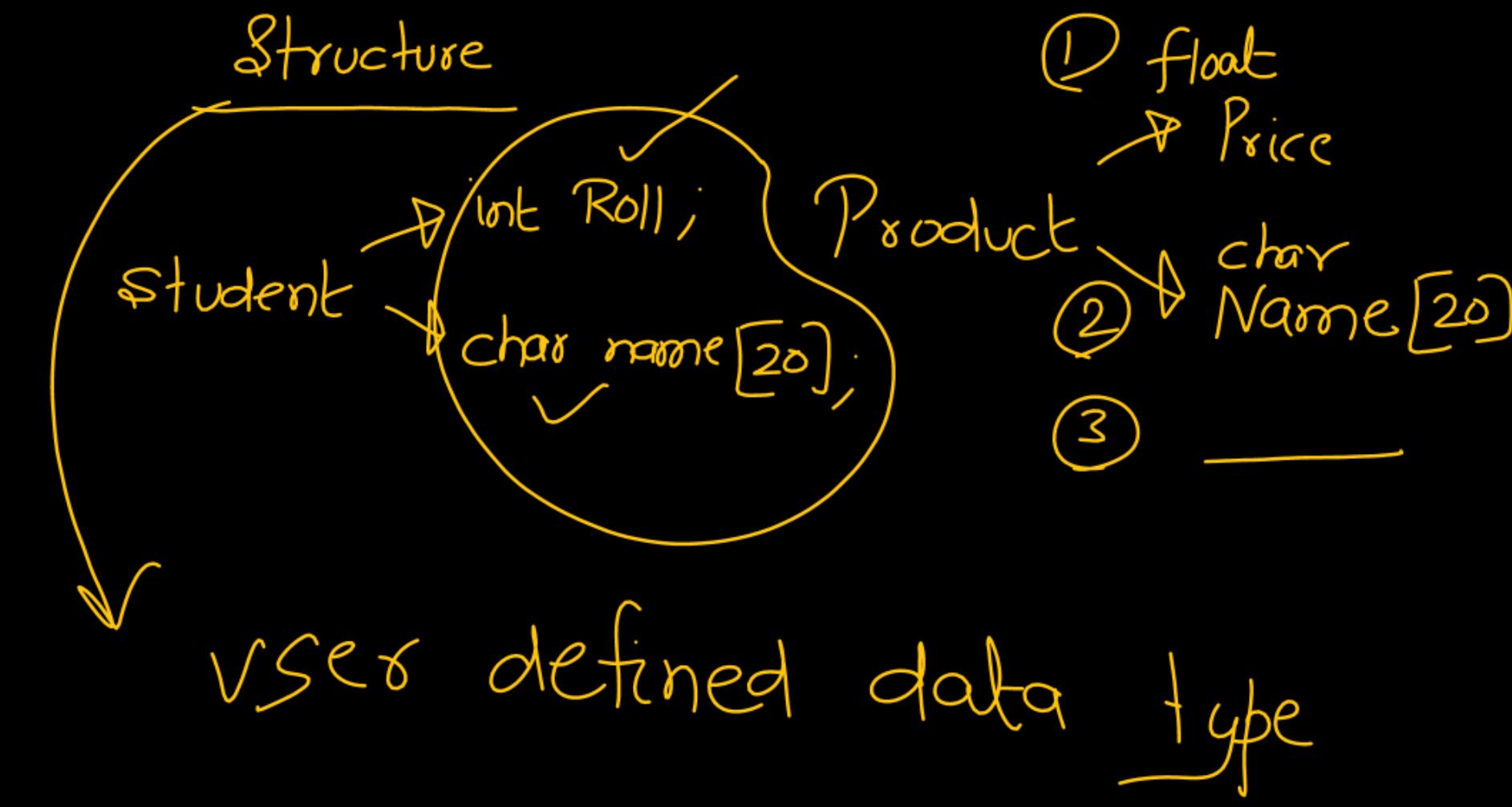


By- Pankaj Sharma sir

## TOPICS TO BE COVERED



Structure and Union-I



↳ struct is the keyword used to create user defined data type.

```
struct student {  
    int Roll;  
    char name [20];  
};
```

Template  
blueprint  
primitive  
int, char, float

⇒ No memory is allocated  
⇒

```
struct student {  
    int Roll;  
    char name[20];  
};
```

```
void main() {
```

```
    int ;
```

```
    struct student ;
```

```
}
```

} memory allocate X

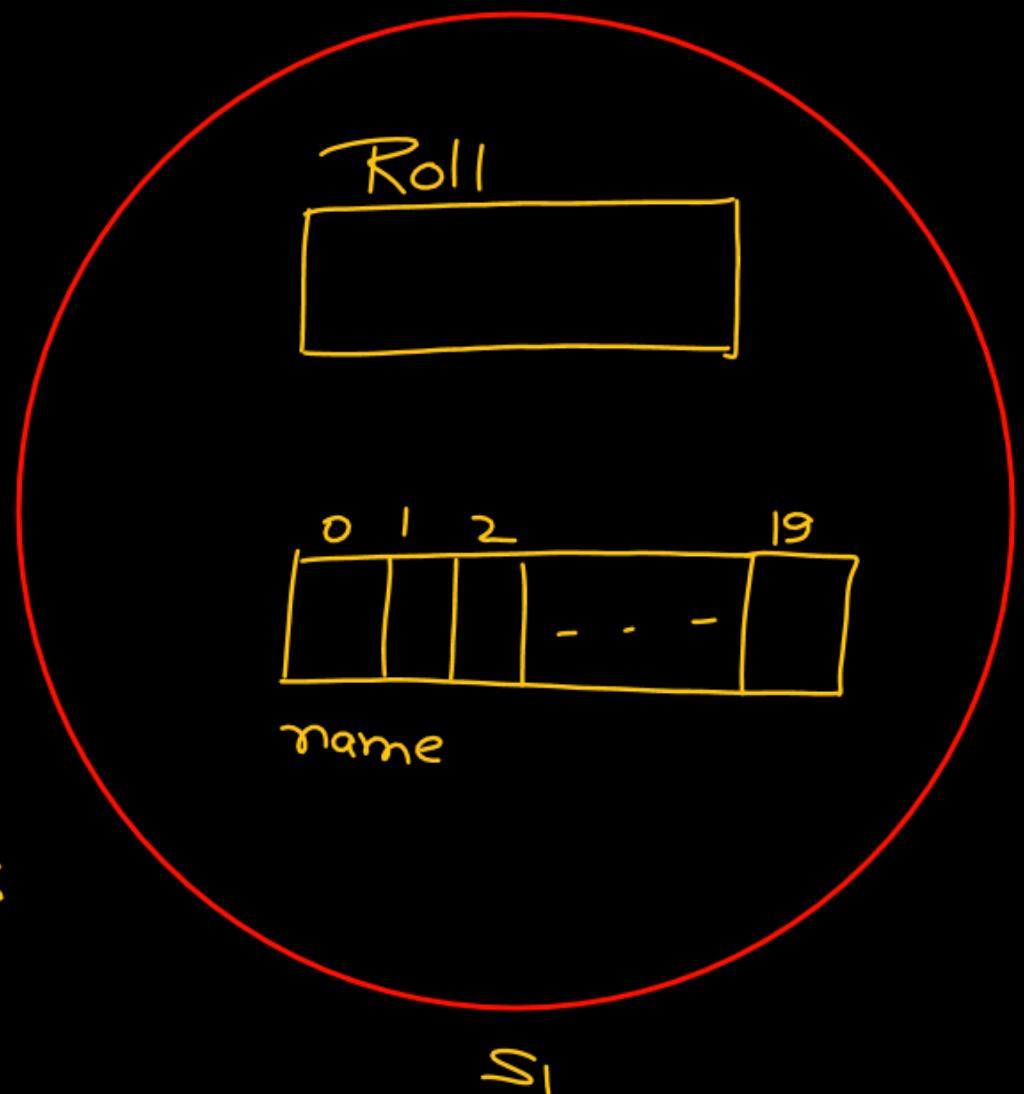
data type ch  
variable

```
struct student {  
    int Roll;  
    char name[20];  
};
```

```
void main(){  
    struct student s1;
```

} } Global

s1 is a group  
with 2 members  
Roll  
name



```
struct student {  
    int Roll;  
    char name[20];  
};
```

```
void main() {
```

```
    struct student s1, s2;
```

```
    s1.Roll = 10; ✓
```

~~```
s2.Roll = 20;
```~~

```
s1.name = "Pankaj";
```

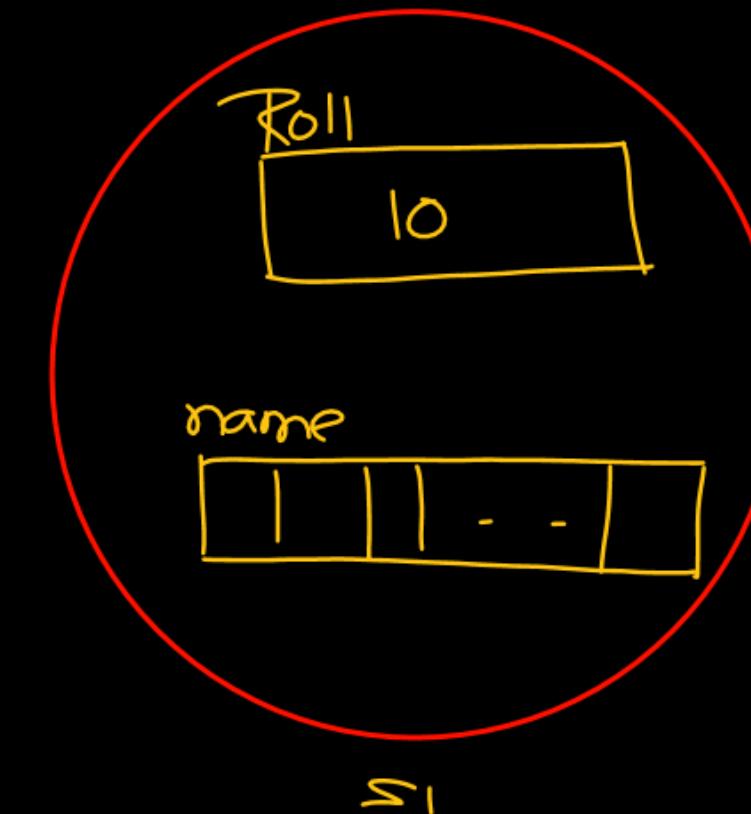
invalid

solution

→

```
strcpy(s1.name, "Pankaj");
```

```
strcpy(s2.name, "Neeraj");  
}
```



$\Rightarrow$   
 $s1.Roll$

```
struct stu {  
    int Roll = 10;  
    char name [20] = "Pankaj";  
};  
void main() {  
    struct stu s1, s2;  
}
```

No  
memory  
is  
allocated

X Invalid

① struct stu{  
    int Roll;  
    char name[20];  
};

void main(){  
    struct stu s1,s2; ✓  
    =  
}

Void f(){  
    struct stu s3,s4;  
    =  
}

Tag of structure

② struct {  
    int Roll;  
    char name[20];  
}s1,s2,s3;

void main(){  
    =

③ typedef struct stu{  
    int Roll;  
    char name[20];  
}Pankaj;

void main(){  
    Pankaj s1,s2;  
    =  
}

```
struct stu{  
    int Roll;  
    char name[20];  
};
```

int a[4] = {10}; ✓

```
void main(){
```

```
struct stu s = { 10, "Pankaj" }; ✓
```

```
struct stu s = { "Pankaj" }; // s.Roll = "Pankaj"; ✗
```

```
struct stu s2 = { 10 }; ✓
```

s2.Roll = 10;  
s2.name = ?

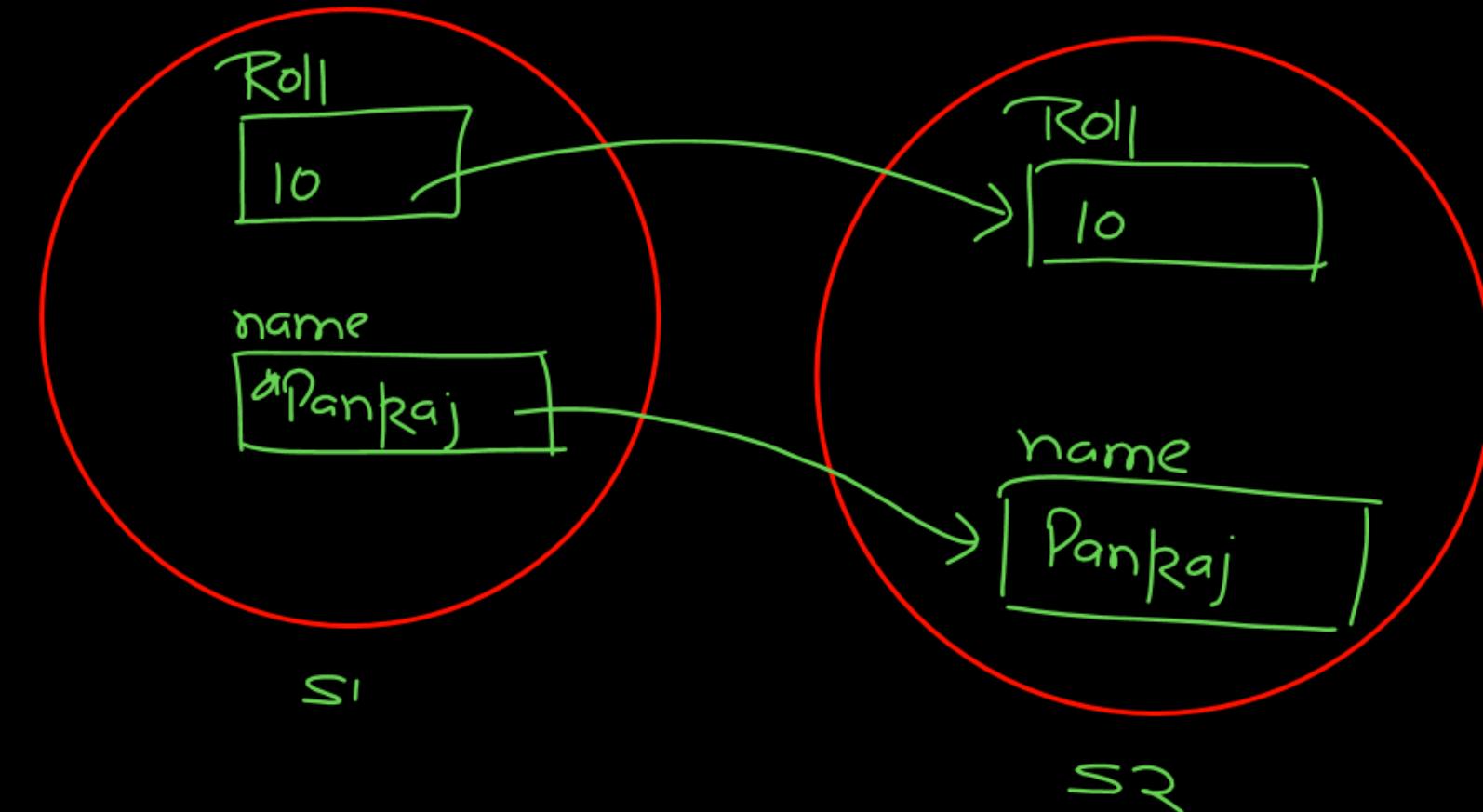
```
struct stu{  
    int Roll;  
    char name[20];  
};
```

```
void main(){
```

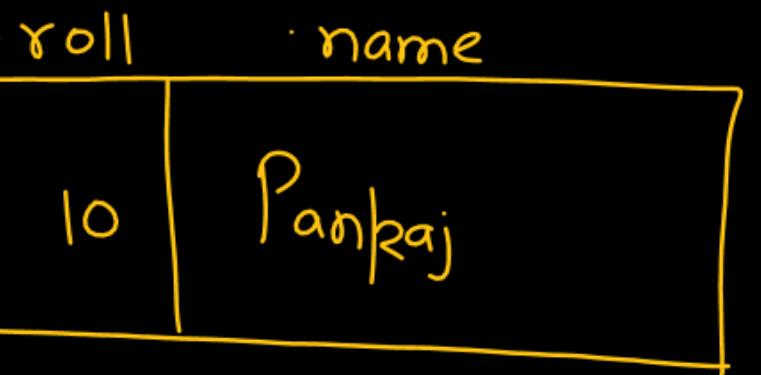
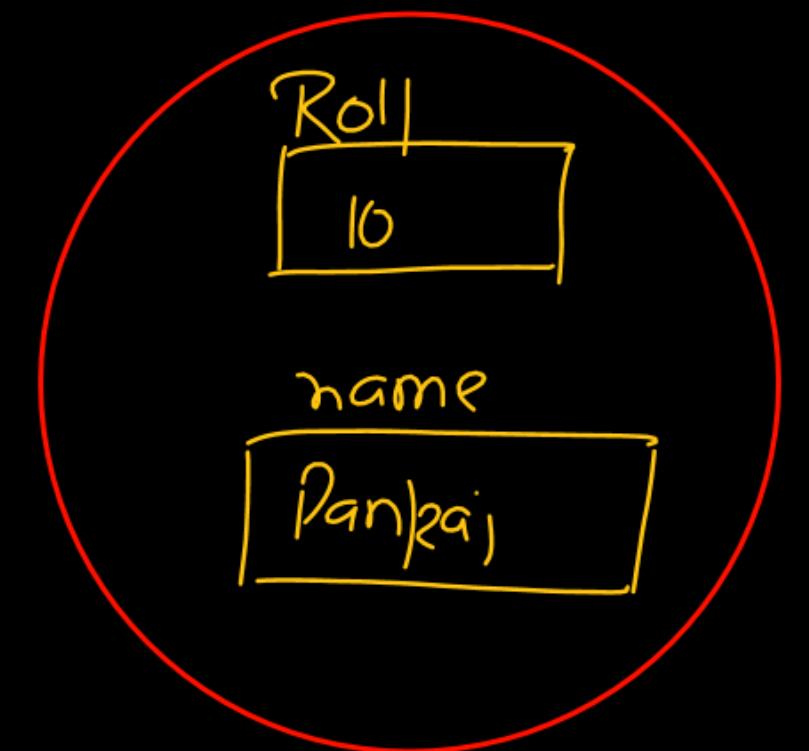
```
    struct stu s1 = {10, "Pankaj"};  
    struct stu s2 = s1; ✓
```

```
    struct stu s3 = {"Pankaj", 10}; ✗ Invalid
```

```
}
```



✗ Invalid



Padding / Alignment restriction

```

struct stu{
    int roll;
    char name[20];
};

void display( struct stu );
void main()
{
    struct stu s;
    printf("Enter Roll no");
    scanf("./d", s.roll); 10
    printf("Enter name");
    scanf("./s", s.name); Pankaj
    display(s);
}

```



```

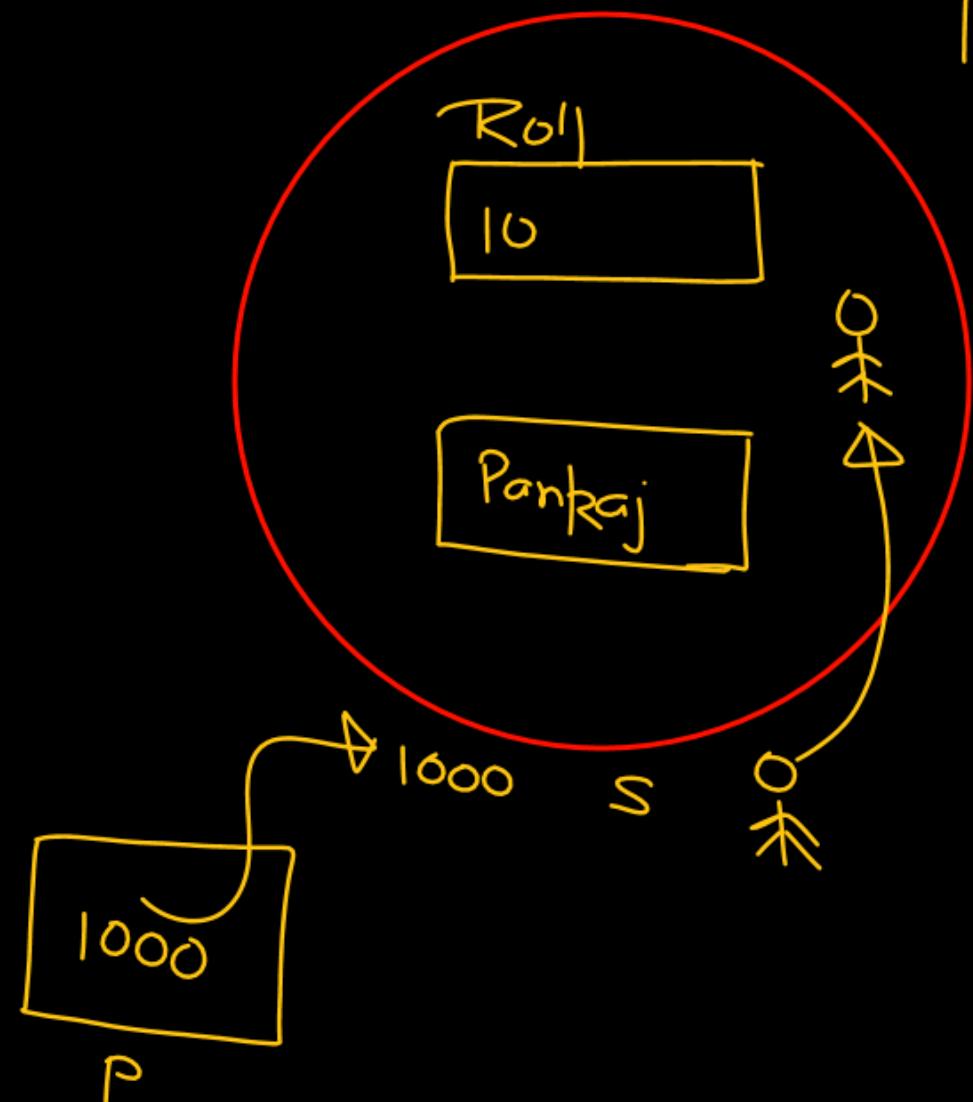
void display(struct stu t)
{
    printf("./d", t.roll);
    printf("./s", t.name);
}

```

```

struct stu{ int Roll; → 4
            char name[20]; → 20
        };
void display();
void main(){
    struct stu s= {10,"Pankaj"};
    display(&s);
}

```



```

void display(struct stu *p)
{
    printf("%d", (*p). Roll);
    printf("%s", (*p). name);
}

```

P : Pointer to structure (structure var. ~~int~~ add)

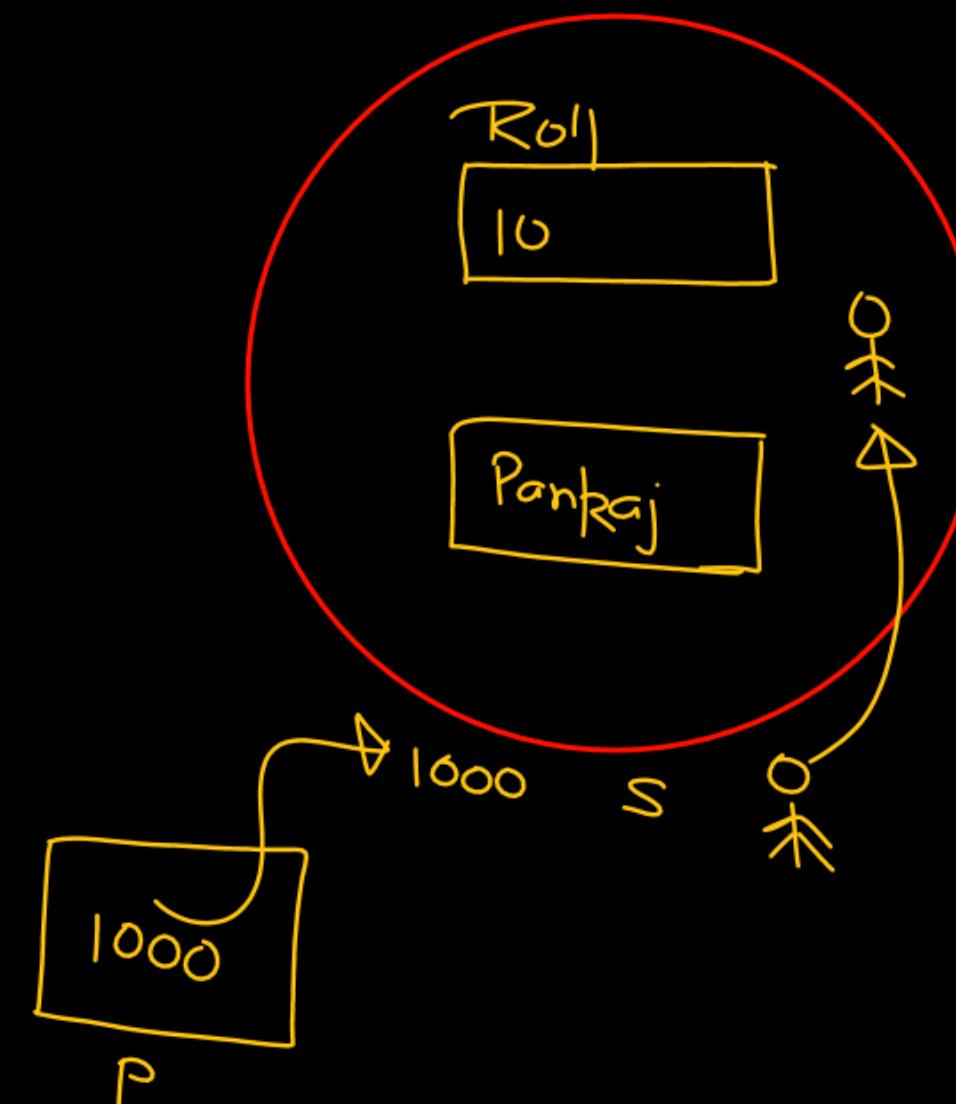
( $^*$ P).member-name

( ) $\rightarrow$

Pointers to  
structure

P  $\rightarrow$  member ✓

( $^*$ P).member



void display(struct stu  $^*$ P)

```
{  
    printf("/d", ( $^*$ P). Roll);  
    printf("/s", ( $^*$ P). name);  
}
```

```
[ printf("/d", P->Roll);  
  printf("/s", P->name); ]
```



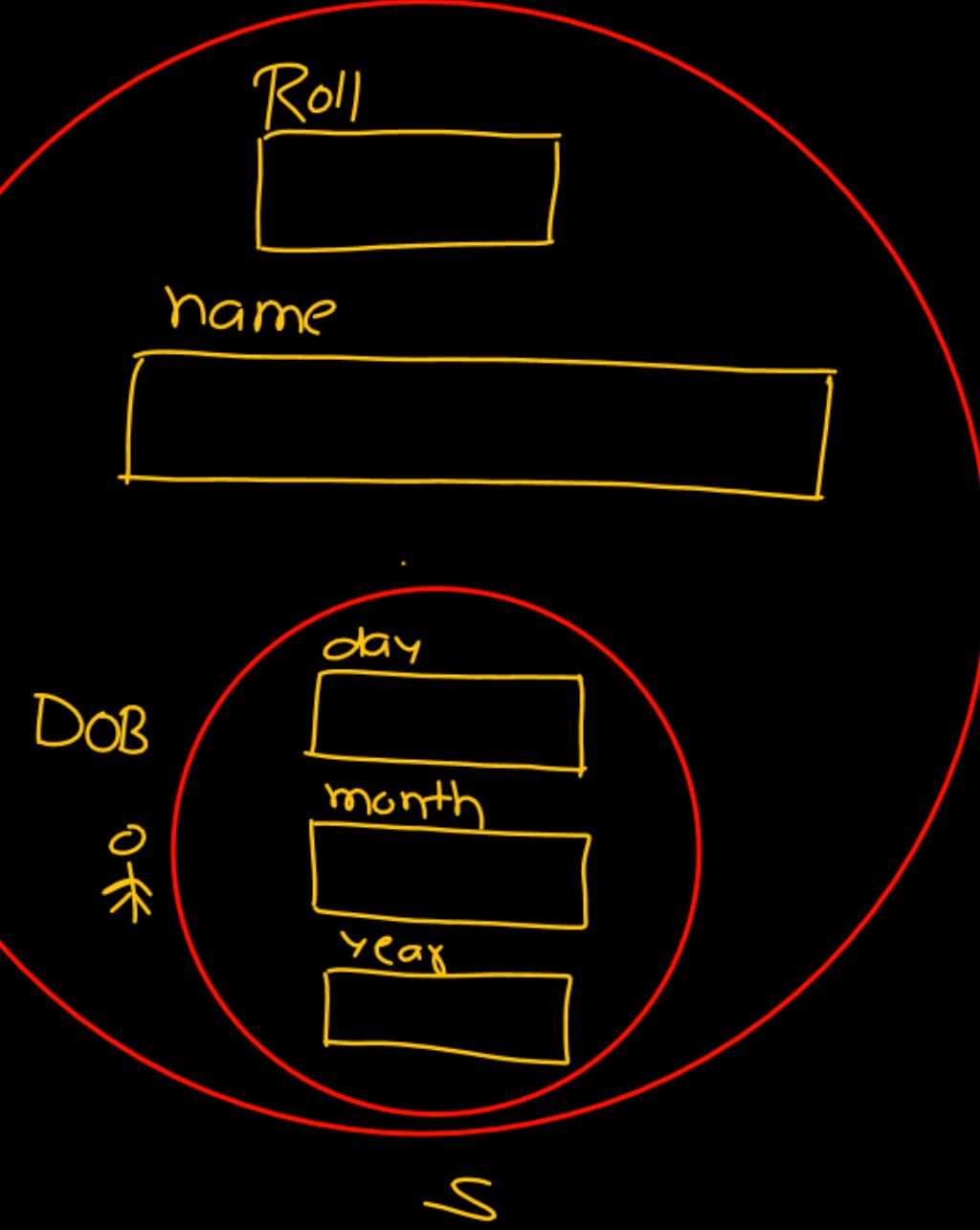
02 | 03 | 1982  
 day      ↑      ↑  
 moth     year

```

struct date-of-birth{
    int day;
    int month;
    int year;
};
```

```
struct date_of_birth{  
    int day;  
    int month;  
    int year;  
};  
  
struct stu {  
    int Roll;  
    char name[20];  
};  
  
struct date-of-birth DOB;
```

```
void main(){  
    struct stu s;  
    s.Roll = 10;  
    strcpy(s.name, "Pankaj");  
    s.DOB.day = 2;  
    s.DOB.month = 3;  
    s.DOB.year = 1982;  
}
```



```
struct date-of-birth{  
    int day;  
    int month;  
    int year;  
};
```

```
struct stu {  
    int Roll;  
    char name[20];  
    struct date-of-birth DOB;  
};
```

```
void main(){  
    struct date-of-birth d; } valid  
    struct stu s; }  
    =  
}
```

```
struct stu {
```

```
    int Roll;  
    char name[20];
```

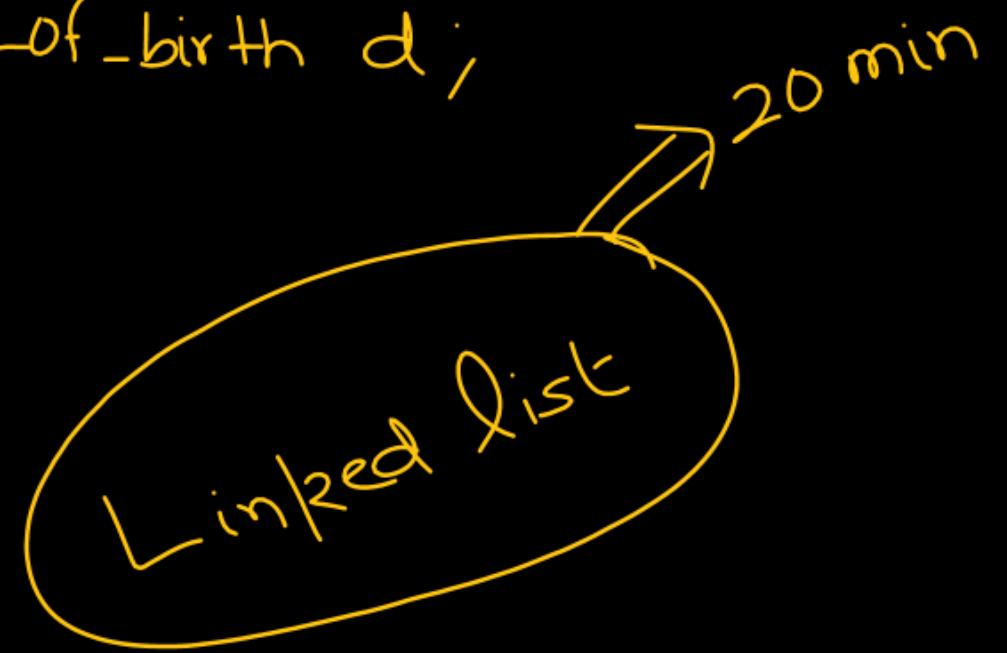
```
    struct date-of-birth{
```

```
        int day;  
        int month;  
        int year;  
    } DOB;
```

```
};
```

```
struct stu {  
    int Roll;  
    char name[20];  
    struct date-of-birth{  
        int day;  
        int month;  
        int year;  
    } DOB;  
};
```

```
void main(){  
    struct date-of-birth d;
```



—

Misc

[  
union  
scoping  
comma  
getch  
getchar  
getche  
]

7 PM

