Generally use for making Record

DATE: / 20
PAGE No

3/4/19 Draft

* Structure:

- Structure is a user-defined data type which allows us to combine data of different types together.

- It is a collection of variables (can be of different types) under a single name.

- Defining Structure:
  - keyword struct is used for creating a structure.

* Syntax:

```
struct structure_name
{
    data_type member1;
    data_type member2;
    :
    data_type member;
};
```

members can be arrays or vars

* Ex:

```
struct Person
{
    char name[50];
    int id_No;
    float salary;
};
```

DATE: / 20
PAGE No

* Creating structure variable:

M-1

```
struct Person
{
    char name[50];
    int id_NO;
    float Salary;
};
void main()
{
    struct Person Person1, Person2;
    . .
}
```

M-2

```
struct Person
{
    char name[50];
    int id_NO;
    float Salary;
} Person1, Person2;
```

DATE : / 20
PAGE No

* Accessing Structure members

- There are two type of operators used for accessing members of a structure.

1. Member operater (.)
2. Structure pointer operater (->).

Known pointer defin

Example:

```
Person 1. id_NO.
Person 1. Salary.
```

* Structure initialization:

```
struct Patient
  {
      float height;
      float weight;
       int age;
    } Person, Person 2;

Void main()
    {
       Person 1. height = 180.75;
       Person 1. weight = 65.50;
       Person 1. age = 25;
    }.
```

difference b/w union and structure

DATE: / / 20
PAGE No

```
struct Patient
{
    float height;
    float weight;
    int age;
};
void main
{
    struct Patient person1 = {180.75, 65.50, 25};
}
```

Example:

```
struct person
{
    char name[10];
    int id_No;
    float salary;
} person1;

void main()
{
  printf("Enter id");
  scanf("%d", &person1.id_No);
  printf("Enter name");
  gets(person1.name);
  printf("Enter salary");

scanf("%f", &person1.salary);
```

DATE : / 20
PAGE No

```
printf("ID: %d", person.id_no);
printf("Name:");
puts(person1.name);
printf("Salary: %f", person1.salary);
}
```

Que WAP to print Record of two students. The data of first student must be initialized and second student ~~should~~ should be during Run time.
The student must have following entities
1. Roll no
2. name
3. Section
4. Percentage
5. Address.

Ans
```
Struct Student
  {
      int roll_no.;
      char name;
      char section;
      float Percentage;
      char Address;
  } student1, student2;
```

DATE : / 20
PAGE No

5/4/19

* Typedef keyword :

* Used to assign alternative names to existing data types.

* Syntax :
  typedef <existing_name> <alias_name>;

* Example :
  typedef unsigned long ulong;
  ulong i, j;

* Also used to assign alternative names to user defined data type.

* Syntax :
  typedef struct structure_name
  {
      type member 1;
      type member 2;
      type member 3;
  } type_name;

Example :

```
typedef struct employee
{
    char name[60];
    int salary;
} emp;
```

DATE: / 20
PAGE No

```
void main()
{
    emp e1, e2;
}
```

* Nested Structure :-

```
struct Complex
{
    int imag;
    float real;
};
  struct number
{
    struct complex comp;
      int integer;
  } num1, num2;

    num1.comp.imag = 10;
    num1.comp.real = 15.75;
```

Having more no. of People

DATE : / 20
PAGE No

* Array of Structure :

```
Struct Person
{
        char name [50];
        int id_No;
        float salary;
};
    void main()
    {
        Struct Person p[10];
        p[0].id_NO = 101;
    }
```

Q. WAP to store and Print Record of 30 Students. the student will have following entities.

1. roll_no.
2. Name
3. DOB

```
Struct student
{
        char name [30];
        int rollno.;

    struct dob
      {
            int dd;
            int mm;
            int yy;
      };
```

union

DATE: / 20
PAGE No

```
struct student
{
  int roll_no;
      char name[50];
    struct dob d;
} s[30];

struct dob
{ int dd;
  int mm;
  int yy;
};

void main()
{

  for(i=0; i<30; i++)
{
    scanf("%d", &s[i].roll_no);
  gets(s[i].name);
  scanf("%d%d%d", &s[i].d.dd,
```

DATE : / 20
PAGE No

* structure as function ~~Argument : Argument~~

```
struct student
{
    char name[10];
    int roll;
};
void show( struct student st);
    void main()
{
    struct student std;
    printf("\n student name :\t");
    gets( std.name);
    printf("\n Enter student roll no :\t");
    scanf("%d", &std.roll);
    show (std);
}
    void show(struct student st)
{
    printf("\n student name is %s",
                                st.name);
    printf("\n roll no is %d", st.roll);
}
```

⇒

DATE: / 20
PAGE No

```
struct student input();
void show (struct student);
void main()
{
    struct student s;
    s = input();
    show(s);
}
struct Student input()
{
    struct student s;
    gets(s.name);
    scanf("%d", s.roll);
    return(s);
}
```