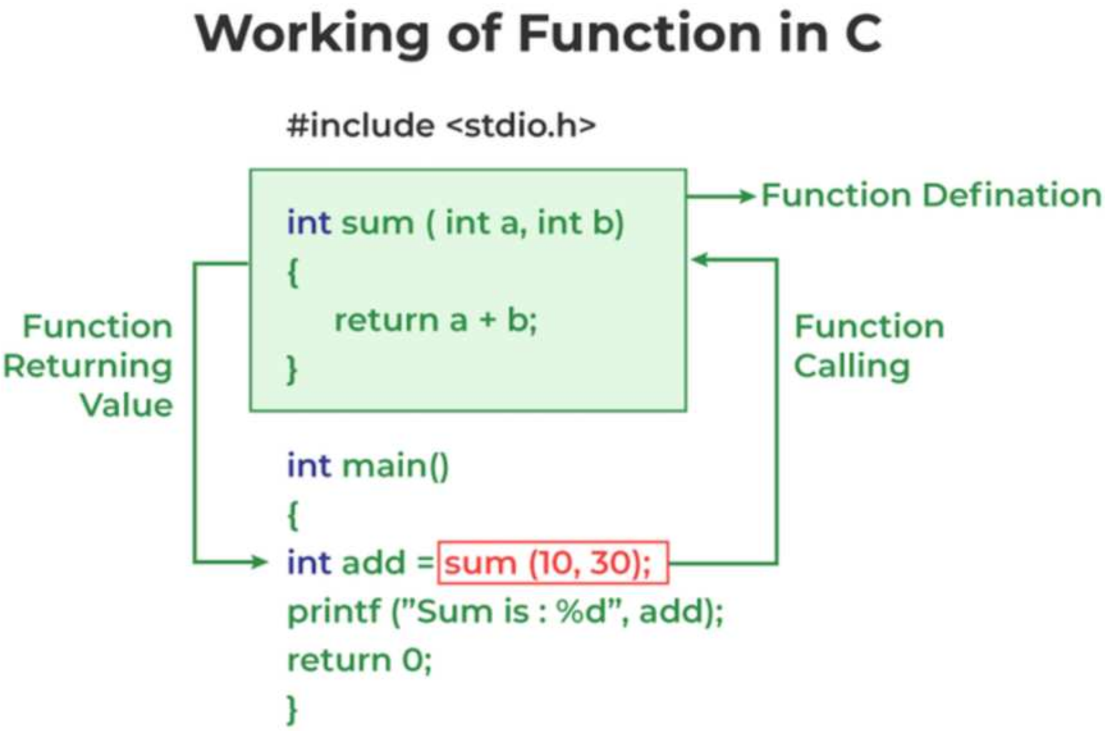
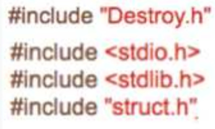


Working of Function in C

```
#include <stdio.h>

int sum ( int a, int b)
{
    return a + b;
}

int main()
{
int add = sum (10, 30);
printf ("Sum is : %d", add);
return 0;
}
```

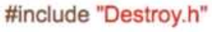
main.c



```
#include "Destroy.h"
#include <stdio.h>
#include <stdlib.h>
#include "struct.h"
```

```
 int main(){
        .
        .
        printf..
        .
        .
        Destroy()
        .
        .
        return 0;

 }
```

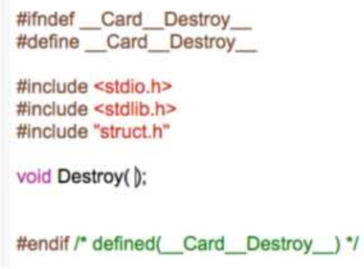
Destroy.c



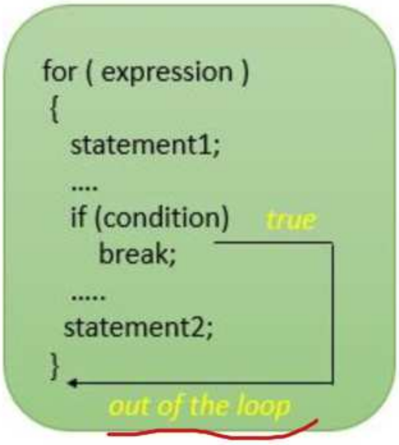
```
#include "Destroy.h"
```

```
void Destroy(){
.
.

.
}
```

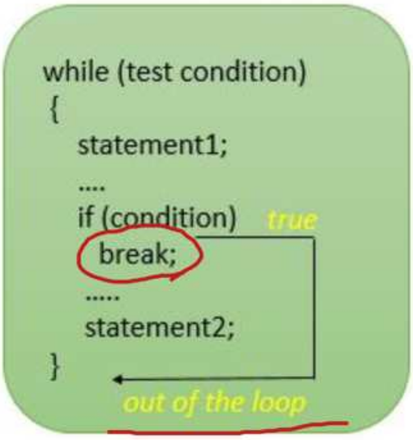
Destroy.h



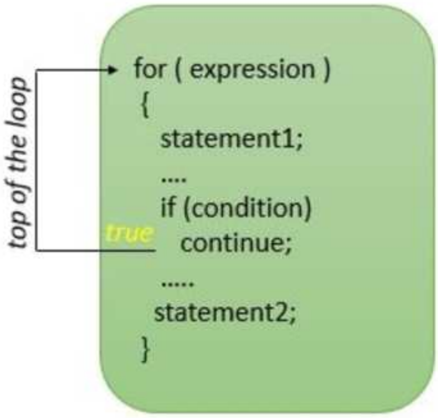
```
#ifndef __Card__Destroy__
#define __Card__Destroy__

#include <stdio.h>
#include <stdlib.h>
#include "struct.h"

void Destroy( );


#endif /* defined(__Card__Destroy__) */
```



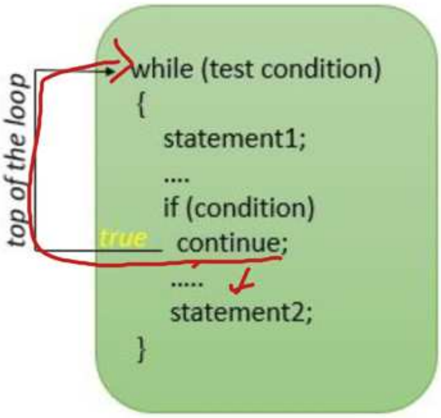
```
for ( expression )
 {
   statement1;
   ....
   if (condition)
      break;
   .....
  statement2;
 }
```



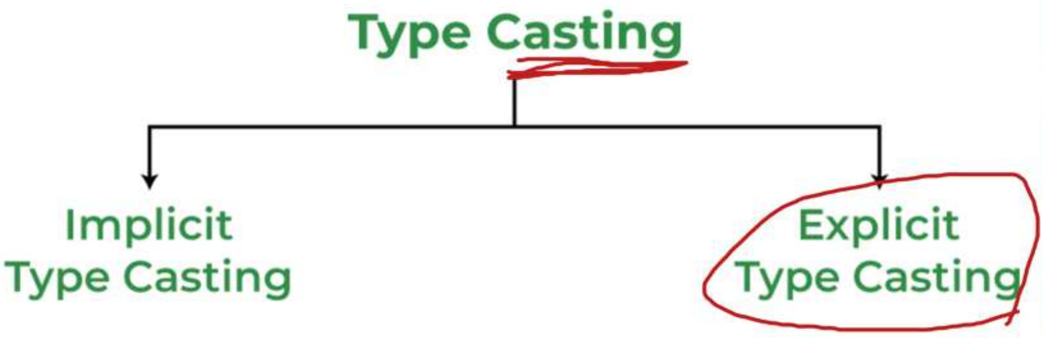
```
while (test condition)
 {
    statement1;
    ....
    if (condition)
       break;
     .....
    statement2;
 }
```



```
for ( expression )
 {
   statement1;
   ....
   if (condition)
     continue;
   .....
  statement2;
 }
```



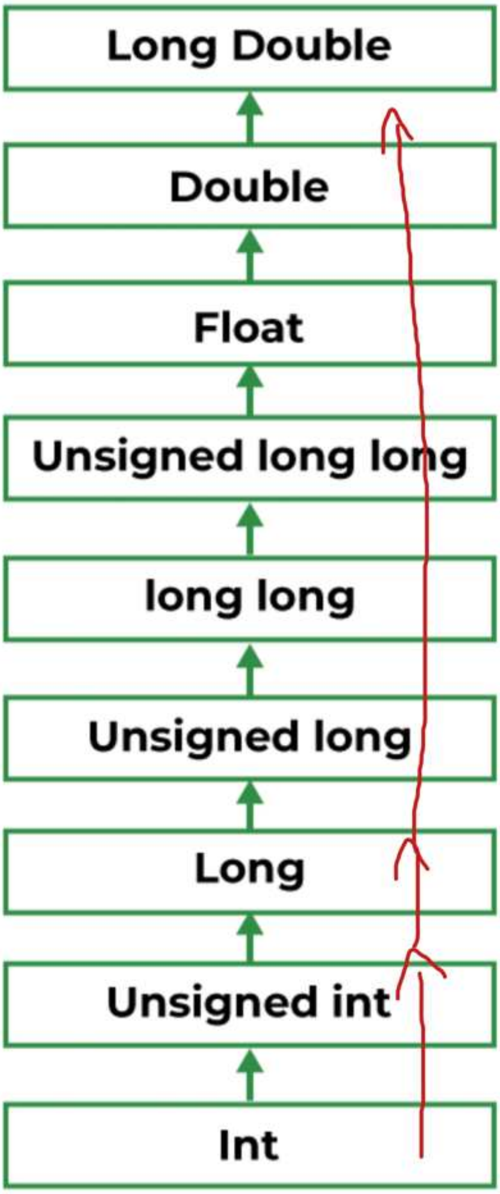
```
while (test condition)
 {
    statement1;
    ....
    if (condition)
     continue;
     .....
    statement2;
 }
```



Type Casting

Implicit Type Casting

Explicit Type Casting



Long Double

Double

Float

Unsigned long long

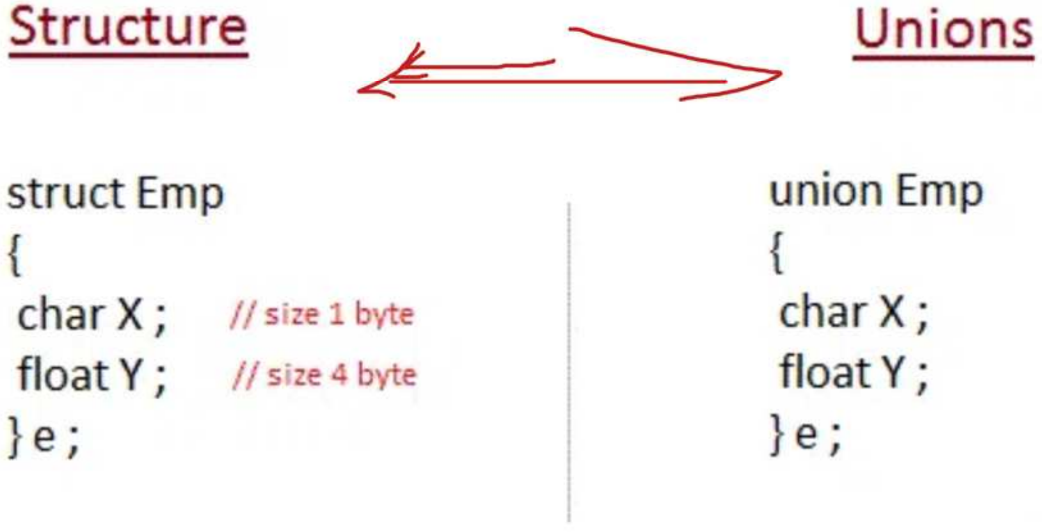
long long

Unsigned long

Long

Unsigned int

Int

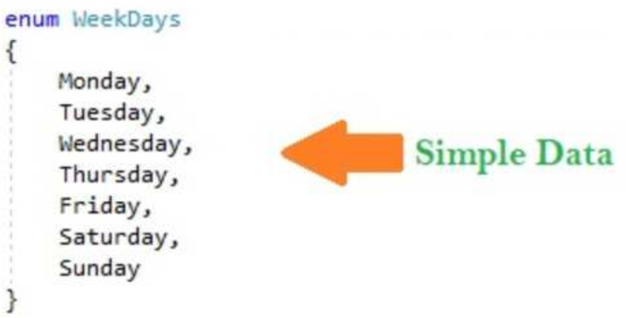


Structure

Unions

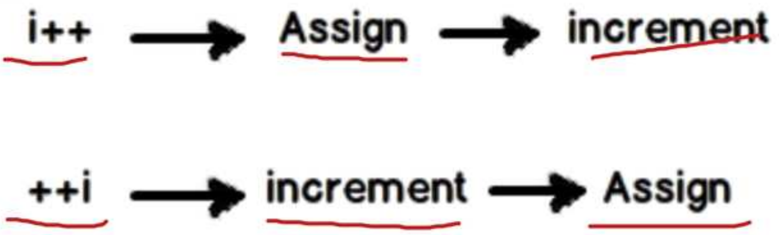
```
struct Emp
{
 char X ;      // size 1 byte
 float Y ;     // size 4 byte
} e ;
```

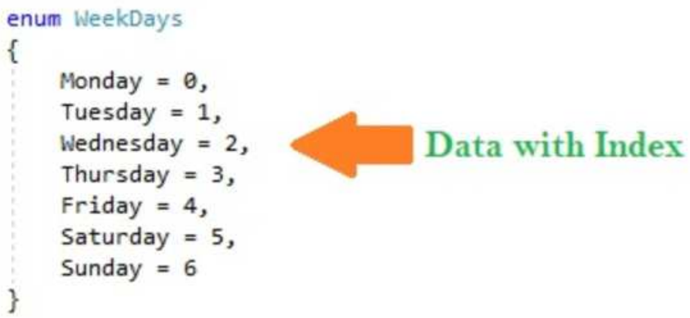
```
union Emp
{
 char X ;
 float Y ;
} e ;
```



```
enum WeekDays
{
    Monday,
    Tuesday,
    Wednesday,
    Thursday,
    Friday,
    Saturday,
    Sunday
}
```

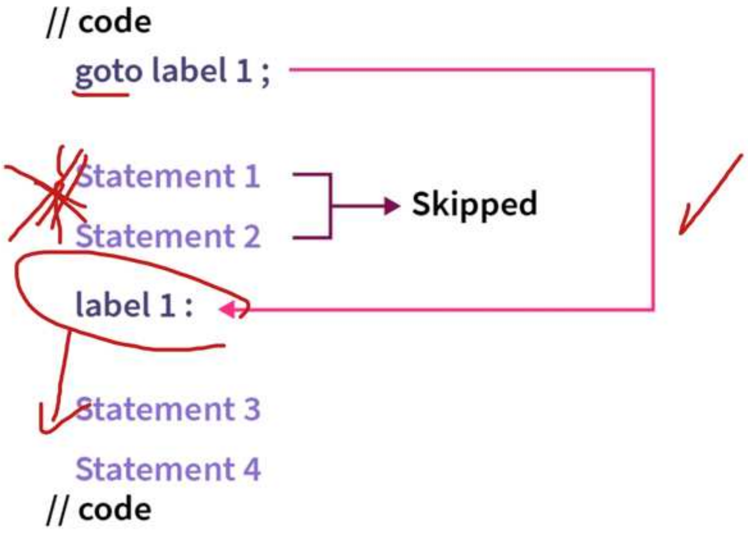
Simple Data



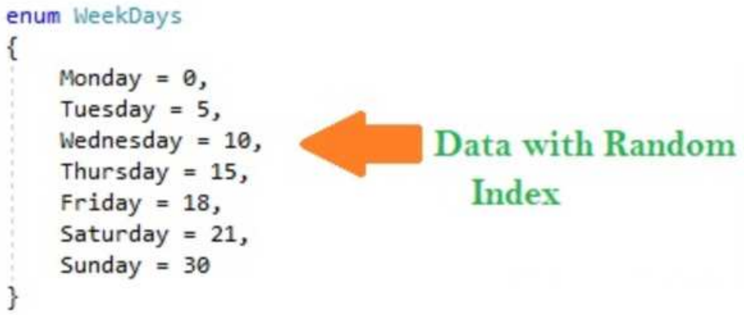


```
enum WeekDays
{
    Monday = 0,
    Tuesday = 1,
    Wednesday = 2,
    Thursday = 3,
    Friday = 4,
    Saturday = 5,
    Sunday = 6
}
```

Data with Index

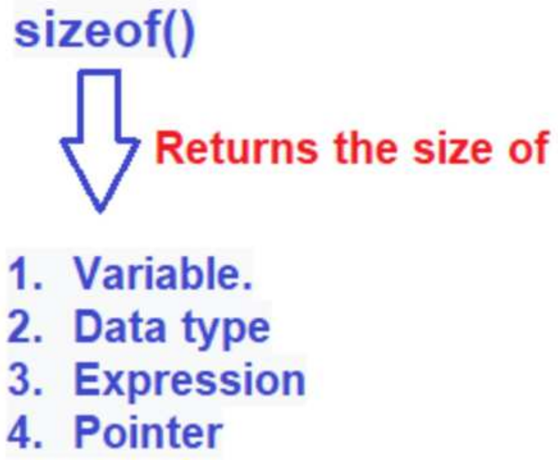


```
// code
  goto label 1 ;

  ~~Statement 1~~
  Statement 2
  label 1 :

  Statement 3
  Statement 4
// code
```



```
enum WeekDays
{
    Monday = 0,
    Tuesday = 5,
    Wednesday = 10,
    Thursday = 15,
    Friday = 18,
    Saturday = 21,
    Sunday = 30
}
```

Data with Random Index



sizeof()

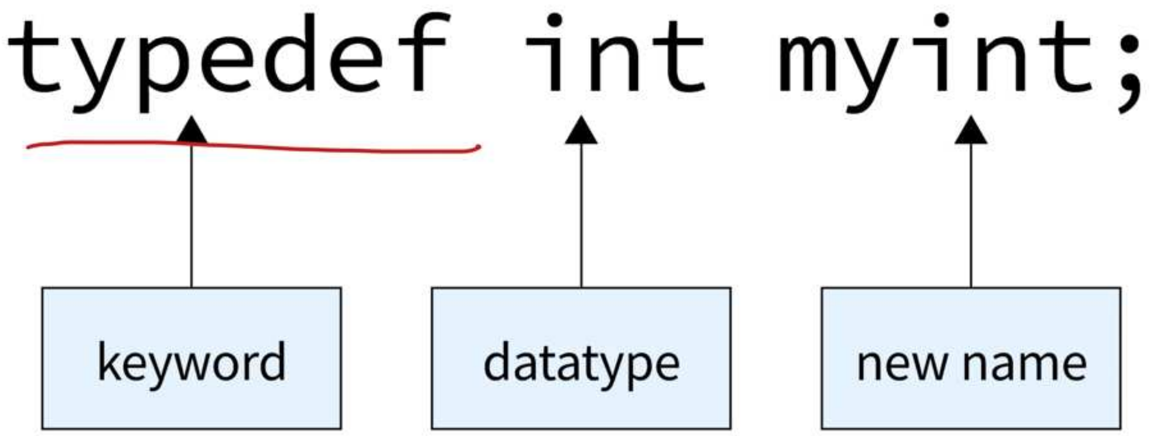
Returns the size of

1. Variable.
2. Data type
3. Expression
4. Pointer

Volatile

Keyword in C

Register



```
typedef int myint;
```