STORAGE CLASS

VARIABLES

1. DATA   TYPE (int , float , char , ...)
2. STORAGE CLASS

STORAGE CLASS

1. SCOPE   :- LOCAL , GLOBAL

2. MEMORY   :- RAM   , CPU-REGISTER

3. DEFAULT VALUE :- GARBAGE VALUE , ZERO

TYPE OF STORAGE CLASS

1. **AUTOMATIC (LOCAL)   STORAGE CLASS**
2. **EXTERNAL (GLOBAL)   STORAGE CLASS**
3. **STATIC   STORAGE CLASS**
4. **REGISTER   STORAGE CLASS**

e.g. **auto   int   a ;**
        storage   data   variable
        class   type   name

1. **AUTOMATIC (LOCAL) STORAGE CLASS**

keyword :- auto

e.g.   :- auto int a ; // int a;

1. SCOPE   :-   LOCAL
2. MEMROY   :-   RAM
3. DEFAULT VALUE   :-   GARBAGE VALUE

```
#include<stdio.h>
  int    main()
  {
        auto  int  a  = 3 ;
        printf(" %d\n " , a ); // 3

        {
            auto int b = 5;
            printf(" %d\n " , a); // 3
            printf(" %d\n " , b); // 5
        }
        printf(" %d\n " , a ); // 3
        printf(" %d " , b );   //error :- undefined symbol 'b'
  }
```

```c
#include<stdio.h>
  void   fun();

  int    main()
  {
          auto  int  a = 3;
          printf(" %d\n ", a); // 3
          fun();
  }
  void   fun()
  {
      printf(" %d\n " , a);   // error :- undefined symbol 'a'
  }
```

```c
          t = fun( a );

          int   fun( int  p )
          {
                  int  b;
                  b = p * p ;
                  return( b);
          }
```

----

2. REGISTER   STORAGE CLASS

keyword :- register

e.g.   :- register int i ;

register char i ;

1. SCOPE   :- LOCAL
2. MEMROY   :- CPU - REGISTER
3. DEFAULT VALUE :- GARBAGE VALUE

AUTOMATIC STORAGE   CLASS

hard disc ---> ram ---> cpu - register

REGISTER   STORAGE   CLASS

hard disc   ---> cpu - register

----

3. EXTERNAL (GLOBAL) STORAGE CLASS

keyword :- extern

e.g.   :- extern int a ;

1. SCOPE   :- GLOBAL
2. MEMROY   :- RAM
3. DEFAULT VALUE :- ZERO

```c
#include<stdio.h>

int   a = 3;  // global
void  fun();
int  main()
{
    printf("%d\n", a); // 3

    fun();
}
void  fun()
{
    printf("%d\n", a);  // 3
}
```

```c
#include<stdio.h>
 void  fun() ; // extern void fun();

 int   main()
 {
     extern int  a ;  // ONLY DECLARATION

     printf("   %d\n " , a); // 3

     fun();
 }
int   a = 3;   // global
void   fun()
{
     printf("  %d\n " , a );   // 3
}
```

----

4. STATIC STORAGE CLASS
   keyword :- static

   e.g.   :- static int a ;

   1. SCOPE   :- LOCAL or GLOBAL
   2. MEMROY   :- RAM
   3. DEFAULT VALUE :- ZERO

```c
#include<stdio.h>
  void   fun();
  int    main()
  {
          fun();
          fun();
          fun();
  }
  void    fun()
  {
          int  i = 1;
          printf("    %d\n", i);
          i++;
  }
```

----

            ans :-   1
                        1
                        1

```c
#include<stdio.h>

 void  fun();

 int   main()
 {
        fun();   // 1
        fun();   // 2
        fun();   // 3
 }
 void  fun()
 {
      static  int  i = 1;
      printf("   %d\n" , i);
      i++ ;
 }
```