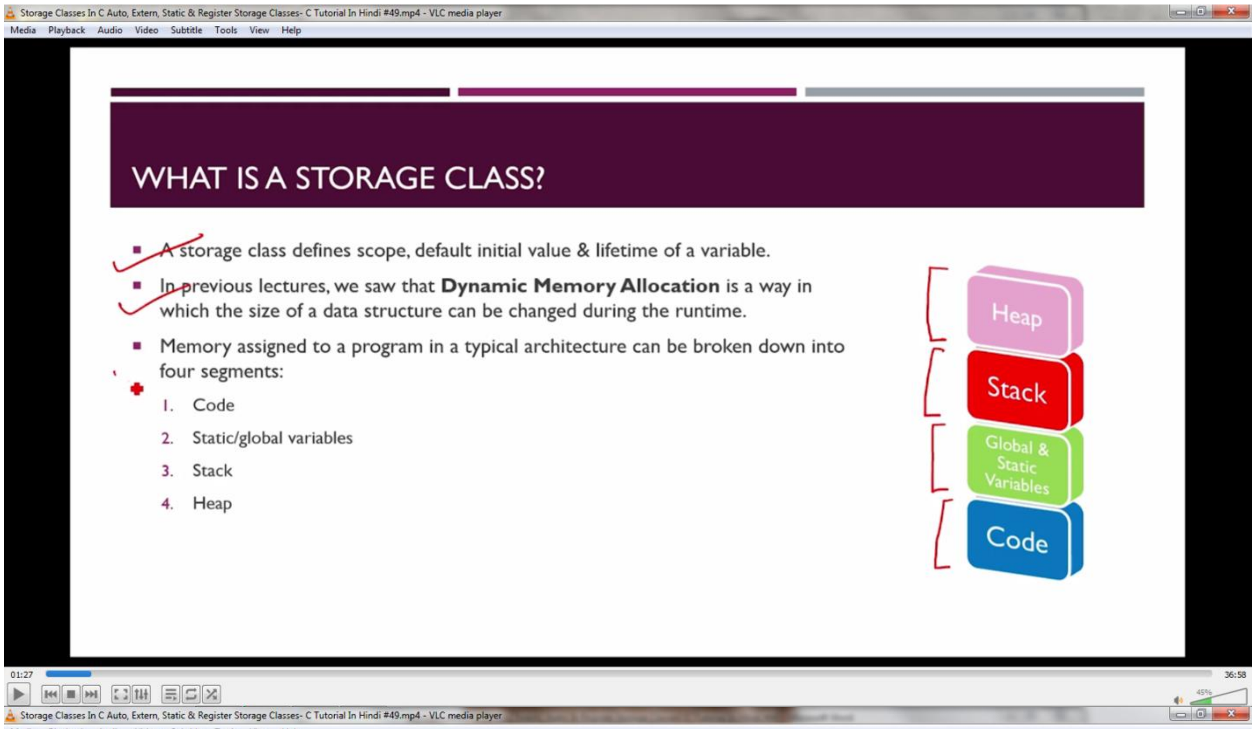
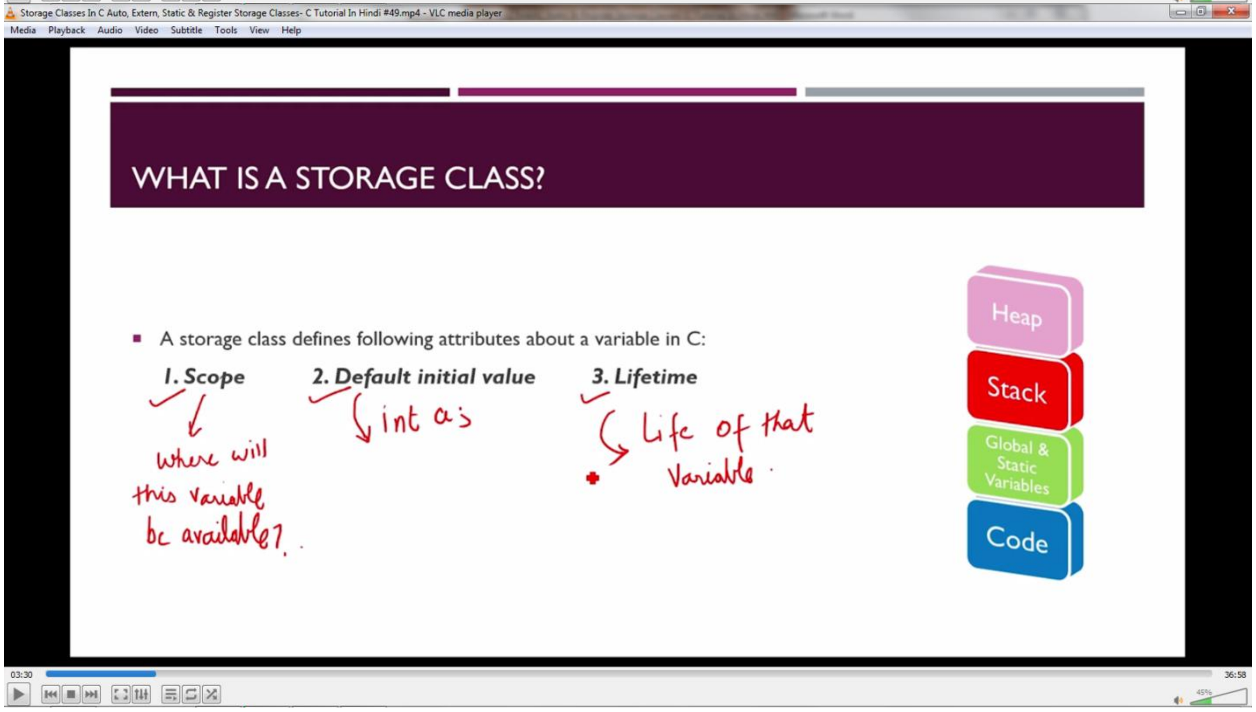

Storage Classes In C Auto, Extern, Static & Register Storage Classes- C Tutorial In Hindi #49.mp4 - VLC media player
Media Playback Audio Video Subtitle Tools View Help
Heap
Stack
Global & Static Variables
Code
01:27
36:58

WHAT IS A STORAGE CLASS?

- A storage class defines scope, default initial value & lifetime of a variable.
- In previous lectures, we saw that **Dynamic Memory Allocation** is a way in which the size of a data structure can be changed during the runtime.
- Memory assigned to a program in a typical architecture can be broken down into four segments:
  1. Code
  2. Static/global variables
  3. Stack
  4. Heap


Storage Classes In C Auto, Extern, Static & Register Storage Classes- C Tutorial In Hindi #49.mp4 - VLC media player
Media Playback Audio Video Subtitle Tools View Help
Heap
Stack
Global & Static Variables
Code
03:30
36:58

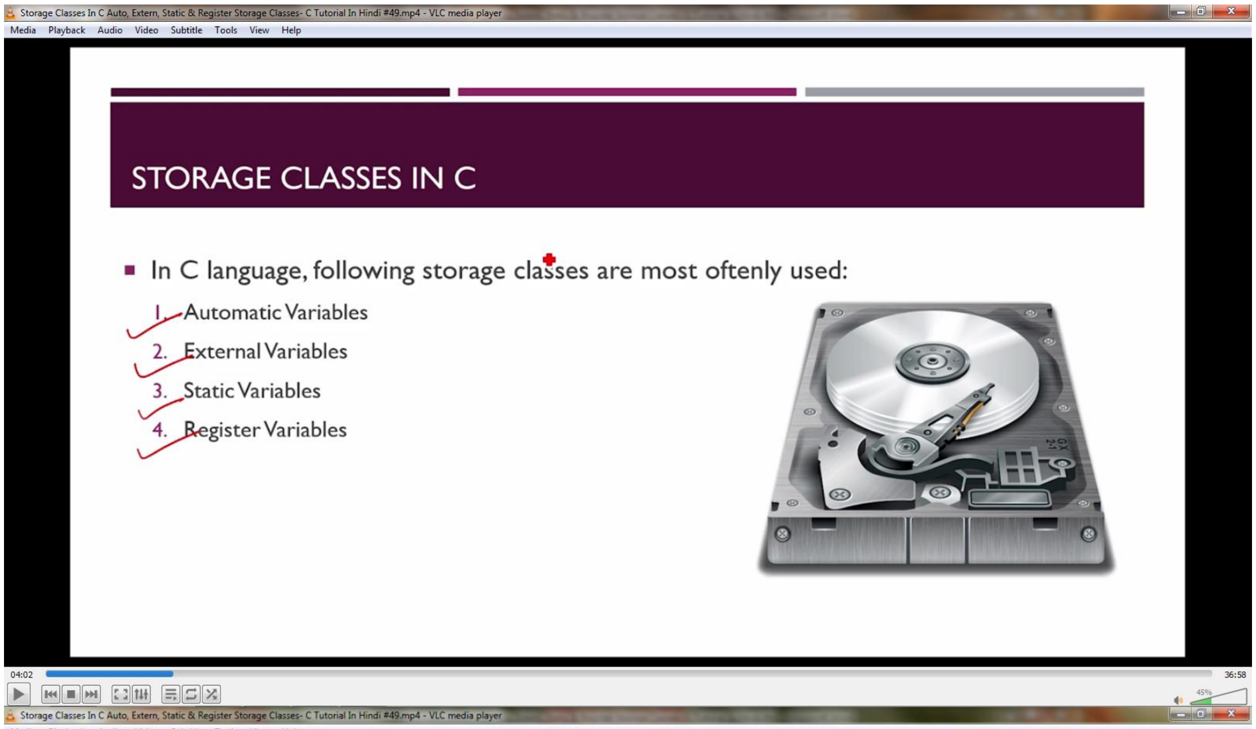
WHAT IS A STORAGE CLASS?

- A storage class defines following attributes about a variable in C:

1. Scope — where will this variable be available?

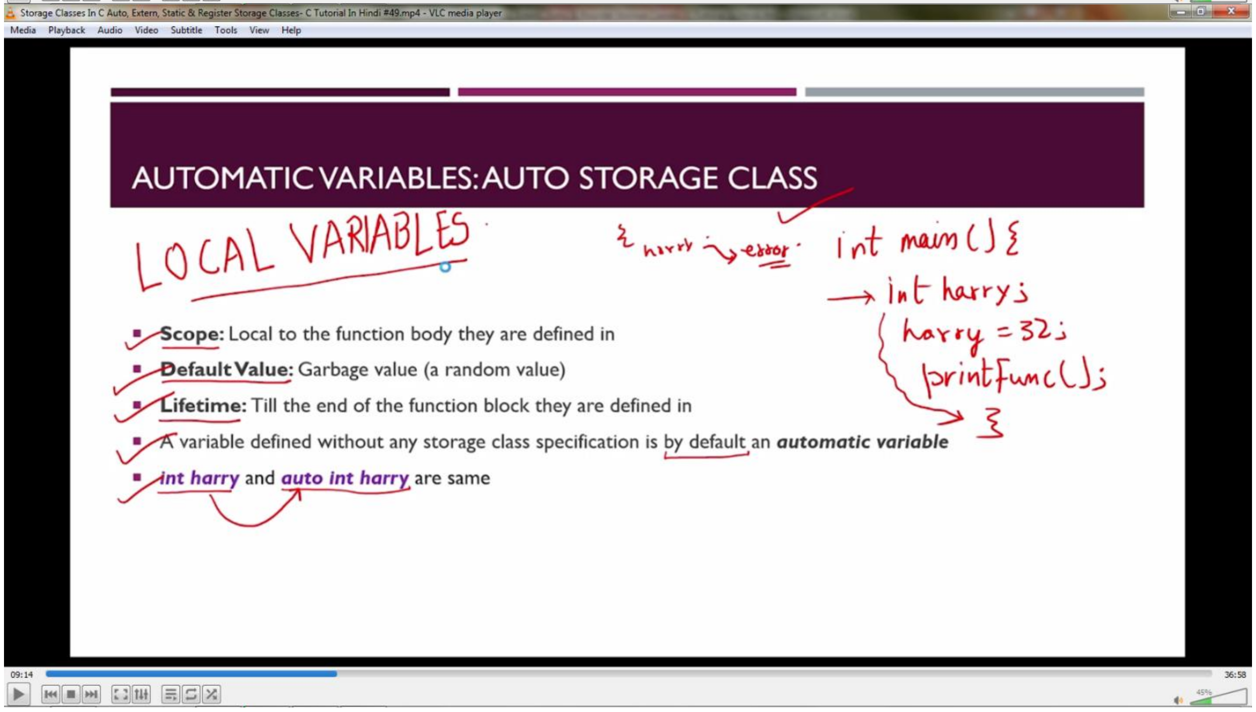
2. Default initial value — int a;

3. Lifetime — Life of that variable.


Storage Classes In C Auto, Extern, Static & Register Storage Classes- C Tutorial In Hindi #49.mp4 - VLC media player
Media Playback Audio Video Subtitle Tools View Help
04:02
36:58
45%

STORAGE CLASSES IN C

- In C language, following storage classes are most oftenly used:
  1. Automatic Variables
  2. External Variables
  3. Static Variables
  4. Register Variables


Storage Classes In C Auto, Extern, Static & Register Storage Classes- C Tutorial In Hindi #49.mp4 - VLC media player
Media Playback Audio Video Subtitle Tools View Help
09:14
36:58
45%

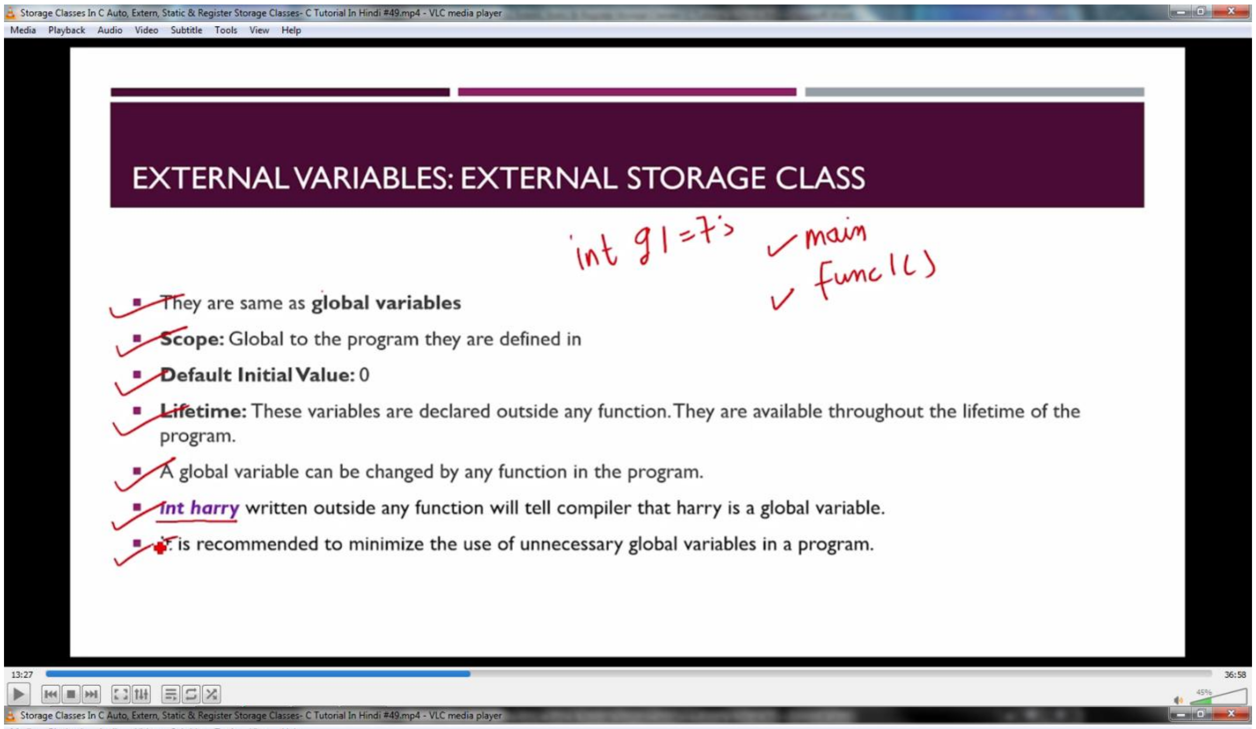
AUTOMATIC VARIABLES: AUTO STORAGE CLASS

LOCAL VARIABLES

{ harry → error.

```
int main() {
    int harry;
    harry = 32;
    printFunc();
}
```

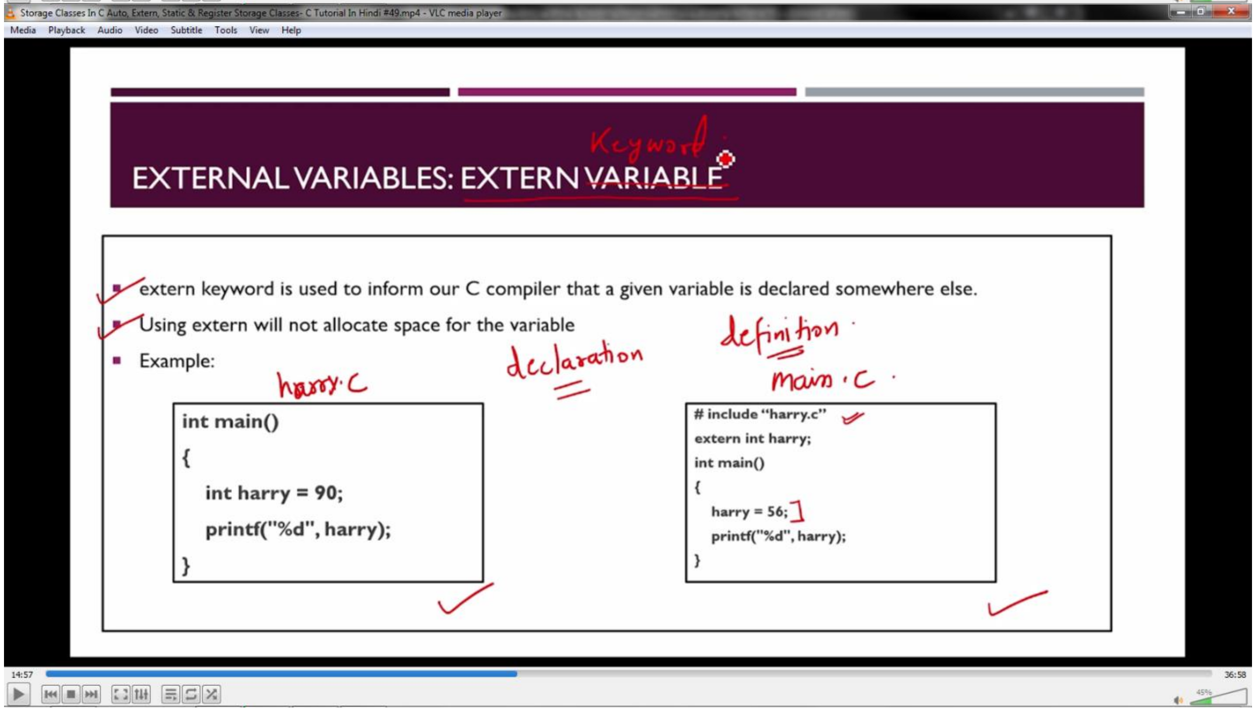
- **Scope:** Local to the function body they are defined in
- **Default Value:** Garbage value (a random value)
- **Lifetime:** Till the end of the function block they are defined in
- A variable defined without any storage class specification is by default an ***automatic variable***
- ***int harry*** and ***auto int harry*** are same


Storage Classes In C Auto, Extern, Static & Register Storage Classes- C Tutorial In Hindi #49.mp4 - VLC media player
Media Playback Audio Video Subtitle Tools View Help
13:27
36:58

EXTERNAL VARIABLES: EXTERNAL STORAGE CLASS

int g1 = 7;
✓ main
✓ func()

- They are same as **global variables**
- **Scope:** Global to the program they are defined in
- **Default Initial Value:** 0
- **Lifetime:** These variables are declared outside any function. They are available throughout the lifetime of the program.
- A global variable can be changed by any function in the program.
- ***int harry*** written outside any function will tell compiler that harry is a global variable.
- It is recommended to minimize the use of unnecessary global variables in a program.


Storage Classes In C Auto, Extern, Static & Register Storage Classes- C Tutorial In Hindi #49.mp4 - VLC media player
Media Playback Audio Video Subtitle Tools View Help
14:57
36:58

EXTERNAL VARIABLES: EXTERN VARIABLE

Keyword

- extern keyword is used to inform our C compiler that a given variable is declared somewhere else.
- Using extern will not allocate space for the variable
- Example:

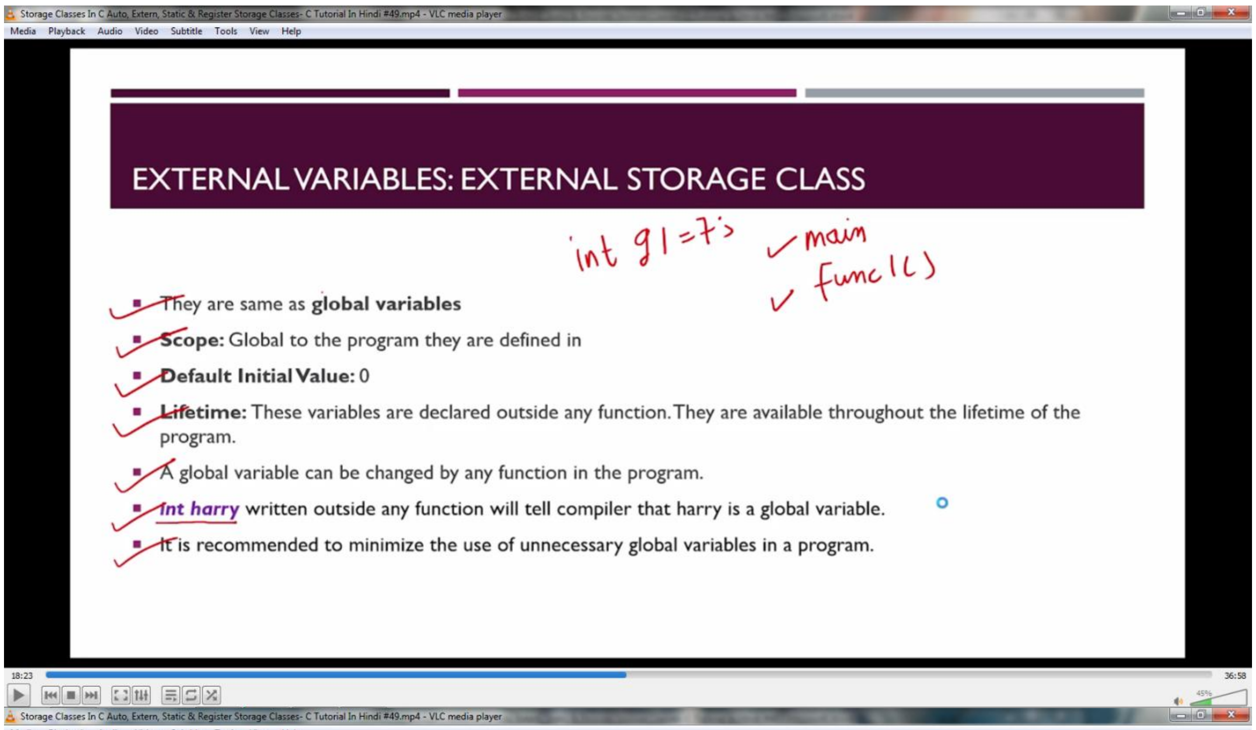
declaration
definition

harry.c

```
int main()
{
   int harry = 90;
   printf("%d", harry);
}
```

main.c

```
# include "harry.c"
extern int harry;
int main()
{
   harry = 56;
   printf("%d", harry);
}
```


Storage Classes In C Auto, Extern, Static & Register Storage Classes- C Tutorial In Hindi #49.mp4 - VLC media player

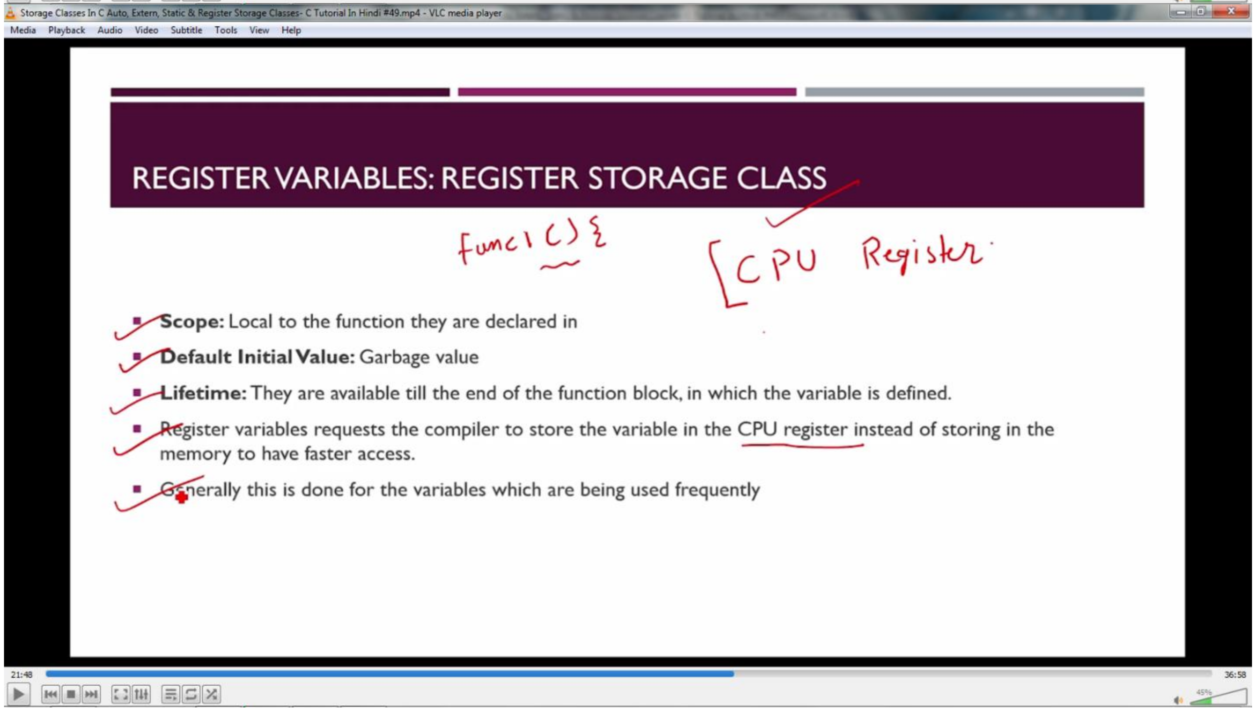
EXTERNAL VARIABLES: EXTERNAL STORAGE CLASS

int g1=7;

✓ main

✓ func1()

- They are same as **global variables**
- **Scope:** Global to the program they are defined in
- **Default Initial Value:** 0
- **Lifetime:** These variables are declared outside any function. They are available throughout the lifetime of the program.
- A global variable can be changed by any function in the program.
- ***int harry*** written outside any function will tell compiler that harry is a global variable.
- It is recommended to minimize the use of unnecessary global variables in a program.


Storage Classes In C Auto, Extern, Static & Register Storage Classes- C Tutorial In Hindi #49.mp4 - VLC media player

REGISTER VARIABLES: REGISTER STORAGE CLASS

func1(){

[CPU Register

- **Scope:** Local to the function they are declared in
- **Default Initial Value:** Garbage value
- **Lifetime:** They are available till the end of the function block, in which the variable is defined.
- Register variables requests the compiler to store the variable in the CPU register instead of storing in the memory to have faster access.
- Generally this is done for the variables which are being used frequently