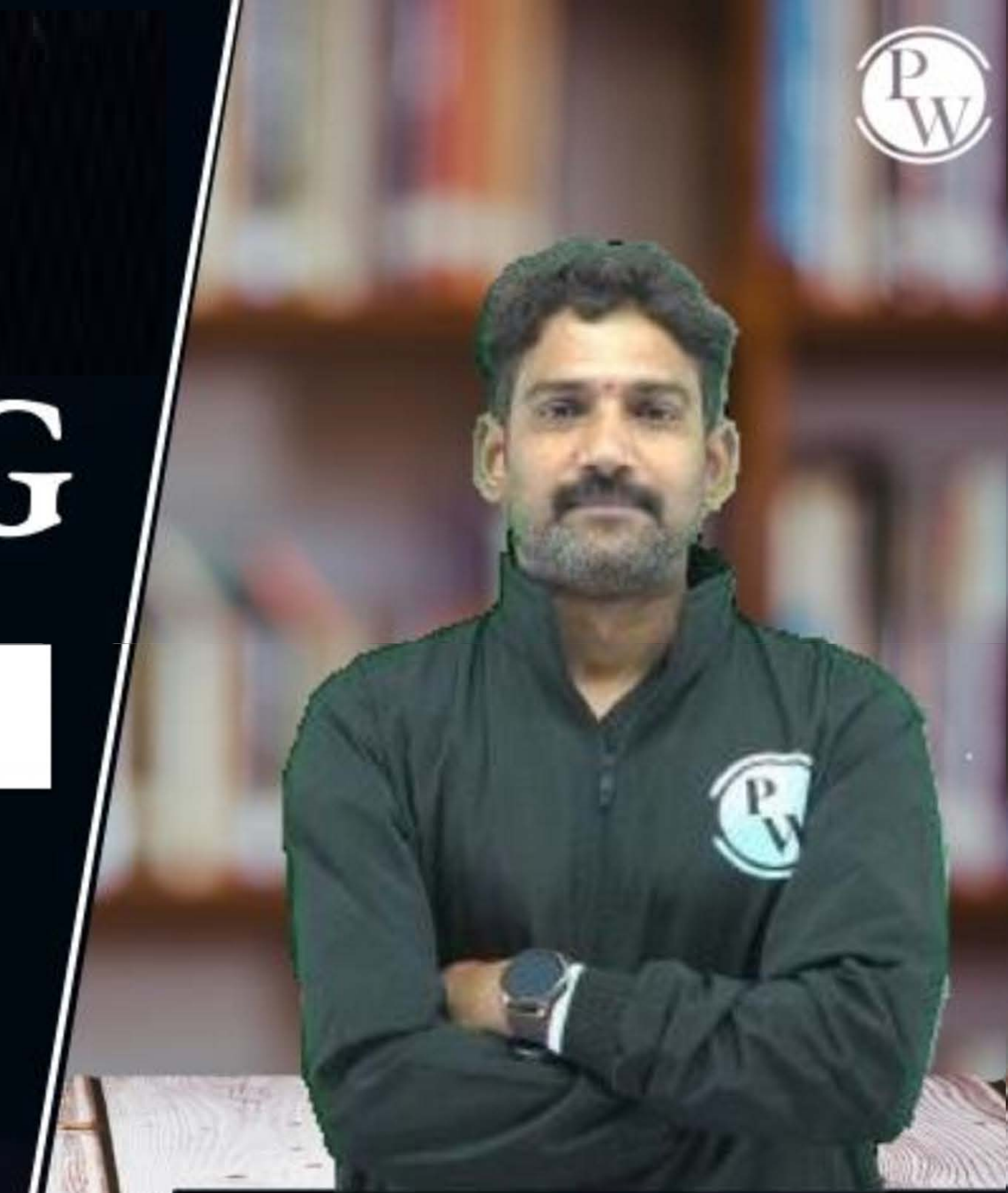
CS & IT ENGINEERING

'C' Programming

File Handling

ONE SHOT



By- Satya sir

Recap of Previous Lecture

- Pre Processor Directives
  - File inclusion
  - Macro Substitution as a Variable

Topics to be Covered

- Macro Substitution as function
- Conditional Compilation Directives
- File Handling

Topic : Pre Processor Directives

Function as Macro

Ex:

```
#define X(i, j)    i*j
Void main( )
{
    int a=5, b=6;
    Printf("%d", x(a,b));
}
```

o/p : 30

// Macro is Processed only first time (one time)

// Less Execution time

```
void x (int i, int j)
{
    return i*j;
}
Void main( )
{
    int a=5, b=6;
    Printf("%d", x(a,b));
}
```

o/p : 30

// On Every Execution, X( ) is called, i, j variables are created.

// More Execution time

Topic : Pre Processor Directives

Conditional Compilation Directives

#if	≅ if
#elif	≅ else if
#else	≅ else
#ifdef	
#ifndef	
#endif	≅ // End of if block

```
if ( Expression )
{
      ≡ //stmts
}
else
{
   1 stmt;
}
```

// if block statements will be compiled always irrespective of condition result.

```
#if Exp
      ≡
#else
      —
#endif
```

// statements of if will be compiled only when Expression is TRUE.

Topic : Pre Processor Directives

Examples :

```
#define  x  5 // x=5
 #if  x >1   True
   #undef x //x is undefined
 #define  x  10 //x=10
   #ifdef  x  // True
    # undef x // x is undefined
  # ifndef  x // True
     #define  x   20 //x=20
  # undef x // x is undefined
  # ifdef  x  // FALSE
    #define  x 30 //Not
                   Compiled
```

```
#else
   #define x  50   // x=50
#end if

void  ABC(  )
{
    Printf(" %d", x);  // x=50
 }
void  main(  )
{
    ABC( );
}
```

o/p: 50

Topic : File Handling

- // Normal Program, when Executed, results are displayed, Stored in RAM, Deleted on System turnoff.

  so, Past Execution generated results, if to be Stored Permanently, Files are Useful.

- Let i/p array of 200 numbers, need to be given Commonly for 5 Programs.

  Ex: $\underbrace{A[10][50]}_{\text{500 numbers}}$, $\underbrace{B[50][10]}_{\text{500 numbers}}$

  - Matrix addition $A+B$
  - Matrix multiplication $A*B$, $B*A$
  - Transpose A & B
  - Inverse of A, B

  Common input, Though Need to feed repeatedly.

  Instead, if input Stored in Files, All Can access it.

Topic : File Handling

Steps for File Handling

① Create a File Pointer

FILE *Pointer Name; Ex: FILE *fp;

② open File and Keep reference at Pointer

File Pointer = fopen("File Name", "mode");

Ex: fp = fopen("Sample.txt", "w");

③ Perform i/o operation →

i/p	o/p
fscanf( )	fprintf( )
fgets( )	fputs( )
fgetc( )	fputchar( )
fgetw( )	fputw( )

④ close file.

Files
- Text File (.txt)
- Binary File (.bin)

Mode (Purpose)
- R (Read)
- w (write)
- A (Append)

Topic : File Handling

fscanf( )

```
int i, char str[10], float x;

fscanf( File Pointer, "format specifiers", &variables );

Ex: fscanf( fp, "%d %s %f" &i, str, &x);
```

fprintf( )

```
fprintf( File Name, "format specifiers", Variables );

fprintf( fp, "%d %s %f", i, str, x);
```

Topic : File Handling

Close File : fclose(Pointer name);

Ex: fclose(fp);

- fseek( ) : To take control to required Position/index
- ftell( ) : Current Position of the control
- rewind( ) : To move file control back to Starting of file.
- EOF( ) : End of file

2 mins Summary

- Conditional Compilation

- File Handling

- Stay Tuned for PYQ Practice Sessions.

THANK - YOU