# C LIBRARY FUNCTION - FSETPOS



## Description

The C library function **int fsetpos***FILE ∗ stream, constfpos*$_t$ *∗ pos* sets the file position of the given **stream** to the given position. The argument **pos** is a position given by the function fgetpos.

## Declaration

Following is the declaration for fsetpos function.

```
int fsetpos(FILE *stream, const fpos_t *pos)
```

## Parameters

- **stream** − This is the pointer to a FILE object that identifies the stream.
- **pos** − This is the pointer to a fpos_t object containing a position previously obtained with fgetpos.

## Return Value

This function returns zero value if successful, or else it returns a non-zero value and sets the global variable **errno** to a positive value, which can be interpreted with perror.

## Example

The following example shows the usage of fsetpos function.

```
#include <stdio.h>

int main ()
{
   FILE *fp;
   fpos_t position;

   fp = fopen("file.txt","w+");
   fgetpos(fp, &position);
   fputs("Hello, World!", fp);

   fsetpos(fp, &position);
   fputs("This is going to override previous content", fp);
   fclose(fp);

   return(0);
}
```

Let us compile and run the above program to create a file **file.txt** which will have the following content. First of all we get the initial position of the file using **fgetpos** function, and then we write *Hello, World!* in the file but later we used **fsetpos** function to reset the write pointer at the beginning of the file and then over-write the file with the following content −

```
This is going to override previous content
```

Now let's see the content of the above file using the following program −

```
#include <stdio.h>

int main ()
{
   FILE *fp;
```

```
   int c;

   fp = fopen("file.txt","r");
   while(1)
   {
      c = fgetc(fp);
      if( feof(fp) )
      {
         break;
      }
      printf("%c", c);
   }
   fclose(fp);
   return(0);
}
```

Let us compile and run the above program to produce the following result –

```
This is going to override previous content
```