errno, perror(), and strerror()

The C programming language provides perror() and strerror() functions which can be used to display the text message associated with errno.

- The perror() function displays the string you pass to it, followed by a colon, a space, and then the textual representation of the current errno value.
- The strerror() function, which returns a pointer to the textual representation of the current errno value.

Let's try to simulate an error condition and try to open a file which does not exist. Here I'm using both the functions to show the usage, but you can use one or more ways of printing your errors. Second important point to note is that you should use stderr file stream to output all the errors.

```
#include <stdio.h>
#include <errno.h>
#include <string.h>
extern int errno ;
int main () {
   FILE * pf;
   int errnum;
   pf = fopen ("unexist.txt", "rb");
   if (pf == NULL) {
      errnum = errno;
      fprintf(stderr, "Value of errno: %d\n", errno);
      perror("Error printed by perror");
      fprintf(stderr, "Error opening file: %s\n", strerror( errnum
));
   } else {
      fclose (pf);
   }
   return 0;
}
```

When the above code is compiled and executed, it produces the following result −

```
Value of errno: 2
Error printed by perror: No such file or directory
Error opening file: No such file or directory
```