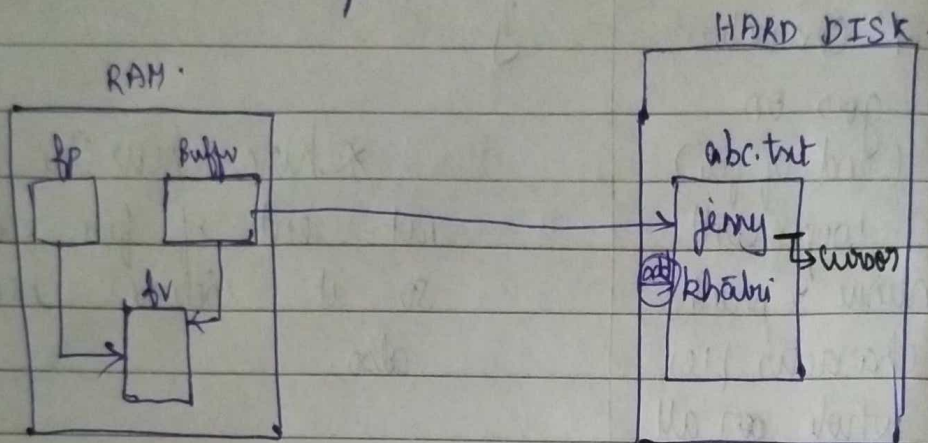


C-123 \Rightarrow Append Mode in File Handling

* How to add more content in existing file without overwriting into a file?
This is done using append mode.

* In write mode the previous content will be ~~erased~~ and written but in append mode the previous content is not ~~erased~~ and it will add the additional content with previous content.



* We want to add khatri in the file "abc.txt", then do append mode "a".

```
main()
```

```
{
```

```
FILE *fp = NULL;
```

```
Char str[50];
```

```
fp = fopen("abc.txt", "a");
```

```
if (fp == NULL)
```

```
{  
    printf("Error");  
    exit(1);
```

```
}
```

```
printf("Enter the content to append:");
scanf("%s", &str);
//or
gets(str);
```

```
fputs(str, fp); → There is no "\n" so it
//or will put in same line of
file
fprintf(fp, "%s", str);
      ↘ new line
//or
for(i=0; i!= '\0'; i++)
{
    fputc(str[i], fp);
}
```

which is a drawback so use %s with "\n"

```
printf("successfully appended");
```

```
fclose(fp);
```

```
}
```

NOTE:-

* Whenever we open the file in append mode; the cursor will be at the last of the existing file

* So that only we put "\n" for new line.

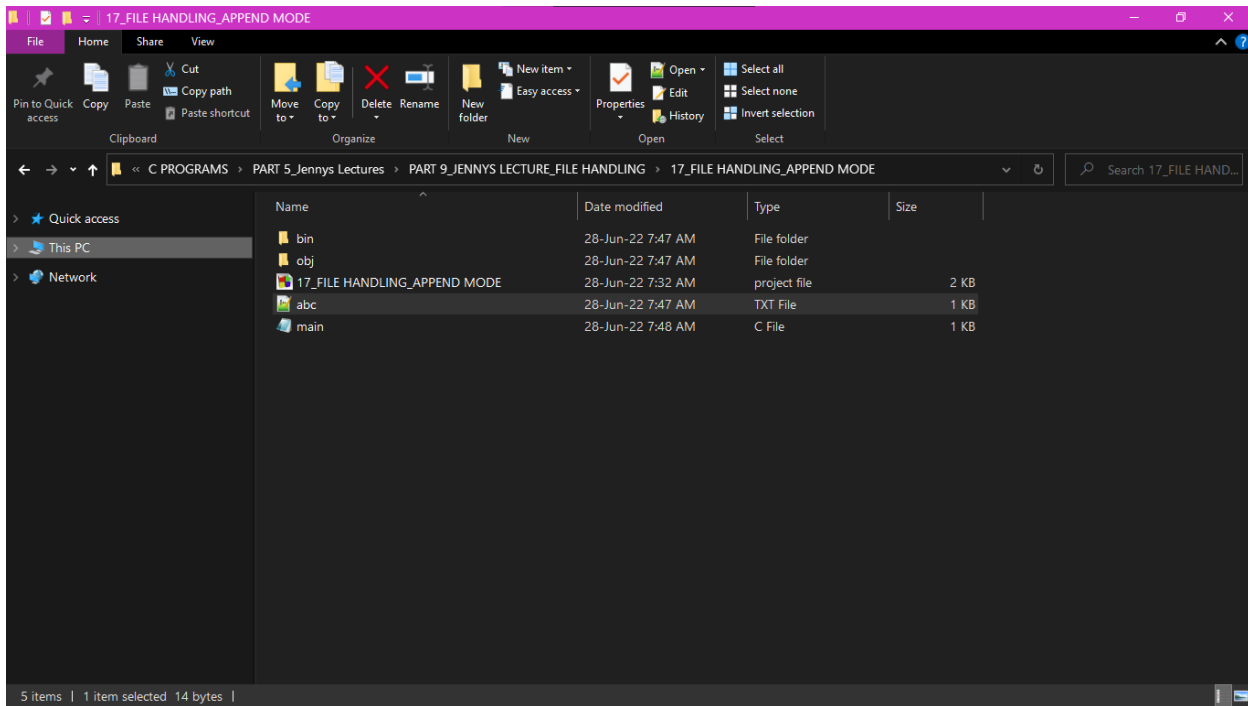
* Like write mode; read mode will also create a file if the file do not exist

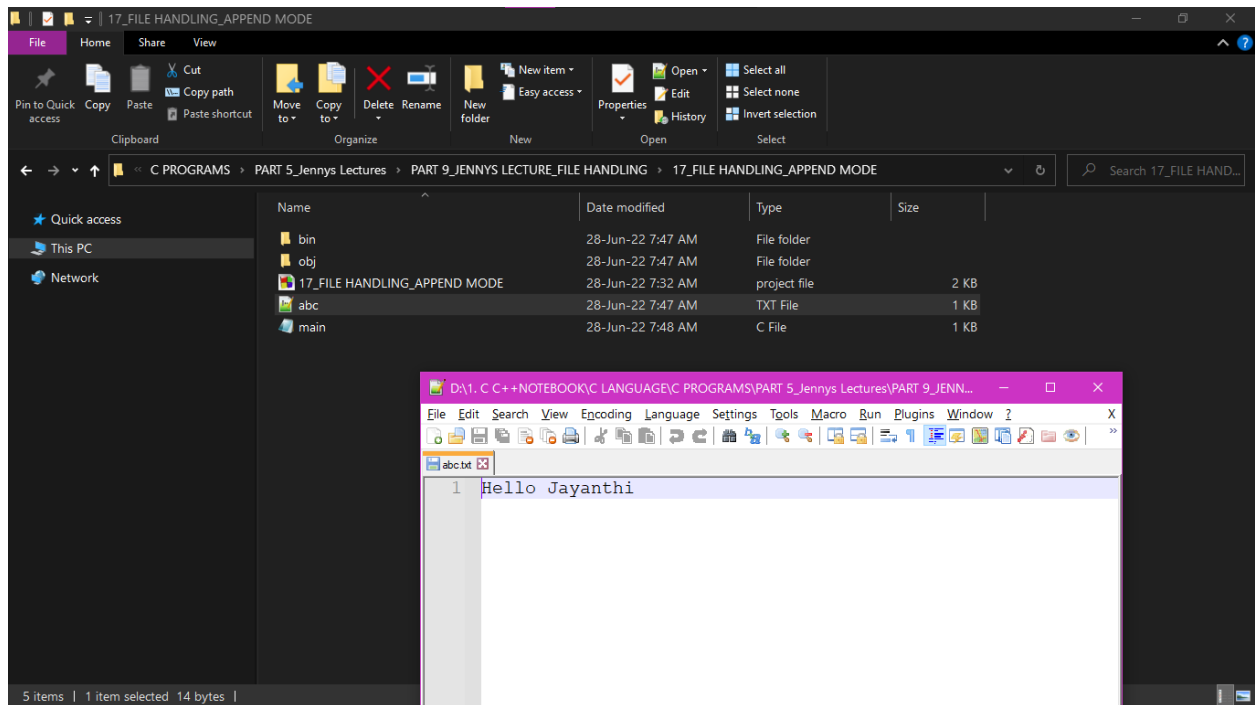
```

3  /** 17-FILE HANDLING-APPEND MODE **/
4  /** Append mode will create a file if file do not exist same as write mode **/
5  /** Append mode will add the content in a file instead of overwriting the file **/
6  int main()
7  {
8
9      FILE *fp=NULL;
10     fp=fopen("abc.txt", "a");
11     if (fp==NULL)
12     {
13         printf("Error");
14         exit(1);
15     }
16     char str[50];
17     printf("Enter the content to append:");
18     gets(str);
19     fprintf(fp, "%s", str);
20     printf("Successfully appended...");
21     fclose(fp);
22
23     return 0;
24 }
25
26

```

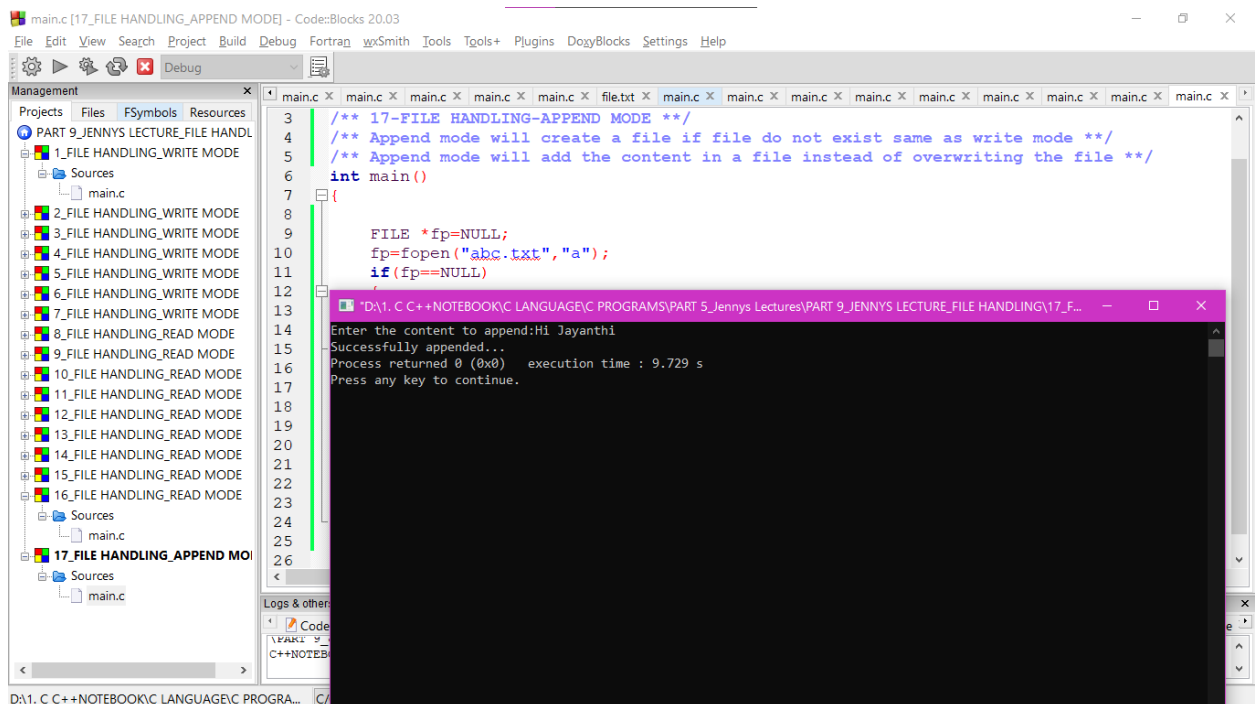
We don't have the file, so it first creates the file and append our content

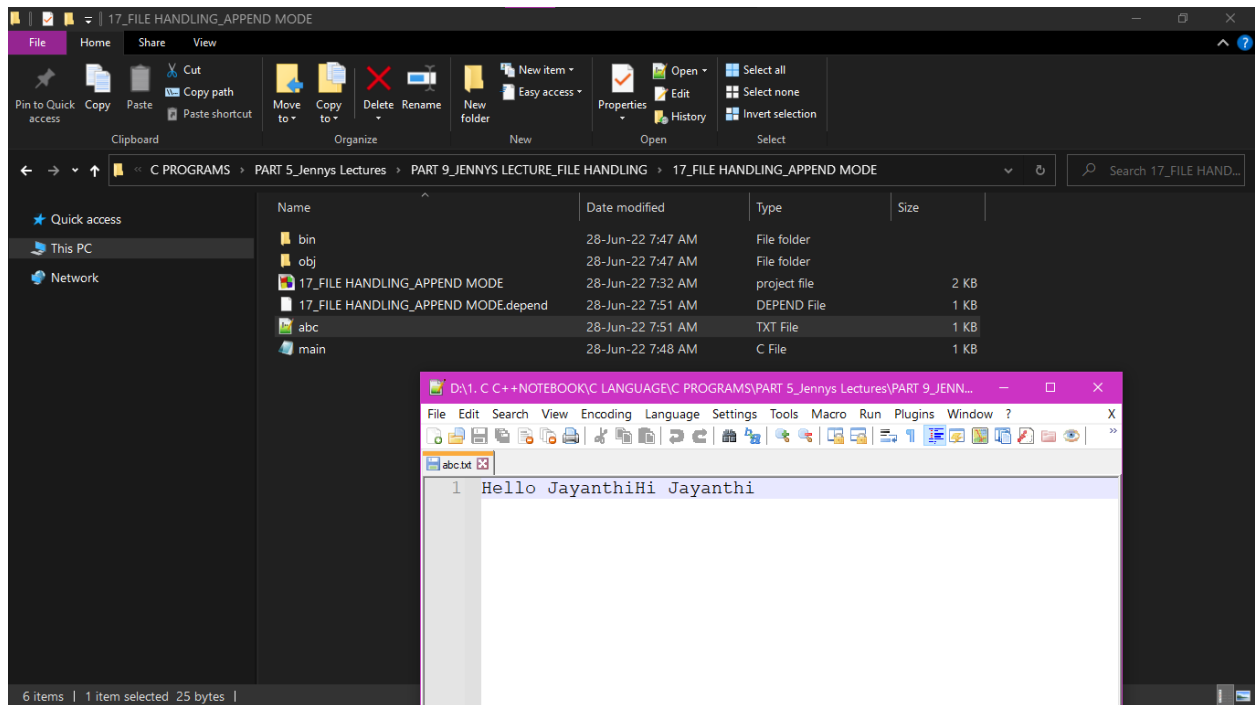




Now again run the same program with the same file name “abc.txt”

Now the file exist and it will append the content with the previously stored content in the file





NOTE:

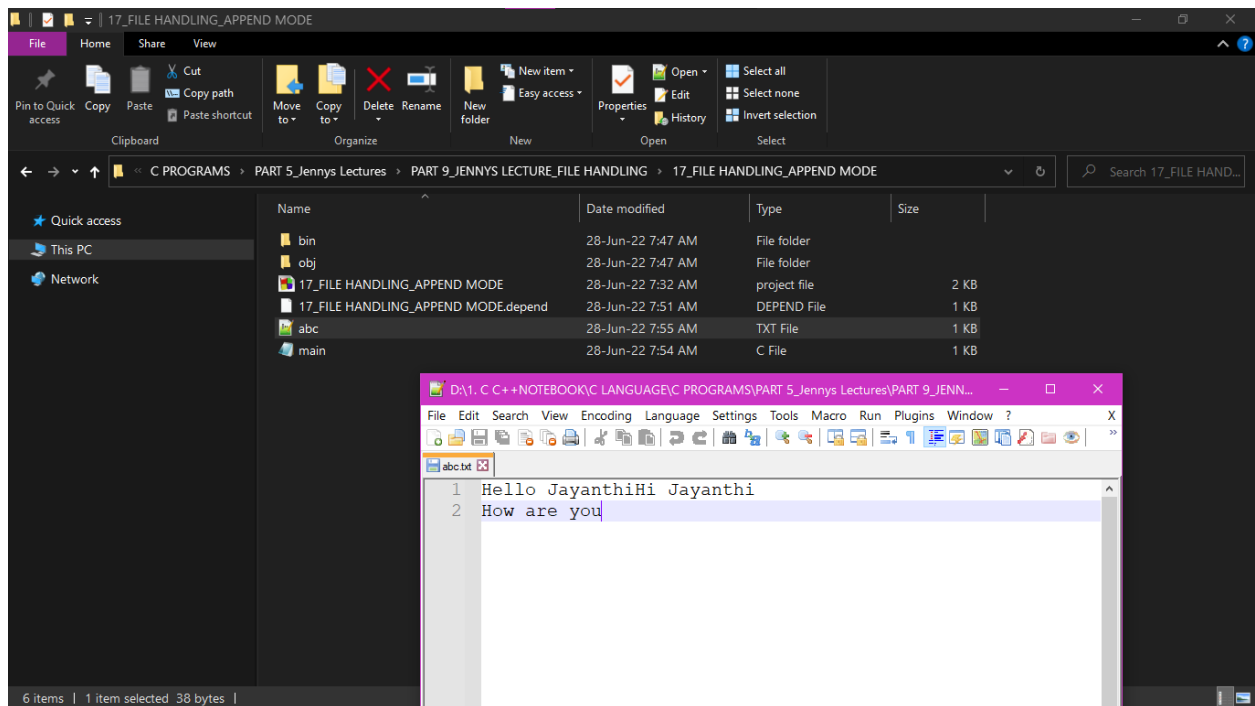
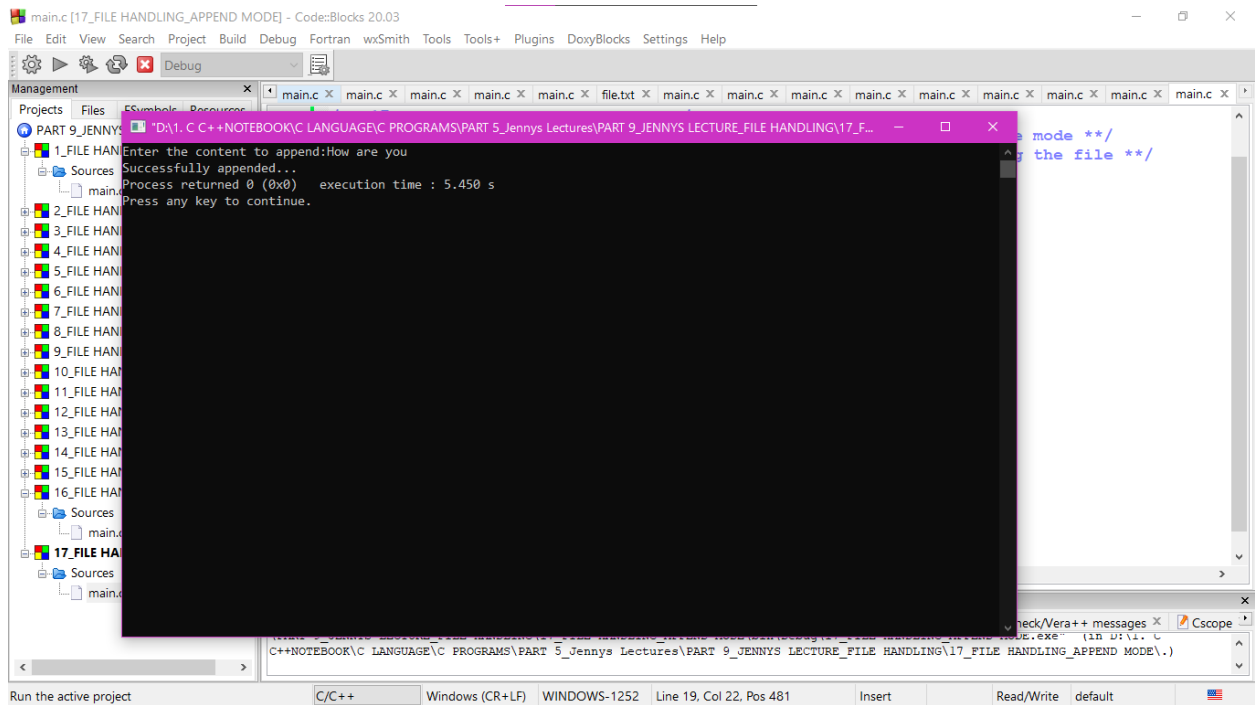
Append mode will append the content from the end of the file.

If we need to append the content in new line use `\n`

```

3  /** 17-FILE HANDLING-APPEND MODE **/
4  /** Append mode will create a file if file do not exist same as write mode **/
5  /** Append mode will add the content in a file instead of overwriting the file **/
6  int main()
7  {
8
9      FILE *fp=NULL;
10     fp=fopen("abc.txt", "a");
11     if (fp==NULL)
12     {
13         printf("Error");
14         exit(1);
15     }
16     char str[50];
17     printf("Enter the content to append:");
18     gets(str);
19     fprintf(fp, "\n%s", str);
20     printf("Successfully appended...");
21     fclose(fp);
22
23     return 0;
24 }
25
26

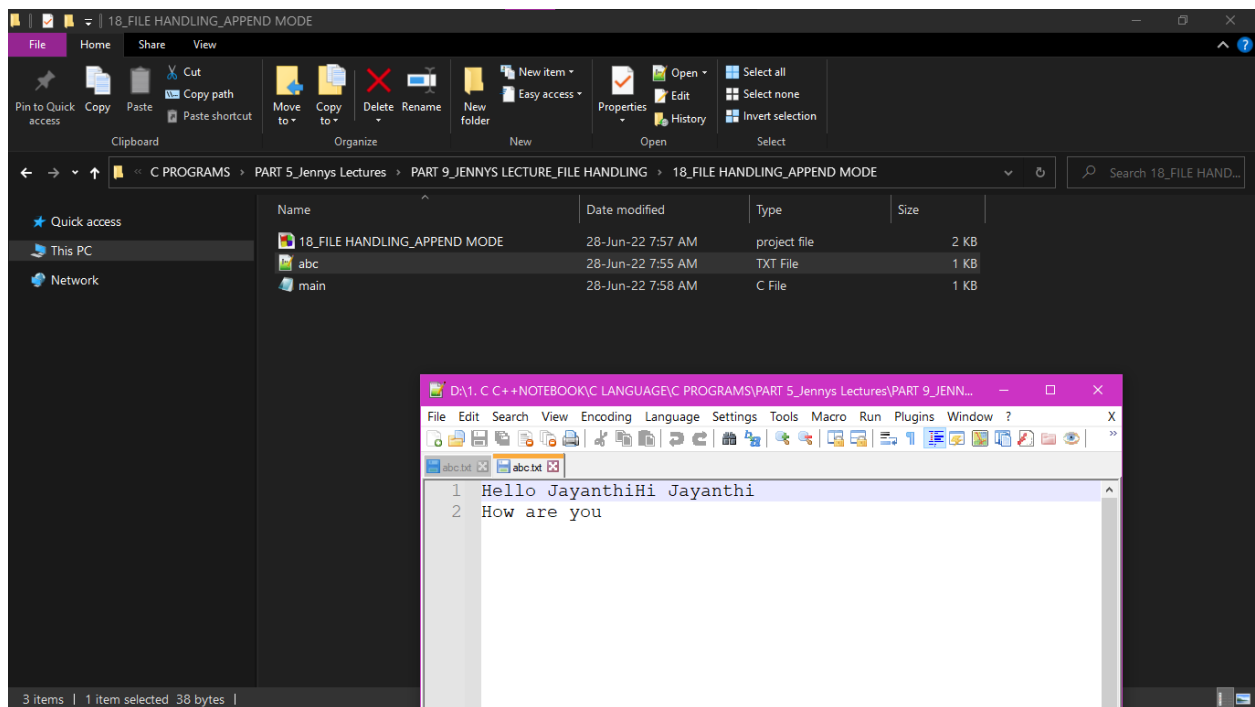
```

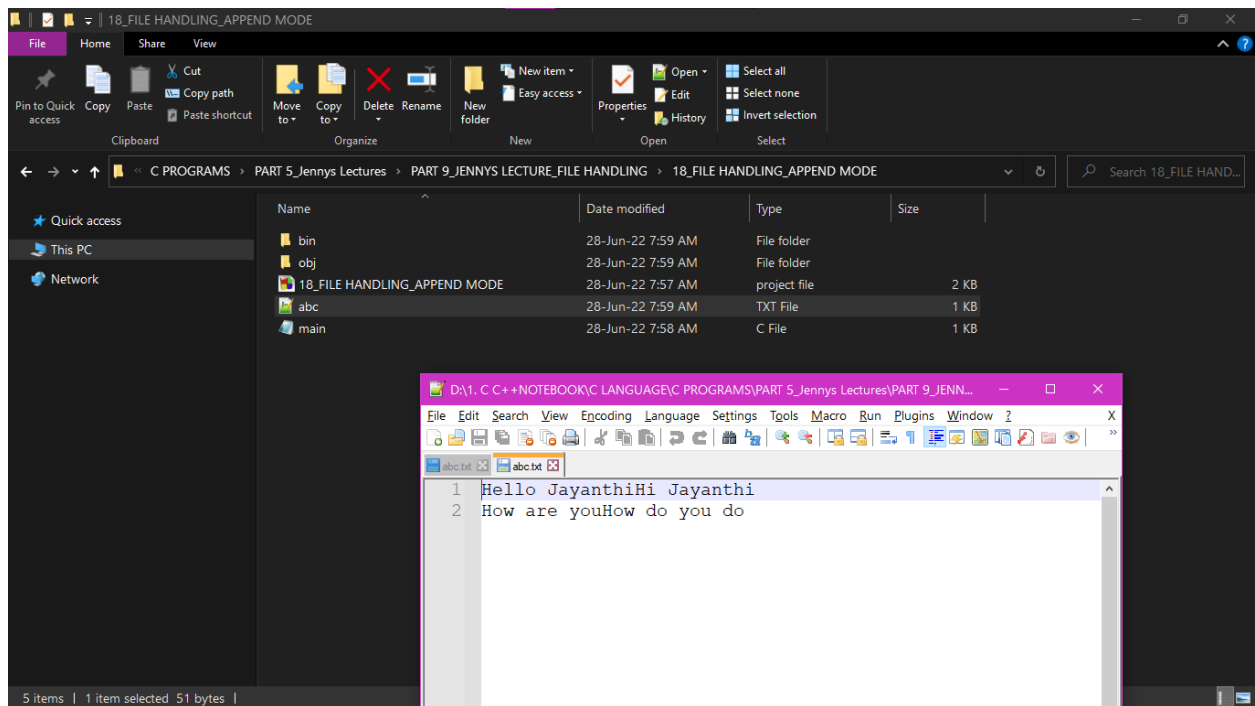
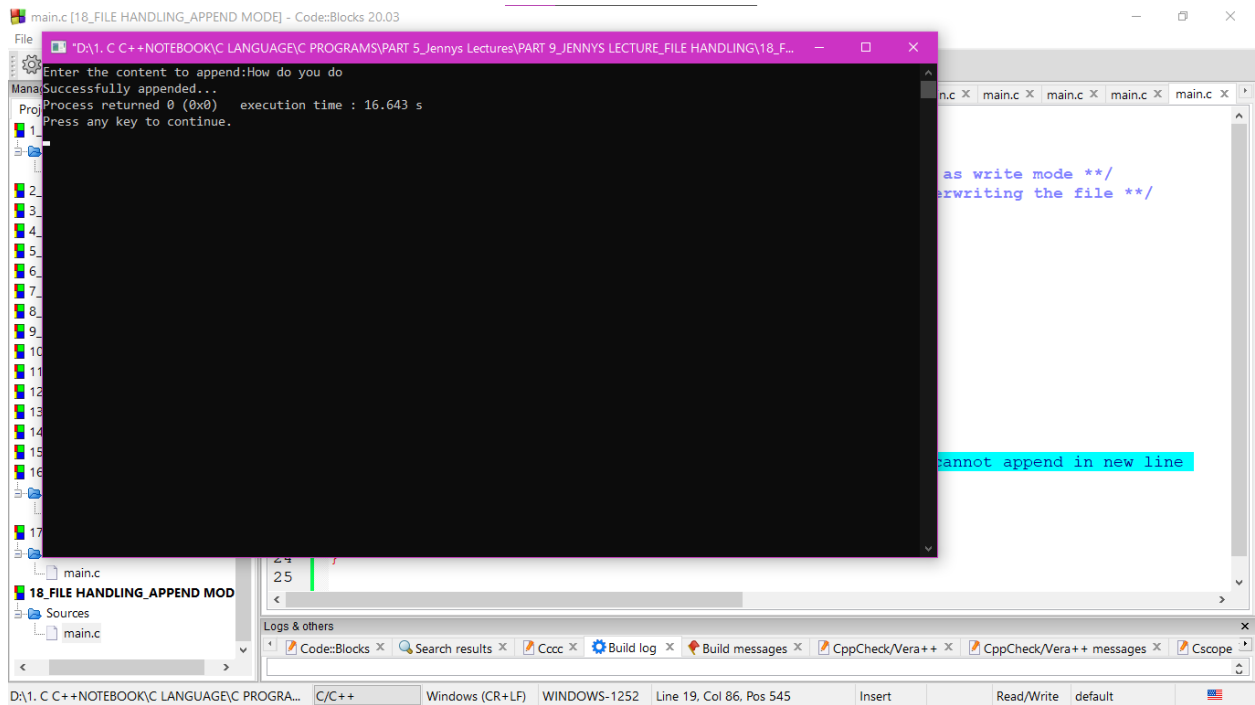


```

1  #include <stdio.h>
2  #include <stdlib.h>
3  /** 18-FILE HANDLING-APPEND MODE **/
4  /** Append mode will create a file if file do not exist same as write mode **/
5  /** Append mode will add the content in a file instead of overwriting the file **/
6  int main()
7  {
8
9      FILE *fp=NULL;
10     fp=fopen("abc.txt","a");
11     if (fp==NULL)
12     {
13         printf("Error");
14         exit(1);
15     }
16     char str[50];
17     printf("Enter the content to append:");
18     gets(str);
19     fputs(str,fp); //using fputs function we can append but cannot append in new line
20     printf("Successfully appended...");
21     fclose(fp);
22
23     return 0;
24 }
25

```






```

3  /** 19-FILE HANDLING-APPEND MODE **/
4  /** Append mode will create a file if file do not exist same as write mode **/
5  /** Append mode will add the content in a file instead of overwriting the file **/
6  int main()
7  {
8      FILE *fp=NULL;
9      fp=fopen("abc.txt","a");
10     if (fp==NULL)
11     {
12         printf("Error");
13         exit(1);
14     }
15     char str[50];
16     printf("Enter the content to append:");
17     gets(str);
18     for(int i=0;str[i]!='\0';i++)
19     {
20         fputc(str[i],fp); //using fputc we can append but cannot append in new line
21     }
22     printf("Successfully appended...");
23     fclose(fp);
24
25     return 0;
26 }
27
28

```

