

[khushanshuparihar-hub / khushanshu](#) Public[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#)[khushanshu / check\\_file1.sh](#) [Raw](#) [khushanshuparihar-hub](#) [hioc](#)

59dddaf · 1 minute ago



85 lines (60 loc) · 1.71 KB

[Code](#) [Blame](#)[Raw](#)    

```
1  #!/bin/bash
2  # check_file.sh
3  # Usage: ./check_file.sh filename.txt
4
5  if [ $# -ne 1 ]; then
6      echo "Usage: $0 <filename>"
7      exit 1
8  fi
9
10 file="$1"
11 if [ -e "$file" ]; then
12     echo "File exists: $file"
13     echo "----- contents -----"
14     cat -- "$file"
15 else
16     echo "File '$file' does not exist."
17     read -p "Create it now? (y/N): " ans
18     case "$ans" in
19         [Yy]*) touch "$file"; echo "Created $file"; echo "You can edit it using your favorite edit"
20         *) echo "Not creating file." ;;
21     esac
22 fi
23
24 ## Line-by-Line Explanation:
25
26 #!/bin/bash
27 → Shebang line, tells the system to use bash to run this script.
28
29 2-3. Comments explaining the script name and usage.
30
31 4-7.
32
33 if [ $# -ne 1 ]; then
34     echo "Usage: $0 <filename>"
35     exit 1
36 fi
~~
```

```
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
```

38  
39   \$# = number of command-line arguments.  
40  
41   If not equal to 1, print usage and exit.  
42  
43   \$0 is the script name itself.  
44  
45   👉 Ensures the user must provide one filename.  
46  
47   file="\$1"  
48  
49   \$1 is the first argument passed (the filename).  
50  
51   Stores it in variable file.  
52  
53   10-14.  
54  
55   if [ -e "\$file" ]; then  
56     echo "File exists: \$file"  
57     echo "----- contents -----"  
58     cat -- "\$file"  
59  
60  
61   -e "\$file" checks if the file exists.  
62   If yes → print confirmation and display its contents using cat.  
63   15-21 (else part):  
64   If the file does not exist:  
65     Inform the user: "File '\$file' does not exist."  
66     Ask: "Create it now? (y/N): "  
67     Use a case statement:  
68     If answer starts with y or Y → touch creates an empty file, then prints confirmation.  
69     Otherwise → "Not creating file."  
70  
71    What this script does  
72  
73   Takes a filename as input.  
74  
75   If file exists → shows its contents.  
76  
77   If file does not exist → asks user whether to create it.  
78  
79   👉 Example run:  
80  
81   \$ ./check\_file.sh notes.txt  
82   File 'notes.txt' does not exist.  
83   Create it now? (y/N): y  
84   Created notes.txt  
85   You can edit it using nano/vi/etc.