|  |  |  |  |
| --- | --- | --- | --- |
| S/N | Operator | Description | Example |
| 1 | -b file | Checks if file is a block special file; if yes, then the condition becomes true. | [ -b $file ] is false. |
| 2 | -c file | Checks if file is a character special file; if yes, then the condition becomes true. | [ -c $file ] is false. |
| 3 | -d file | Checks if file is a directory; if yes, then the condition becomes true. | [ -d $file ] is not true. |
| 4 | -f file | Checks if file is an ordinary file as opposed to a directory or special file; if yes, then the condition becomes true. | [ -f $file ] is true. |
| 5 | -g file | Checks if file has its set group ID (SGID) bit set; if yes, then the condition becomes true. | [ -g $file ] is false. |
| 6 | -k file | Checks if file has its sticky bit set; if yes, then the condition becomes true. | [ -k $file ] is false. |
| 7 | -p file | Checks if file is a named pipe; if yes, then the condition becomes true. | [ -p $file ] is false. |
| 8 | -t file | Checks if file descriptor is open and associated with a terminal; if yes, then the condition becomes true. | [ -t $file ] is false. |
| 9 | -u file | Checks if file has its Set User ID (SUID) bit set; if yes, then the condition becomes true. | [ -u $file ] is false. |
| 10 | -r file | Checks if file is readable; if yes, then the condition becomes true. | [ -r $file ] is true. |
| 11 | -w file | Checks if file is writable; if yes, then the condition becomes true. | [ -w $file ] is true. |
| 12 | -x file | Checks if file is executable; if yes, then the condition becomes true. | [ -x $file ] is true. |
| 13 | -s file | Checks if file has size greater than 0; if yes, then condition becomes true. | [ -s $file ] is true. |
| 14 | -e file | Checks if file exists; is true even if file is a directory but exists. | [ -e $file ] is true. |
| 15 | -eq | Checks if the value of two operands are equal or not; if yes, then the condition becomes true. | [ $a -eq $b ] is not true. |
| 16 | -ne | Checks if the value of two operands are equal or not; if values are not equal, then the condition becomes true. | [ $a -ne $b ] is true. |
| 17 | -gt | Checks if the value of left operand is greater than the value of right operand; if yes, then the condition becomes true. | [ $a -gt $b ] is not true. |
| 18 | -lt | Checks if the value of left operand is less than the value of right operand; if yes, then the condition becomes true. | [ $a -lt $b ] is true. |
| 19 | -ge | Checks if the value of left operand is greater than or equal to the value of right operand; if yes, then the condition becomes true. | [ $a -ge $b ] is not true. |
| 20 | -le | Checks if the value of left operand is less than or equal to the value of right operand; if yes, then the condition becomes true. | [ $a -le $b ] is true. |
| 21 | = | Checks if the value of two operands are equal or not; if yes, then the condition becomes true. | [ $a = $b ] is not true. |
| 22 | ! | This is logical negation. This inverts a true condition into false and vice versa. | [ ! false ] is true. |
| 23 | -o | This is logical OR. If one of the operands is true, then the condition becomes true. | [ $a -lt 20 -o $b -gt 100 ] is true. |
| 24 | -a | This is logical AND. If both the operands are true, then the condition becomes true otherwise false. | [ $a -lt 20 -a $b -gt 100 ] is false. |
| 25 | -z | Checks if the given string operand size is zero; if it is zero length, then it returns true. | [ -z $a ] is not true. |
| 26 | -n | Checks if the given string operand size is non-zero; if it is nonzero length, then it returns true. | [ -n $a ] is not false. |
| 27 | str | Checks if str is not the empty string; if it is empty, then it returns false. | [ $a ] is not false. |
| 28 |  |  |  |
| 29 |  |  |  |
| 30 |  |  |  |
|  |  |  |  |