
Linux™
SHELL
SCRIPTING

Decision Making

1. **if-then-fi**
2. **if-then-else-fi**
3. **if-then-elif-else-fi**
4. **netsted ifs**
5. **case statement**

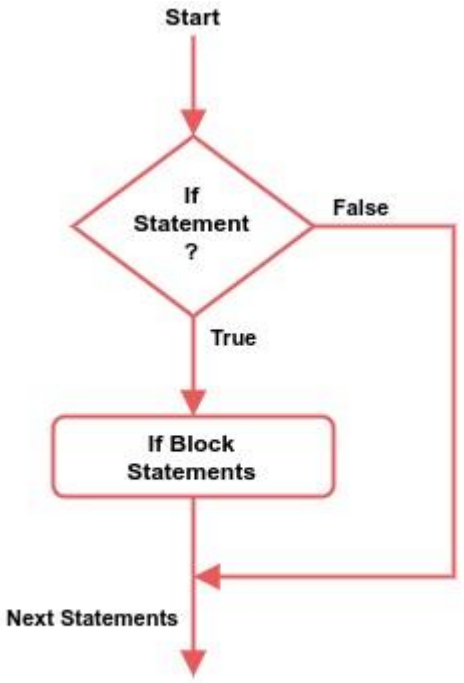
if-then-fi

If control command

then
        command

fi

if-then-fi



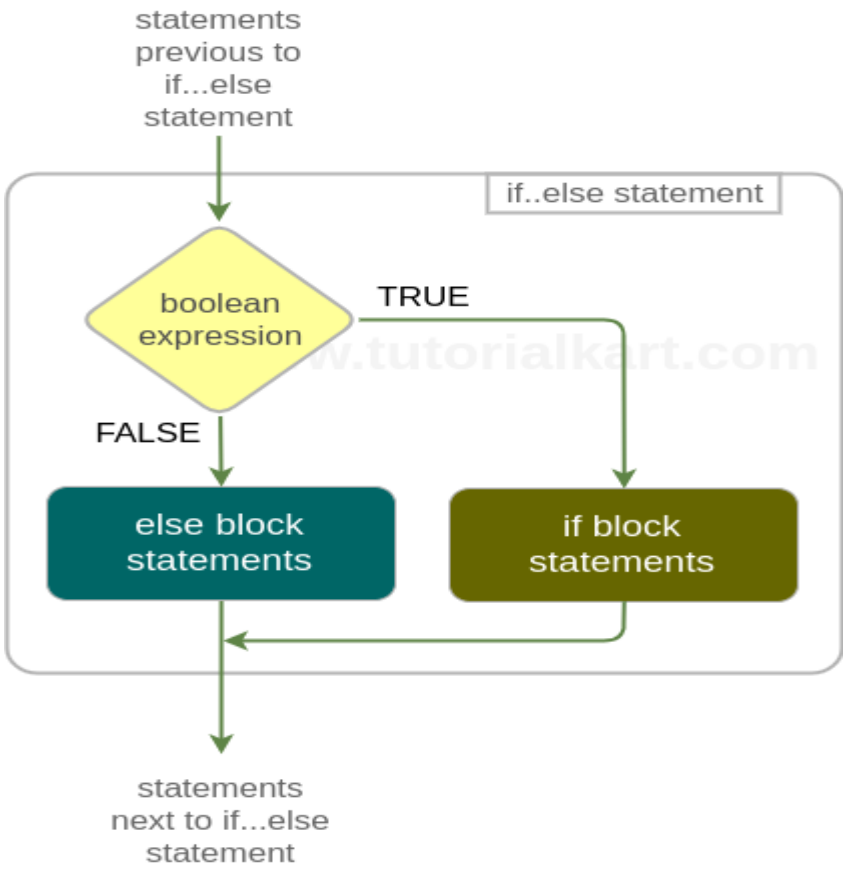
IF Statement Flow Diagram

if-then-fi

```
#!/bin/bash

if [ "$(whoami)" != 'root' ];
then echo "You have no permission to run $0 as
non-root user."
exit 1;
 fi
```

if-then-else-fi



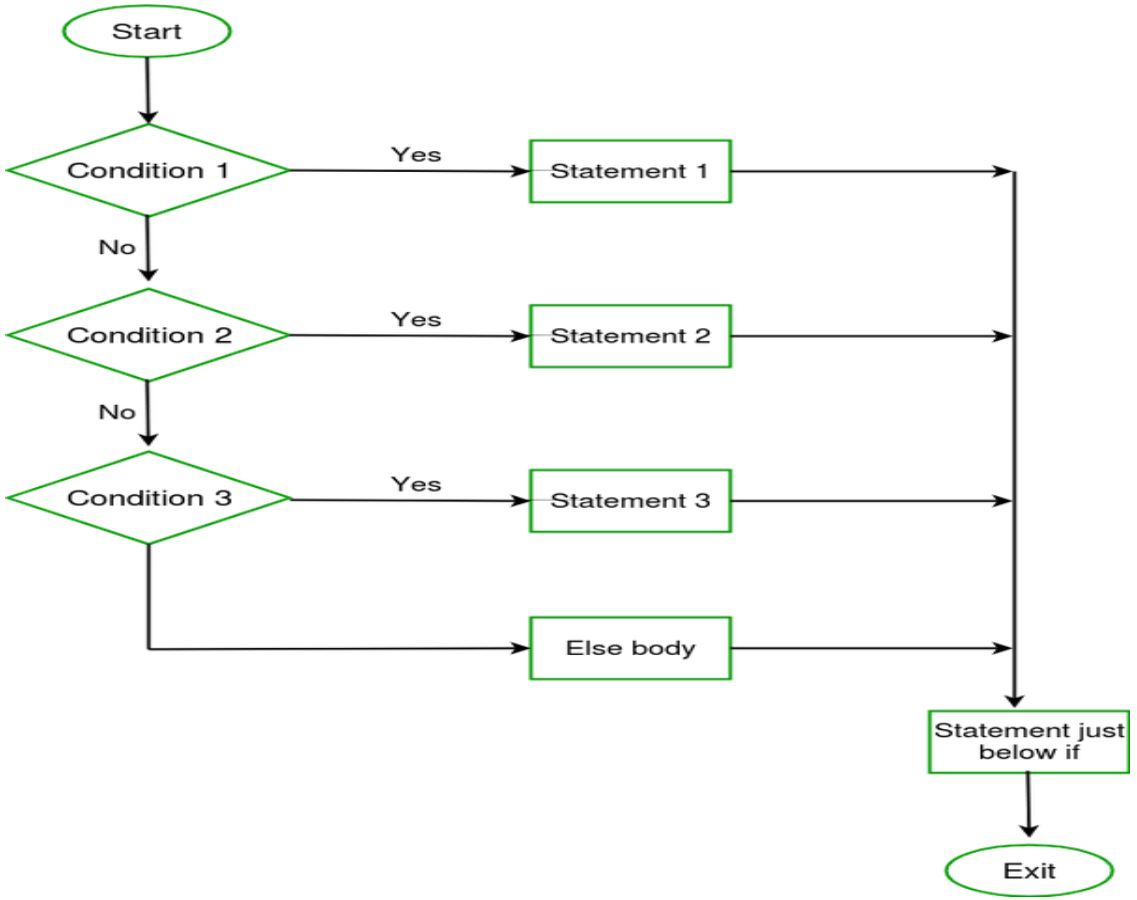
if-then-else-fi

```
if   control command
       command 1
else
       command 2
fi
```

if-then-else-fi

```
#!/bin/bash
echo  enter the number
read num
if [ $num –gt 10 ]
then
        echo “$num is greater than 10”
else
        echo $num is equal or lesser than 10”
fi
```

if-then-elif-else-fi



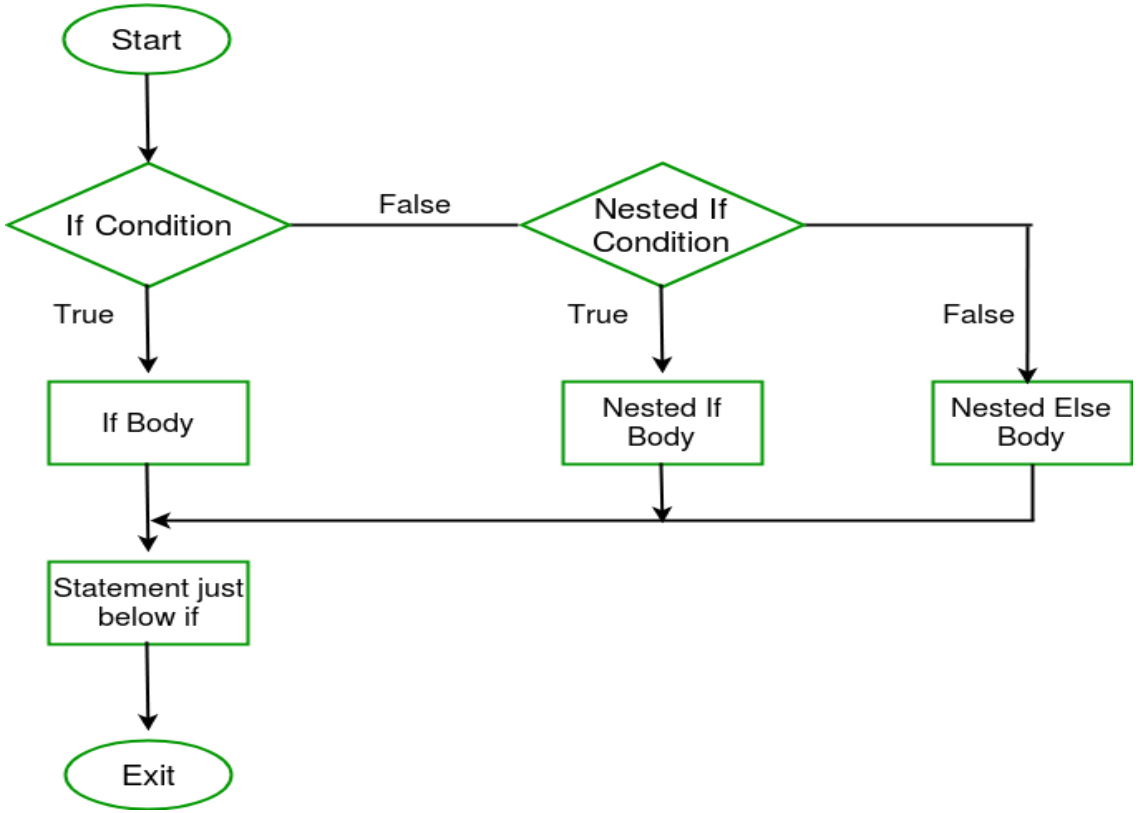
if-then-elif-else-fi

```
if    control command
      command 1
elif
      command 2
else
        command 3
fi
```

if-then-elif-else-fi

```
#!/bin/bash
echo –n “enter the number:”
read num
if [[ $num –gt 10 ]]
then
        echo “$num number is greater than 10”
elif [[ $num –lt 10 ]]
then
        echo “$num number is less than 10”
else
        echo “$num is equal to 10”
fi
```

netsted if


Start
If Condition
False
Nested If Condition
True
True
False
If Body
Nested If Body
Nested Else Body
Statement just below if
Exit

netsted if

```
if   control command
     command 1
       if
              command 2
       fi
else
         command 3
fi
```

netsted if

```
#!/bin/bash
Echo  -n "Enter the first number: "
 read VAR1
Echo  -n "Enter the second number: "
 read VAR2
 echo -n "Enter the third number: "
 VAR3 if [[ $VAR1 -ge $VAR2 ]]
 then
                if [[ $VAR1 -ge $VAR3 ]]
                then
 "$VAR1 is the largest number."
else
 "$VAR3 is the largest number.“
 fi
 else
 if [[ $VAR2 -ge $VAR3 ]] then
 "$VAR2 is the largest number.“
 else
 "$VAR3 is the largest number."
fi
fi
```

If with logical operators

- -a for AND operation
- -o for OR operation
- ! for NOT operation

If with logical operators

```
#!/bin/bash
echo "Enther the first number"
read num1
echo "Enter the second number"
read num2
echo "Enther the third number"
read num3
if [ $num1 -gt $num2 -a $num1 -gt $num3 ]
then
        echo "$num1 is largest number"
       exit
elif [ $num2 -gt $num1 -a $num2 -gt $num3 ]
then
    echo "$num2 is largest number"
       exit
else
     echo "$num3 is largest number"
fi
```

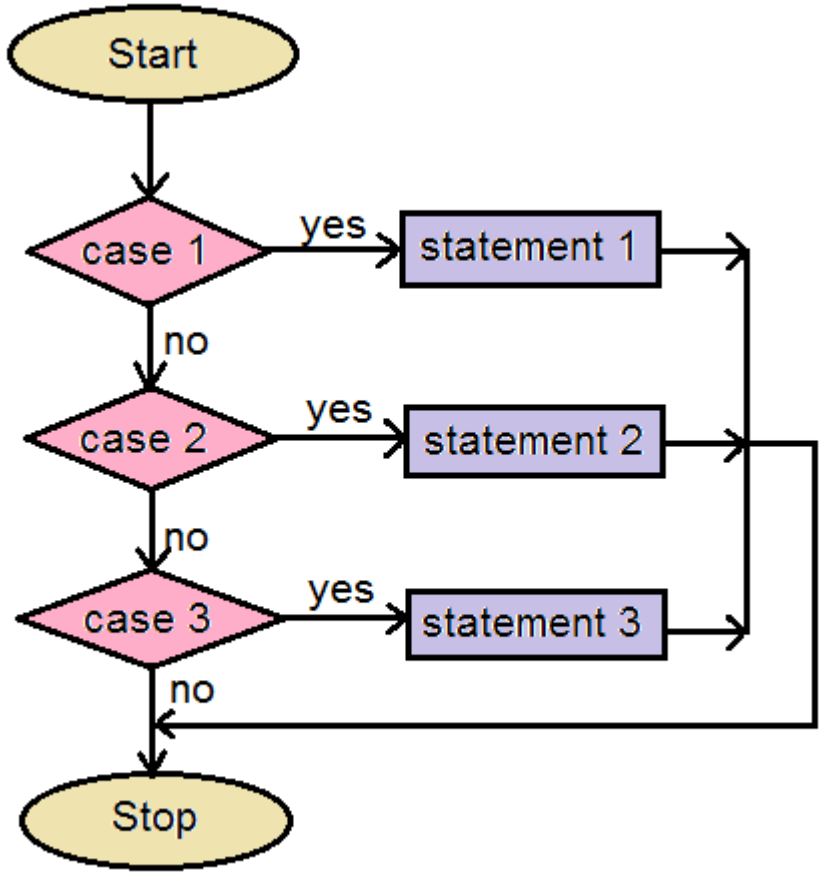
If with “test”, “[ ]”, [[ ]]”

- EXAMPLE: IF… STATEMENT
- **# The following THREE *if*-conditions produce the same result**
- *** DOUBLE SQUARE BRACKETS**
- **read -p "Do you want to continue?" reply**
- **if [[$reply = "y"]]; then**
- **echo "You entered " $reply**
- **fi**
- *** SINGLE SQUARE BRACKETS**
- **read -p "Do you want to continue?" reply**
- **if [$reply = "y"]; then**
- **echo "You entered " $reply**
- **fi**
- *** "TEST" COMMAND**
- **read -p "Do you want to continue?" reply**
- **if test $reply = "y"; then**
- **echo "You entered " $reply**
- **fi**

case statement

```
Case value in
Choice 1)
     command 1
      ;;
Choice 2)
     command 2
      ;;
esac
```

case statement



case statement

```
#!/bin/bash
echo “enter the number”
read  num
case $num in
      1) echo you entered 1 ;;
      2) echo you entered 2 ;;
      *) echo you entered other than 1,2 ;;
esac
```

operator

Operator	Meaning
-gt	Greater than
-lt	Lesser than
-ge	Greater than or equal
-le	Lesser than or equal
-ne	Not equal to
-eq	Equal to

File tests

Operator	Meaning
-s	File exists & size > 0
-f	File exists
-r	File readable
-d	Directory file
-w	File writable
-b	Block file
-x	Executable file
-c	Character file