

Some Important Commands for JAVA Net Beans

Sr. No.	Command	Description
1.	System.exit(0)	to exits the application
2.	Integer.parseInt()	Converts string to Integer
3.	Double.parseDouble()	Converts string to Double
4.	Integer.toString(p)	converts integer value to string
5.	Double.toString(p)	converts double value to string
6.	jTextField3.getText()	to get Input
7.	jTextField1.setText(a + “ “)	to get Output
8.	System.out.println()	Gives Output
9.	jOptionPane.showMessageDialog(“kkkk”)	Use to show any message in Dialog box
10.	pow(a,b)	It computes a^b , where a and b are numbers. e.g. pow(9,3) = 729
11.	round(num1)	It rounds off a given number to its nearest integer. It can take float/double as argument. e.g. round(123.56) = 124 round(123.46) = 123
12.	round(num1, a)	It rounds off a given number upto a places of decimal. e.g. round(123.567,2) = 123.57 round(123.563,2) = 123.56
13.	truncate(num1)	It cuts off decimal part from a given number & removes all decimal values. e.g. round(123.56) = 123 round(123.46) = 123
14.	truncate(num1, a)	It cuts off a given number upto a places of decimal. e.g. round(123.567,2) = 123.56 round(123.563,2) = 123.56
15.	Substring(n);	It will remove starting n characters e.g. str1 = “Kendriya Vidyalaya” str1.Substring(3) = “driya Vidyalaya”
16.	Substring(a,b);	It will give b characters from string starting from a th character e.g. str1 = “Kendriya Vidyalaya” str1.Substring(5,3) = “riy”
17.	Concat()	It will concat 2 strings into 1. e.g. str1 = “Kendriya” str2 = “ Vidyalaya” str1.concat(str2) = “Kendriya Vidyalaya”
18.	toLowerCase()	It converts the string to lowercase letters e.g. str1 = “ Kendriya” toLowerCase(str1) = “ kendriya”
19.	toUpperCase()	It converts the string to uppercase letters e.g. str1 = “ Kendriya” toUpperCase(str1) = “ KENDRIYA”
20.	Trim()	It removes extra blank before & after the characters e.g. str1 = “ Kendriya ” Trim(str1) = “Kendriya”
21.	LTrim()	It removes extra blank before the characters e.g. str1 = “ Kendriya ” LTrim(str1) = “Kendriya ”
22.	RTRIM()	It removes extra blank after the characters e.g. str1 = “ Kendriya ” RTrim(str1) = “ Kendriya”
23.	String.length()	It will calculate the length of string including blank spaces e.g. str1 = “Kendriya Vidyalaya” str1.length() = 16
24.	string.insert(0,”Central”);	Insert given string at given position e.g. str1 = “Kendriya Vidyalaya” str1.insert(0, “aman”) = “aman Kendriya Vidyalaya” e.g. str1 = “Kendriya Vidyalaya” str1.insert(5, “aman”) = “Kendamaniya Vidyalaya”
25.	Max(val1, val2)	Returns Max value
26.	Min(val1, val2)	Returns Min value

27.	Math.sqrt(n)	Returns square root of number e.g. Math.sqrt(144) = 12
28.	Math.abs(n)	Returns positive number e.g. Math.abs(-5) = 5 Math.abs(5) = 5
29.	Sum(val1, val2)	Sum of the values
30.	Avg(val1, val2)	Average of the values
31.	Mod(a,b)	Divides a by b and gives remainder
32.	Ceil(double a)	Returns next whole number. e.g. Math.ceil(1.1) = 2 Math.ceil(-1.1) = -1
33.	Floor(double a)	Returns next integer value
34.	Date();	It returns system date in the given format. Tue Jul 20 17:30:22 GMT+05:30 2010
35.	Now()	It returns system date in the given format. Tue Jul 20 17:30:22 GMT+05:30 2010
36.	jCheckBox1.isSelected()	used to check whether CheckBox is selected or not
37.	jRadioButton.isSelected()	used to check whether RadioButton is selected or not

Some Important Control Structures for JAVA Net Beans

1. SWITCH CASE

```
switch(n){
    case 1 :
        statement1 ;
        break;
    case 2 :
        statement2 ;
        break;
    default :
        statement3;
}
```

2. IF ELSE

```
if(n= =1)
    Statement1 ;
else
    if(n= =2)
        statement 2;
    else
        statement3;
```

3. FOR LOOP

```
int s=0;
for(int a = 0; a<=10; a++)
{
    s = s+a;
}
System.out.println(s);
```

4. WHILE LOOP

```
int s=0, a=0;
while(a<=10)
{
    s = s+a;
    a++;
}
```

```
System.out.println(s);
```

5. DO - WHILE LOOP

```
int s=0, a=0;
do
{
    s = s+a;
    a++;
} while(a<=10);
```

```
System.out.println(s);
```

METHOD TO CONVERT LOOPS

a) IF ELSE TO SWITCH :

1. See the variable used in conditions
2. Take that variable as switch variable
3. Use different values as cases with colon
4. Write the statements of if as it is in case
5. Make last else as default
6. **Don't forget to add break after every case.**

Example : convert if else to switch

```
if(n==1)
    Statement1 ;
else
    if(n==2)
        statement 2;
    else
        statement3;
```

Solution

step 1 : n is used in conditions, use n as switch variable
step 3 : use different values i.e. 1 & 2 as cases with colon
step 4: Write the statements of if as it is in case
step 6: add break after each case
step 5: make last else as default

```
switch(n)
{
    case 1 :
        statement1 ;
        break;
    case 2 :
```

```

statement2 ;

break;

default :

statement3;

}

```

b) SWITCH TO IF ELSE :

1. See the variable used in switch
2. Take that variable as condition variable
3. Use different cases as values to compare
4. Write the statements of case as it is in if
5. Make default as last else

Example : convert switch to if else

```

switch(n)

{

case 1 :

statement1 ;

break;

case 2 :

statement2 ;

break;

default :

statement3;

}

```

Solution

- step 1 : n is used as switch variable, use n in conditions
step 3 : use different cases i.e. 1 & 2 as values with condition
step 4: Write the statements of case as it is in if
step 5: make default as last else

```

if(n= =1)

Statement1 ;

else

if(n= =2)

statement 2;

else

statement3;

```

c) FOR TO WHILE LOOP :

- Step 1: find out the initial, condition & increment part from for loop
Step 2: keep all the statements which are before & after for loop as it is
Step 3: put initial part before the loop
Step 4: put condition word within bracket with while i. e. while(condition)
Step 5: start bracket of while and put all the statements of for condition as it is
Step 6: put increment part as last statement of loop

Example : convert for loop to while loop

```
int s=0;
for(int a = 0; a<=10; a++)
{
    s = s+a;
}
System.out.println(s);
```

Solution

```
int s=0, a=0;
while(a<=10)
{
    s = s+a;
    a++;
}

System.out.println(s);
```

d) FOR TO DO WHILE LOOP :

Step 1: find out the initial, condition & increment part from for loop

Step 2: keep all the statements which are before & after for loop as it is

Step 3: put initial part before the loop

Step 4: write do

Step 5: start bracket of while and put all the statements of for condition as it is

Step 6: put increment part as last statement of loop

Step7: put condition word within bracket with while i. e. while(condition) followed by a semicolon

Example : convert for loop to do while loop

```
int s=0;
for(int a = 0; a<=10; a++)
{
    s = s+a;
}
System.out.println(s);
```

Solution

```
int s=0, a=0;
do
{
    s = s+a;
    a++;
} while(a<=10);

System.out.println(s);
```

e) WHILE TO LOOP FOR:

Step 1: find out the initial, condition & increment part

Step 2: keep all the statements which are before & after for loop as it is

Step 3: put initial part in for loop i.e. for(initial; ;)

Step 4: put condition part in for loop i.e. for(initial; condition ;)

Step 5: put increment part in for loop i.e. for(initial; condition ; increment)

Step6: keep all the statements of the loop as it is

Example : convert for loop to while loop

```
int s=0, a=0;
while(a<=10)
{
    s = s+a;
    a++;
}
System.out.println(s);
```

Solution

```
int s=0;
for(int a = 0; a<=10; a++)
{
    s = s+a;
}
System.out.println(s);
```

f) DO WHILE TO LOOP FOR:

Step 1: find out the initial, condition & increment part
Step 2: keep all the statements which are before & after for loop as it is
Step 3: put initial part in for loop i.e. for(initial; ;)
Step 4: put condition part in for loop i.e. for(initial; condition ;)
Step 5: put increment part in for loop i.e. for(initial; condition ; increment)
Step6: keep all the statements of the loop as it is

Example : convert for loop to while loop

```
int s=0, a=0;
do
{
    s = s+a;
    a++;
} while(a<=10);
System.out.println(s);
```

Solution

```
int s=0;
for(int a = 0; a<=10; a++)
{
    s = s+a;
}
System.out.println(s);
```

g) DO WHILE TO WHILE LOOP:

Step 1: take the condition in place of do and do remember to remove semicolon i.e. ;
Step 2: keep all the statements of the loop as it is

Example : convert for do while loop to while loop

```
int s=0, a=0;
do
{
    s = s+a;
    a++;
} while(a<=10);
System.out.println(s);
```

Solution

```
int s=0, a=0;
while(a<=10)
{
    s = s+a;
    a++;
}
System.out.println(s);
```

h) WHILE TO DO WHILE LOOP:

Step 1: take the condition after the loop and do remember to add semicolon i.e. ;
Step 3: add do before the loop
Step 2: keep all the statements of the loop as it is

Example : convert while loop to do while loop

```
int s=0, a=0;
while(a<=10)
{
s = s+a;
a++;
}
System.out.println(s);
```

Solution

```
int s=0, a=0;
do
{
s = s+a;
a++;
} while(a<=10);

System.out.println(s);
```

For finding OUTPUT Questions

1. Make columns with variables in rough.
2. Move stepwise & note down the changes in columns you had made.
3. Write the final answer from columns.
4. Be careful not need to hurry, you have ample time in paper.

Example 1 :

What will be the value of a and b after execution of following code :

```
int a = 1 , b = 2 ;
if ( ++b < 5 )
a * = b ;
```

Solution : 1. make 2 columns a & b in rough

2. give value 1 & 2 in a & b respectively
3. move to next step, ++b means b is incremented first so now, b = 3. Compare it with 5 condition is true
4. move to next step, a = a * b i.e. a= 1 * 3 = 3. Put 3 in column for a
5. as no more statement, the final answers are a = 3 and b = 3.

Example 2 :

What will be displayed in jTextField1 after executing the following code ?

```
int m = 16 ;
m = m + 1 ;
if ( m < 15 )
jTextField1. setText ( Integer . toString ( m ) ) ;
else
jTextField1. setText ( Integer . toString ( m + 15 ) ) ;
```

Solution : 1. make 2 columns textfield & m in rough

2. give value 16 to m
3. move to next step, m is incremented so now, m = 17. Compare it with 15 condition is false
4. move to next step, jTextField1. setText (Integer . toString (m + 15)) ; Put 17 + 15 = 32 in column for textfield
5. as no more statement, the final answer is 32.