DAY-12

------

5.SHIFT OPERATOR

----------------

Shift Operator is classified into two types:

1.Right Shift Operator (>>)

2.Left Shift Operator (<<)

example for Right Shift Operator

--------------------------------

ShiftRight.java

---------------

class ShiftRight

{

public static void main(String[] args)

{

int a = 10;

int b = 2;

int res;

res = a>>b;

System.out.println(res); o/p --> 2

}

}

explanation refer the diagram.

example for Left Shift Operator

--------------------------------

ShiftRight.java

---------------

class ShiftRight

{

public static void main(String[] args)

{

int a = 10;

int b = 2;

int res;

res = a<<b;

System.out.println(res); o/p --> 40

}

}

explanation refer the diagram.

---------------------------------------------------------------------------------

6.SHORT CIRCUIT/LOGICAL OPERATOR.

---------------------------------

example:1 [bitwise operator]

----------------------------

ShortCircuit1.java

------------------

class ShortCircuit1

{

public static void main(String[] args)

{

int a = 5;

int b = 7;

if((a == 6) & (++b == 7))

{

System.out.println(a);

System.out.println(b);

}

else

{

System.out.println(a); o/p--> 5

System.out.println(b); o/p--> 8

}

}

}

NOTE: ShortCircuit Operator is used to increse the performance(speed of execution) of the program.

example:1.1 [ShortCircuit operator]

----------------------------

ShortCircuit1.java

------------------

class ShortCircuit1

{

public static void main(String[] args)

{

int a = 5;

int b = 7;

if((a == 6) && (++b == 7))

{

System.out.println(a);

System.out.println(b);

}

else

{

System.out.println(a); o/p--> 5

System.out.println(b); o/p--> 7

}

}

}

--------------------------------------------------------------

example:2 [bitwise operator]

----------------------------

ShortCircuit1.java

------------------

class ShortCircuit1

{

public static void main(String[] args)

{

int a = 5;

int b = 7;

if((a == 6) | (++b == 7))

{

System.out.println(a);

System.out.println(b);

}

else

{

System.out.println(a); o/p--> 5

System.out.println(b); o/p--> 8

}

}

}

example:2.1 [shortcircuit operator]

----------------------------

ShortCircuit1.java

------------------

class ShortCircuit1

{

public static void main(String[] args)

{

int a = 5;

int b = 7;

if((a == 5) || (++b == 7))

{

System.out.println(a);

System.out.println(b);

}

else

{

System.out.println(a); o/p--> 5

System.out.println(b); o/p--> 7

}

}

}

NOTE:

&& : if exp1 outcome is zero then second exp will not be executed.

|| : if exp1 outcome is one then second exp will not be executed.

-----------------------------------------------------------------------------------------------------------------------------------------------------

TYPE CASTING

-------------

Converting one datatype into another is called as type casting.

It has been classified into two types:

1. IMPLICIT type casting

2. EXPLICIT type casting

1.IMPLICIT type casting: converting lower datatype to higher datatype is called as implicit type casting.

ex:1

----

TypeCast.java

-------------

class TypeCast

{

public static void main(String[] args)

{

byte b = 12;

int i;

i = b;

System.out.println(i); o/p --> 12

}

}

we can convert byte to int datatype [implicit typecasting]

ex:2

----

TypeCast.java

-------------

class TypeCast

{

public static void main(String[] args)

{

short s = 143;

int i;

i = s;

System.out.println(i); o/p --> 143

}

}

we can convert short to int datatype [implicit typecasting]

ex:3

----

TypeCast.java

-------------

class TypeCast

{

public static void main(String[] args)

{

long l = 35l;

float f;

f = l;

System.out.println(f); o/p --> 35.0

}

}

we can convert long to float datatype [implicit typecasting]

ex:4

----

TypeCast.java

-------------

class TypeCast

{

public static void main(String[] args)

{

char ch = '?';

int i;

i = ch;

System.out.println(i); o/p --> 63 [unicode value]

}

}

we can convert char to int datatype [implicit typecasting]

ex:5

----

TypeCast.java

-------------

class TypeCast

{

public static void main(String[] args)

{

double d = 69.5;

byte b;

b = d;

System.out.println(b); o/p--> error: incompatible types: possible lossy conversion from double to byte b = d;

}

}

EXPLICIT TYPE CASTING

---------------------

Converting the Higher datattype to lower datatype is called as Explicit typecasting.

ex:1

----

TypeCast.java

-------------

class TypeCast

{

public static void main(String[] args)

{

double d = 69.5;

byte b;

b = (byte)d;

System.out.println(b); //

}

}

NOTE: loss of data will not occur in case of implicit type casting but loss of data will occur in explicit type casting

we can convert double to byte datatype [explicit typecasting]

ex:2

----

TypeCast.java

-------------

class TypeCast

{

public static void main(String[] args)

{

int i = 129;

byte b;

b =(byte)i;

System.out.println(b); o/p --> -127 [for explanation refer diagram]

}

}

---------------------------------------------------------------------------------------------------

e