**Lecture 2 : Getting Started with Java**

1. Primitive data type 'long' is having size 8 byte(s) in Java Programming.

#### Which of the following data type stores longest decimal number ?

#### Double

#### Only float and double can hold decimal numbers. Size of the float is 4 bytes and double is 8 bytes (in most of the compilers, as the size of data types is compiler specific).

#### Which of these values can be assigned to a boolean variable in Java?

#### true or false Boolean variable in java can only hold either true or false value.

#### Which of these is a valid variable name ?

#### Var1,var\_1

#### n java, you cannot start your variable name with numbers and also they cannot contain any other special character except underscore (\_) and dollar ($).

#### Compiler never assigns a default value to an uninitialized local variable in Java Programming. Whether this statement is true or false ?

#### true

#### In java, its compulsory to initialise any local variable before using it because compiler don't assign any default/garbage value to variables.

#### What is the output of the following code if the input is : 5 10 ?

Scanner s = new Scanner(System.in);

int a = s.nextInt();

int b = s.nextInt();

System.out.println(a+b);

15

"s.nextInt()" scans and returns the next token as int. A token is part of entered line that is separated from other tokens by space, tab or newline. So when input line is : “5 10”, then s.nextInt() returns the first token i.e. “5” as int and s.nextInt() again returns the next token i.e. “10” as int

#### What is the output of the following code if the input string is "Coding Ninjas"?

Scanner s = new Scanner(System.in);

String str;

str = s.next();

System.out.print(str);

Coding

"s.next()" returns the next token as String. A token is part of entered line that is separated from other tokens by space, tab or newline. So when input line is - “Coding Ninjas” then s.next() returns the first token i.e. “Coding”

#### What is the output of the following code if input is :10 abc def

Scanner s = new Scanner(System.in);

int a = s.nextInt();

String str = s.next();

System.out.print(a);

System.out.println(str);

10abc

"s.nextInt()" scans and returns the next token as int. A token is part of entered line that is separated from other tokens by space, tab or newline. So when input line is - “10 abc def” then s.nextInt() returns the first token as int i.e. “10” and s.next() returns the next token "abc”. While printing, in first statement a is printed and then str. There is no space or nextline between both print. Hence output is : 10abc.

#### What is the output of the following code if input is : abc def 10

Scanner s = new Scanner(System.in);

String str = s.next();

int a = s.nextInt();

System.out.print(str + " " + a);

InputMismatchException.

"s.next()" scans and returns the next token as String. A token is part of entered line that is separated from other tokens by space, tab or newline. So when input line is - “abc def 10” then s.next() returns the first token as String i.e. “abc” and s.nextInt() tries to convert the next token i.e. “def” into an int, which gives InputMismatchException.

**Lecture 3 : How is Data Stored?**

#### Short data type has a minimum value of -2^15 = -32,768

Short is of 16 bits. So the smallest value short can store is : -2^15.

#### Range of ****byte**** data type is ?

#### -128 to 127 Range of byte(8 bits) is : -2^7 to 2^7-1

#### Will following statement give an error ?

float f = 1.4;

Yes

System treats all decimal numbers as double by default. So 1.4 is stored as double(which is of 8 bytes). When we will try put a double into a float(which is of 4 bytes), it will give you error.

#### What will be the output of the following statement ?

System.out.println('a' + 1);

98

When you add a character and an int, it will add the ASCII value of char ‘a’ i.e 97 and int 1. So ans will be 98.

#### What will be the output ?

int i = 'c';

System.out.println(i);

99 When we put char ‘c’ into an int, its ASCII value will be put in the int i.e. 99.

#### Automatic type conversion in Java takes place when :

#### Two type are compatible and size of destination type is larger than source type.

#### Lecture 4 : Conditionals and Loops

#### What is the error in this code?

byte b = 50;

b = b \* 50;

operator has converted b \* 50 into int, which can not be converted to byte without casting

b is multiplied with 50 where b is byte variable and 50 is an integer. So after the multiplication, result comes as an integer value which is 2500. Now we are trying to assign this integer to a byte variable, which will cause the error

1. public class Solution{

public static void main(String [] args) {

double a = 6 / 4;

int b = 6 / 4;

double c = a + b;

System.out.println(c);

}

}

2.0

When 6 / 4 is performed, both the operands of / are integer. Hence answer will be an int i.e. 1. When we store it in a (which is double), value of a will be 1.0 and value of b will be 1. Thus a + b will be 2.0.

1. public class Solution{

public static void main(String [] args) {

double a = 55.5;

int b = 55;

a = a % 10;

b = b % 10;

System.out.println(a + " " + b);

}

}

5.5 5

% operator gives remainder. So a % 10 will give us 5.5 and b % 10 will give us 5. Hence output is : 5.5 5

1. public class Solution {

public static void main(String [] args) {

int var1 = 5;

int var2 = 6;

System.out.print(var1 > var2);

}

}

false.

>is a relational operator. So it will give the result as true or false only. var1 is not greater than var2, hence result is false.

#### Find the output of the following code:

public static void main(String args[])

{

int a=10,b=15;

if(a>b)

{

System.out.print("a ");

}

else

{

System.out.print("b ");

}

System.out.print("is greater");

}

b is greater "is greater" is written outside if-else so it would always print

#### Find the output of the following code:

int a=50;

if(a>10)

{

System.out.print("Coding");

}

else(a>20)

{

System.out.print("Ninjas");

}

Compile time error else (a>20) is wrong syntax. We cannot use condition after else.

1. Find the output

public static void main(String args[])

{

int x = 5;

if (x < 6)

System.out.print("Hello ");

if(x == 5){

System.out.print("Hi ");

}

else{

System.out.print("Hey ");

}

}

Hello Hi

As x is equal to 5, it enters in first if condition and prints "Hello". After that, second if condition will be checked and that is true again, so it will next print "Hi". Else part will be skipped.

1. Figure out the output

public static void main(String[] args) {

int x = 15;

if(x <= 15){

System.out.print("Inside if ");

}else if(x == 15){

System.out.print("Inside else if ");

}

System.out.println(x);

}

Inside if 15

#First if condition will be checked first, and it evaluates to true. Hence, the statement inside if will be executed, so it will print "Inside if ".

#After that, else if part will be skipped. Because in if-else statements, once a condition is satisfied remaining all the conditions are skipped without evaluation.

#After exit from if-else statement, the last print statement will be executed. Hence it will next print value of x i.e. 15.

#So the output is : "Inside if 15"

1. public static void main(String args[])

{

int var1 = 5;

int var2 = 6;

if ((var2 = 1) == var1)

System.out.print(var2);

else

System.out.print(var2 + 1);

}

2

#Inside if condition, we are actually assigning 1 to var2 and then comparing it with var1. So, after assignment, var2 becomes equal to 1 and then we are comparing it with var1 whose value is 5. Both are not equal, hence else part will be executed.

#So the output will be 2, as var2 was updated to value 1 in the if condition

#### Find the output of the following code:

public static void main (String[] args) {

int a=50;

int b=Integer.MIN\_VALUE;

if(a/0==b)

{

System.out.println("Hello");

}

else

{

System.out.println("Hi");

}

}

run time error

The syntax of the code is correct but on execution dividing a number by zero gives a run time error.

#### Let a and b are the two integers. Which option can be used to check out of two numbers one is positive and the other is negative

#### a\*b<0

#### Given two integers a and b, the product of two integers is negative means either of a or b is negative.i.e, If we multiply a\*b and the result is less than 0 , it means either a or b is negative

#### Find the output of the code :

public static void main (String[] args) {

int a=50,b=20;

if(a>b)

{

if(a>100)

System.out.println("Ace");

if(b<100)

b=50;

}

else if(a==b)

{

System.out.println("King");

}

else

{

System.out.println("Queen");

}

}

no output

(a>b) is true,So we go inside the if. (a>100) is false. (b<100) is true,so it is executed and b is updated to 50. But no print statement is executed hence no output.

#### What will be the output of the following code:

public static void main (String[] args) {

int i=0;

while(i<10)

{

i=i+1;

System.out.print(i);

i=i+1;

}

}

#### 13579

#### The number of Hello printed on the screen for the following code will be:

public static void main (String[] args) {

int x=5;

int y=5;

while((x=5)==y)

{

System.out.println("Hello");

x++;

y++;

}

}

#### One The loop is executed only once when y=5.The condition is false when y=6.

#### The number of Hello printed on the screen for the following code will be:

public static void main (String[] args) {

int x=5;

int y=5;

while(x==y)

{

System.out.println("Hello");

x++;

y++;

}

}

Infinite x and y are equal every time . Hence infinite loop.

#### Whichof the following codes gives same output:

1

int i=1;

while(i<5)

{

System.out.print(2\*i);

i=i+1;

}

#### 2

int i=2;

while(i<10)

{

System.out.print(i);

i=i+2;

}

#### 3

int i=2;

while(i<10)

{

System.out.print(i);

i\*=2;

}

#### 4

int i=10;

while(i>0)

{

if(i%2==0)

{

System.out.print(10-i);

}

i--;

}

12Output of the codes are:Code 1: 2468 Code 2: 2468 Code 3: 248 Code 4: 02468

#### Can this code be used to check primality of a positive integer:

public static void main (String[] args) {

Scanner s=new Scanner (System.in);

int n=s.nextInt();

boolean isprime=true;

if(n%2==0)

isprime=false;

int i=3;

while(isprime&&i<n){

isprime=!(n%i==0);

i+=2;

}

if(isprime){

System.out.println("Prime");

}else{

System.out.println("Composite");

}

}

No The code fails for n=2. Otherwise the code runs fine for every other positive integer.

#### Find the output for the following code:

int i=10;

while((i=i-1)>0) {

System.out.print(i);

if(i%5==0)

return;

}

98765

The code executes for i=9,8,7,6,5 and prints 98765.At i=5 the if statement is executed and the code terminates.

#### Will following code generate error ?

public class Main {

public static void main(String[] args) {

int a = 10;

if(a > 5) {

int b = 10;

}

System.out.println(b);

}

}

yes

#### Will following code generate error ?

public class Main {

public static void main(String[] args) {

int a = 10;

if(a > 5) {

a = 100;

}

System.out.println(a);

}

}

No

#### Will following code generate error ?

public class Main {

public static void main(String[] args) {

int a = 10;

if(a > 5) {

int a = 100;

}

System.out.println(a);

}

}

Yes

#### Will following code generate error ?

public class Main {

public static void main(String[] args) {

for(int i = 0; i < 3; i++) {

System.out.print(i + " ");

}

System.out.print(i + " ");

}

}

Yes

#### What is the output ?

public class Main {

public static void main(String[] args) {

int a = 10;

while(a > 5) {

int b = 1;

System.out.print(b + " ");

a--;

}

}

}

1 1 1 1 1

**Lecture 7 : Operators & For Loop**

1. Choose the correct output for the following code:

public static void main (String[] args) {

int a=10,b=50;

a++;

--b;

int c=a--+b--;

System.out.print(++c);

}

61 a++,a=11. --b,b=49. c=a--+b--=11+49=60 a=10 b=48 print(++c)->print(61)

#### Which line(s) of the following code would give an error:

public static void main (String[] args) {

int a=10,b=20;

System.out.println(a+++--b);//line 1

System.out.println(a--+++b);//line 2

System.out.println(a++-++b);//line 3

System.out.println(a+++++b);//line 4

}

Line2 and line 4

As + operator and ++ operator have equal precedence.Similarly - and -- have equal precedence.So in line 2 +++b and in line 4 +++b gives an error.To solve this error we can use +(++b).

#### Choose the correct output of the following code:

public static void main (String[] args) {

int a=10,b=20;

int c=a&b;

System.out.print(c);

int d=a|b;

System.out.print(d);

int e=a^b;

System.out.print(e);

int f=c+d+e;

System.out.print(~f);

}

03030-61 c=10&20=0. d=10|20=30. e=10^20=30. f=0+30+30=60. ~f=-61.

#### Choose the correct output for the given code:

public static void main (String[] args) {

int a=10,b=-20;

System.out.print(a^b);

}

-26

#### What will be the output ?

public static void main(String args[]){

int a = 42;

int b = ~a; (there is tilde sign before a)

System.out.print(a + " " + b);

}

42 -43

1. class Output {

public static void main(String args[]) {

int x , y = 1;

x = 10;

if (x != 10 && x / 0 == 0)

System.out.println(y);

else

System.out.println(++y);

}

}

2

1. class Solution {

public static void main(String args[]) {

int x = 15;

int y = x++;

int z = ++x;

System.out.println(y +" " + z);

}

}

15 17

1. class Solution {

public static void main(String args[]) {

int g = 3;

System.out.print(++g \* 8);

}

}

#### 32

1. class Solution {

public static void main(String args[]) {

int x =10;

int y = 20;

if(x++ > 10 && ++y > 20 ){

System.out.print("Inside if ");

}else{

System.out.print("Inside else ");

}

System.out.println(x +" "+y);

}

}

Inside else 11 20

1. public static void main(String args[]) {

int x = 10;

int y = 20;

if(x++ > 10 || ++y > 20 ){

System.out.print("Inside if ");

}else{

System.out.print("Inside else ");

}

System.out.println(x + " " + y); }

Inside if 11 21

#### Select the correct output for the following code:

public static void main (String[] args) {

int a=5;

a+=5+(++a)+(a++);

System.out.print(a);

}

22 a=a+5+(++a)+(a++)=5+5+6+6=22

1. public static void main (String[] args) {

int a=10;

a+=++a-5/3+6\*a;

System.out.print(a);

}

86 a+=++a-5/3+6\*a. a+=11-1+6\*11. a=10+11-1+66=86.

#### How many times will the following loop run?

for(int i=1;i<10;i\*=2)

{

System.out.println(i);

}

4 The loop would run for i=1,2,4,8.

#### Which value(s) can be used to initialize i so that the loop is finite:

public static void main (String[] args) {

for(int i=\_;i>0;i=i%3)

{

System.out.print("\*");

}

}

729,483 The value which is divisible by 3 would result in finite loop.

#### What will be the output the following code?

for(int i = 0; i < 5; i = i + 1){

System.out.print(i + " ");

i = i + 1;

}

0 2 4

#### What will be the output the following code?

for(int i = 1; i < 5; i = i + 1){

System.out.print(i +" ");

i = i - 1;

}

Infinite 1s

#### What will be the output ?

for(int i = 0; i < 2; i = i + 1) {

for(int j = 0; j < 2; j = j + 1) {

if (j == 1)

break;

System.out.print(j +" ");

}

}

0 0

#### Guess the output for the following code:

public static void main (String[] args) {

int i=0;

for(;;){

if(i==5)

break;

System.out.print(i);

i++;

}

}

01234 The loop would execute for i=0,1,2,3,4

#### Select the correct output for the following code:

public static void main (String[] args) {

for(int i=7;i!=0;i--)

{

System.out.print(i--);

}

}

Infinite Loop

#### Which code snippet would generate an error?

1.for(int i=1;;i++){

if(i==5)

break;

System.out.print(i);

}

2.for(int i=1;;i++){

if(i<5)

System.out.print(i);

else

break;

}

3.for(int i=1;;i++){

if(i>5) {

break;

System.out.print("break statement reached");

}

System.out.print(i);

}

In code snippet 3. the statement after break is never executed hence it gives an error

#### Which of these jump statements can skip processing remainder of code in its body for a particular iteration ?Continue

#### "break" is used to exit from the current loop. "return" statement is used to exit from the current function. "continue" is used to skip the current iteration of a loop and continue with the next iteration.

1. int i = 1;

while(i < 5) {

if(i == 3) {

break;

}

System.out.print(i + " ");

i++;

}

1 2

1. int i = 1;

while(i < 5) {

if(i == 3) {

continue;

}

System.out.print(i + " ");

i++;

}

1 2 [with infinte loop]

1. int i = 1;

while(i < 3) {

int j = 1;

while(j < 5) {

if(j == 3) {

break;

}

System.out.print(j + " ");

j++;

}

i++;

}

1 2 1 2

1. int i = 1;

while(i < 3) {

int j = 0;

while(j < 5) {

j++;

if(j == 3) {

continue;

}

System.out.print(j + " ");

}

i++;

}

1 2 4 5 1 2 4 5

**Lecture 8 : Functions & Scope**

#### Predict the output of the following code:

public static void main (String[] args) {

int n=10,r=6;

int factn=1,factr=1,factnr=1;

for(int i=2;i<=n;i++){

factn\*=i;

if(i<=r)

factr\*=i;

if(i<=n-r)

factnr\*=i;

}

int ncr=factn/(factr\*factnr);

System.out.print(ncr);

}

210 factn=3628800 factr=720 factnr=24 ncr=3628800/(720 \* 24) =210

#### What is the return type of a method that does not returns any value ?

#### void The functions which doesn't return any value, their return type is "void".

#### Let’s say the problem is - You will be given two numbers(both integers) and you need to return their sum.For this problem, what should be the return type of function -int

#### We need to return the sum of two integers, which is again an integer. So the sum that we want to return is of type "int". Hence return type should be "int" for this function.

#### What will be the output of the following code ?

public static void func(int a, int b){

System.out.println(a + b);

}

public static void main(String[] args) {

int a = 7;

func(a, 12);

}

19

#### What will be the output of the following code ?

public static void demo(int a, int b){

System.out.println(a + " " + b);

}

public static void main(String[] args) {

int a = 5;

int b = 15;

demo(a);

}

Compile time error

#### What should be return type of the following function:

public static \_\_\_\_ division(int a,int b){

float c=a/b;

return c;

}

Float As we need to return a decimal value we can use float or double.

#### Would the following code generate an error:

public static double add(int a,int b){

float c=a+b;

return c;

}

public static void main (String[] args) {

System.out.print(add(10,3));

}

No There is no problem in converting float to double.

#### Will the given code generate any error:

public static void func1(int a){

System.out.print("a");

}

public static void main (String[] args) {

func1(2.5);

}

Yes The datatype of argument in function is int whereas we have passed a double value. So the code would generate an error.

#### What will be the output of the following code:

public static void func2(){

System.out.print("#");

}

public static void func1(){

System.out.print("\*");

func2();

}

public static void main (String[] args) {

func2();

func1();

}

#\*# First we call func2, and # is printed.Then we call func1, which first prints \* and then calls func2, which prints #.So the output is #\*#.

#### Will the given code generate any error:

public static void func1(int a){

System.out.print("a");

}

public static void main (String[] args) {

func1(2.5);

}

Yes The datatype of argument in function is int whereas we have passed a double value. So the code would generate an error.

#### What will be the output of the following code:

public static void func1(int a,int b){

int ans=1;

for(int i=0;i<b;i++){

ans\*=a;

}

System.out.print(ans);

}

public static void main (String[] args) {

func1(2,5);

}

32 The loop would execute 5 times. ans=2 \* 2 \* 2 \* 2 \* 2. ans=32.

#### What will be the output of the following code ?

public static void doubleValue(int a ){

a = a \* 2;

}

public static void main(String[] args) {

int a = 8;

doubleValue(a);

System.out.println(a);

}

8

#### What will be the output of the following code ?

public static int func(int a){

a += 10;

return a;

}

public static void main(String[] args) {

int a = 5;

func(a);

System.out.println(a);

}

5

#### What will be the output of the following code ?

public static int square(int a){

int ans = a \* a;

return ans;}

public static void main(String[] args) {

int a = 4;

a = square(a);

System.out.println(a);

}

#### 16

#### Will following code generate any error ?

public class Main {

public static void func(int a) {

int b = a;

b = b + 10;

}

public static void main(String[] args) {

int a = 10;

func(a);

System.out.println(b);

}

}

yes Code will generate error as we are trying to print value of variable b in main and there isn't any variable b created or accessible in main. Variable b in the code is created inside function \_func\_ and is local to this function and not accessible outside it.

#### Will following code generate any error ?

public class Main {

public static void func(int a) {

int b = 10;

a = a + 10;

System.out.println(a);

}

public static void main(String[] args) {

int a = 10;

func(a);

System.out.println(a);

}

}

No

#### What will be the output of the following code:

public static boolean isPrime(int x){

for(int i=2;i<x/2;i++){

if(x%i==0)

return false;

}

return true;

}

public static void main (String[] args) {

System.out.print(isPrime(47));

}

true The loop would execute for i=2 to 22. And 47 is not divisible by any of them.

#### What will be the output of the following code:

public static int sum(int a,int b){

System.out.print("int sum ");

return a+b;

}

public static long sum(long a,long b){

System.out.print("long sum ");

return a+b;

}

public static void main (String[] args) {

int a=4;

int b=5;

System.out.print(sum(a,b));

}

True As both the datatypes are int So first function would be called

1. public static int sum(int a,int b)

{

System.out.print("int sum ");

return a+b;

}

public static double sum(double a,double b)

{

System.out.print("float sum ");

return a+b;

}

public static void main (String[] args) {

System.out.print(sum(5,4));

System.out.print(sum(5.0,4.0));

}

int sum 9float sum9.0 sum(5,4) would call the 1st Function and sum (5.0,4.0) would call 2nd Function.