Java Data Types

Data types specifies different sizes and values that can be stored in the variable. There are two types of data types in Java:

1)primitive data type

2)Non-primitive data type

Primitive data types

There are 8 primitive data types in java

1)boolean

2)char

3)Byte

4)short

5)int

6)long

7)float

8)double

1)boolean:

Boolean type represent one bit of information either true or false values of boolean type is converted into any other type

Eg: Boolean a=true;

2)char :

It is a 16-bit signed unicode character

Eg:char a='A';

3) byte:

It is an 8-bit signed two's complement integer, its value ranges from -128 to 127

Byte a=10,b=-20;

4)short:

It is a 16-bit signed two's complement integer, its value ranges from -32768 to 32767

Short a=1000,b=-1000;

5)int:

It is an 32-bit signed two's complement integer, its value ranges from -2147483648 to 2147483647

Int a=100000,b=-100000;

6)long:

It is an 32-bit signed two's complement integer, its value ranges from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807

Long l=1000000000000L,l1=-1000000000L;

7)float:

The float data type is a single-precision 32-bit IEEE 754 floating point.Its value range is unlimited.It is recommended to use a float (instead of double) if you need to save memory in large arrays of floating point numbers.
Eg:float f=232.35,f1=-2342.13;

8)double:
The double data type is a double-precision 64-bit IEEE 754 floating point. Its value range is unlimited. The double data type is generally used for decimal values just like float. The double data type also should never be used for precise values, such as currency. Its default value is 0.0d.
Eg: double d1=123.234

Non-primitive data types

Non primitive data types include **classes,strings and arrays etc..**

Operators in Java

Operators in java are symbols used to perform operations in java

Unary	postfix	*expr*++ *expr*--
	prefix	++*expr* --*expr* +*expr* -*expr* ~ !
Arithmetic	multiplicative	* / %
	Additive	+ -
Shift	shift	<< >> >>>
Relational	comparison	< > <=>=instanceof
	Equality	== !=
Bitwise	bitwise AND	&
	bitwise exclusive OR	^
	bitwise inclusive OR	|
Logical	logical AND	&&
	logical OR	||
Ternary	ternary	? :
Assignment	assignment	= += -= *= /= %= &= ^= |= <<= >>= >>>=

Task

1) public class HelloWorld{

```
    public static void main(String []args){
    int x=10;
    x++;
    System.out.println(x);
    }
 }
```

output:11

Description: Initially variable X is assigned with value 10 and it is post incremented, later 11 is printed

2)public class HelloWorld{

```
    public static void main(String []args){
    int a=10,b=20,c=30;
    b=a;c=b;
    System.out.println(c);
    int a=b=c=10;// Compilation error: variable a is already defined in method
    System.out.println(c);                          main(String[]args)    int a=b=c=10;
    int a,b,c;                                                                    ^
    a=b=c=10;// Compilation error: variable a is already defined in method
    System.out.println(c);              main(String[]args)    int a,b,c; three variables are
declared


    }
}
```

Output:compilation error

Description:variable once declared can not be redeclared in the same method so there exists an error

3)public class HelloWorld{

```
        public static void main(String []args){
        char ch='a';
        ch++;
        System.out.println((ch));



        }
}
```
Output:b

Description:Here initially ch is assigned with character a and its ascii value is 97 when we do ch++ its ascii value is incremented to 98 and the value of 98 in character is b,then b is printed

4)
```
public class HelloWorld{

        public static void main(String []args){
        double d=10.5;
        d++;
        System.out.println(d);



        }
}
```
Output:11.5

Description:Initially value of d is 10.5 and later it is incremented by 1 and it became 11.5

5)
```
public class HelloWorld{

        public static void main(String []args){
        boolean  b=true;
```

```
        b++;
        System.out.println(b);



        }
}
```

Output: compilation error

Description: boolean type cannot be incremented or decremented in java

6)

```
public class HelloWorld{

        public static void main(String []args){
        byte b=20;
byte b=b+1;
byte b=(byte)b+1 ;
System.out.println(b);



        }
}
```

Output:compilation Error

Description:variable once declared can not be redeclared in the same method so there exists an error

7)

```
public class HelloWorld{

        public static void main(String []args){
        byte a=10;
byte b=20;
byte c=a+b;
```

```
byte c=byte(a+b);

System.out.println(c);



        }
}
```

Output:compilation Error

Description:variable once declared can not be redeclared in the same method so there exists an error,while adding byte type we should type cast as byte else it will become int type while getting result value and if we declare result as byte then shows error

8)

```
public class HelloWorld{

        public static void main(String []args){
        System.out.println(10/0);
        System.out.println(10/0.0);



        }
}
```

Output:Run time Arithmetic Exception

Description:As we cannot divide a number by 0 compiler raises the arithmetic exception during runtime

9)

```
public class HelloWorld{

        public static void main(String []args){
        System.out.println('a'+'b');
        System.out.println('a'+1);
```

```
        System.out.println('a'+1.2);



        }
}
```

Output:195,98,98.2

Description: in the above print statments character values are converting into ascii values and the values are added and printed

10)

```
public class HelloWorld{

        public static void main(String []args){
String a="ashok";
int b=10 , c=20 , d=30 ;
a=b+c+d ;
a=a+b+c ;
b=a+c+d ;
        }
}
```

Output:compilation Error

Description:In first two arithmetic operations we are trying to add integer to a string type And for the next one we are adding string to an Integer so we get compilation error

11)

```
public class HelloWorld{

        public static void main(String []args){
        System.out.println(10<10.5);
```

```
        System.out.println('a'>100.5);
        System.out.println('b'>'a');
        System.out.println(true>false);



        }
}
```

Output:Compilation Error

Description:for the First three System.out.print() statements compiler compiles and prints true ,false,true and for the fourth as it is boolean type it can't compare boolean type so it shows an error

12)

```
public class HelloWorld{

        public static void main(String []args){
        System.out.println(10==20);
        System.out.println('a'=='b');
        System.out.println('a'==97.0);
        System.out.println(false==false);



        }
}
```

Output:false,false,true,true

Description:For all the print statements compiler compares the first value with second one and if both are matching prints true else prints false

13)

```
public class HelloWorld extends Thread{

        public static void main(String []args){
        Thread t1=new Thread( ) ;
Thread t2=new Thread( );
Thread t3=t1 ;

System.out.println(t1==t2);
System.out.println(t1==t3);




        }
}
```

Output:false,true

Description:In the first print statement two different statements are being compared so the output was false, In the second print statement we are comparing the same threads so the output was true

14)

```
public class HelloWorld{

        public static void main(String []args){
        System.out.println(true&false);
        System.out.println(true|false);
        System.out.println(true^false);



        }
}
```

Output:false,true,true

Description: true and false are converted into binary numbers and then for the first System.out.println(); it performs AND operation for binary numbers and converts into decimal the value is printed,for the second System.out.println(); it performs OR operation for binary numbers and converts into decimal , value is printed, for the third System.out.println(); it performs XOR operation for binary numbers and is converts into decimal the value is printed

15)

```
public class HelloWorld{

	public static void main(String []args){
	System.out.println(4&5);
	System.out.println(4|5);
	System.out.println(4^5);


	}
}
```

Output:4,5,1

Description: 4 and 5 are converted into binary numbers and then for the first System.out.println(); it performs AND operation for binary numbers and converts into decimal the value is printed,for the second System.out.println(); it performs OR operation for binary numbers and converts into decimal , value is printed, for the third System.out.println(); it performs XOR operation for binary numbers and is converts into decimal the value is printed

16)

```
public class HelloWorld{

	public static void main(String []args){
```

```
        System.out.println(~true);
        System.out.println(~4);



        }
}
```

Output:compilation Error

Description:In first print statement we have boolean value, for boolean” ~” does not work only” !” works, so it is showing compilation error for first print statement and for the second print statement it prints the opposite value of int and adding -1 to it

17)

```
public class HelloWorld{

        public static void main(String []args){
        System.out.println(!false);
        System.out.println(!4);



        }
}
```

Output:compilation Error

Error:For the first print statement it prints true as it is boolean type but for second it prints error because it is not boolean type,in System.out.println(); “!” operator works for only boolean type operator

18)

```
public class HelloWorld{

        public static void main(String []args){
int x=10 , y=15 ;
if(++x < 10  ||  ++y > 15) {
 x++;
}
else {
```

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

Output:12-----16

Description: In the above program x and y are assigned with values10,15 and later in if condition both are incremented by 1 and values became 11 ,16 . as the if condition satisfies again x is incremented and becomes 12 and both x and y are printed

---

```
public class HelloWorld{

        public static void main(String []args){
int x=10 , y=15 ;
if(++x < 10  &&  ++y > 15) {
 x++;
}
else {
        y++;
}
System.out.println(x+"------"+y);
        }
}
```

Output: 11-----16

Description In the above program initially x and y are assigned with 10 and 15 in if condition compiler checks 11<10 condition as it is false compiler doesnot go to and operation in if and directly goes to else part and increments value by 1 and prints the value below

---

```
public class HelloWorld{

        public static void main(String []args){
int x=10 , y=15 ;
```

```
if(++x < 10  &  ++y > 15) {
 x++;
}
else {
        y++;
}
System.out.println(x+"------"+y);
        }
}
```

Output:11--------17

Description:In if statement both x and y are pre incremented and later in if statement 11<10 is true so value 0 is obtained later 16>15 value 1 is obtained now perform “AND” operation for both 1 and 0 result is 0 so compiler goes to else statement and y is incremented

---

```
public class HelloWorld{

        public static void main(String []args){
int x=10 , y=15 ;
if(++x < 10  |  ++y > 15) {
 x++;
}
else {
System.out.println(x+"------"+y);
        y++;
}
System.out.println(x+"------"+y);
        }
}
```

Output:12-------16

Description:In if statement both x and y are pre incremented and later in if statement 11<10 is true so value 0 is obtained later 16>15 value 1 is obtained now perform “OR” operation for both 1 and 0 result is 1 so compiler goes to if statement and x is incremented

---

19)

```
public class HelloWorld{

	public static void main(String []args){
int x=130;
byte b=(byte)x;

System.out.println(b);



	}
}
```

Output:-126

Description:The byte value ranges from -128 to 127,130 has to be converted into byte so after 127, we have to add 3 to make 130 so it will traverse from -128 after 127 but not 128 and it goes to -126 as the third value is -126 and it is printed

20)

```
public class HelloWorld{

	public static void main(String []args){
int x=150;
short s=(short)x;
byte b=(byte)x;

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



	}
```

```
}
```

Output:150,-106

Description:first int value is converted into short and it is printed as 150,later as the byte value ranges from -128 to 127,150 has to be converted into byte so after 127, we have to add remaining 23 for 150 so it will traverse from -128 after 127 but not 128 and it goes to -106 as the twenty third value is -106 and it is printed

21)

```
public class HelloWorld{

        public static void main(String []args){
double d=130.456 ;
int x=(int)d ;

System.out.println(x);
byte b=(byte)d ;
System.out.println(b);



        }
}
```

Output:130,-126

Description: given value 130.456 is initially converted into int type so float value gets removed and printed as 130 as the byte value ranges from -128 to 127,130 has to be converted into byte so after 127, we have to add 3 to make 130 so it will traverse from -128 after 127 but not 128 and it goes to -126 as the third value is -126 and it is printed

22)

```
public class HelloWorld{
```

```
        public static void main(String []args){
        int x=(10>20)?30:((40>50)?60:70);


System.out.println(x);


        }
}
```

Output:70

Description: In the above program if 10 is greater than 20 it goes to 30 else it will go to 40>50 as it is false it went to 40>50 and then checks the condition 40>50 and then it is false so it went to 70 if it is true output is 60 , finally the value 70 is stored in x and it is printed

23)

```
public class OperatorsDemo{
        public static void main(String[] args)  {

                System.out.println(m1(1)+m1(2)*m1(3)/m1(4)*m1(5)+m1(6));

        }
        public static double m1(int i)   {
System.out.println(i);
        return i;
        }

}
```

Output:1,2,3,4,5,6,14.5

Description:Here compiler follows BODMAS rules,m1(2) is multiplied with m1(3) and then divided with m1(4)and later multiplied with m1(5) and then m1(1) is added later m1(6) is added