**Java Identifiers**

Any name in java is called an identifier.

Whether it is a name of data type, name of the method, name of the java class is known as an identifier.

Class A{

p s v main(String args[]){  
int a =5;

}}

Here, A, main, String, args, a are identifiers.

Class A{

p s v main(String args[]){  
int String = 5; //It will compile fine

int Runnable = 4; //It will compile fine

}}

We can use java pre-defined class name as an identifier, but it is not a good practice because it reduces readability.

There is no limit in the size of identifier.

Valid identifier: A-Z, a-z, 0-9, $(all currencies), \_

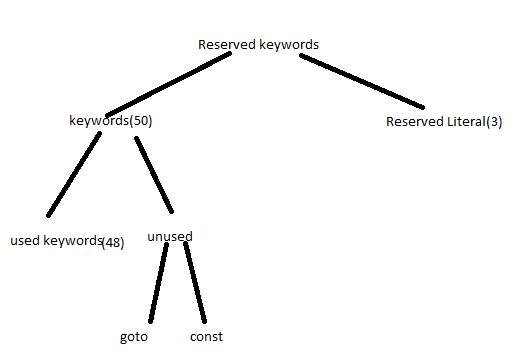
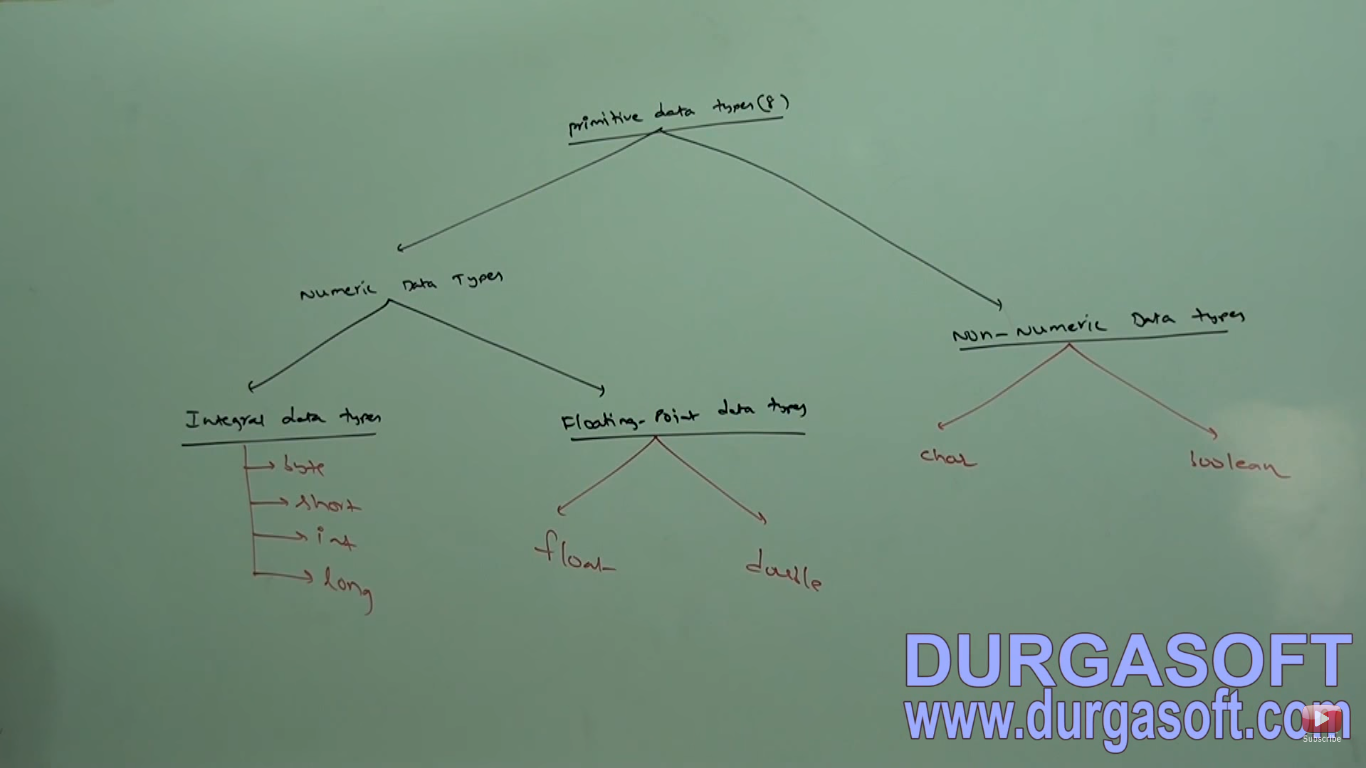
**Invalid:** 123Abc, 123#, a@d

**Valid:** total\_number, \_$\_$

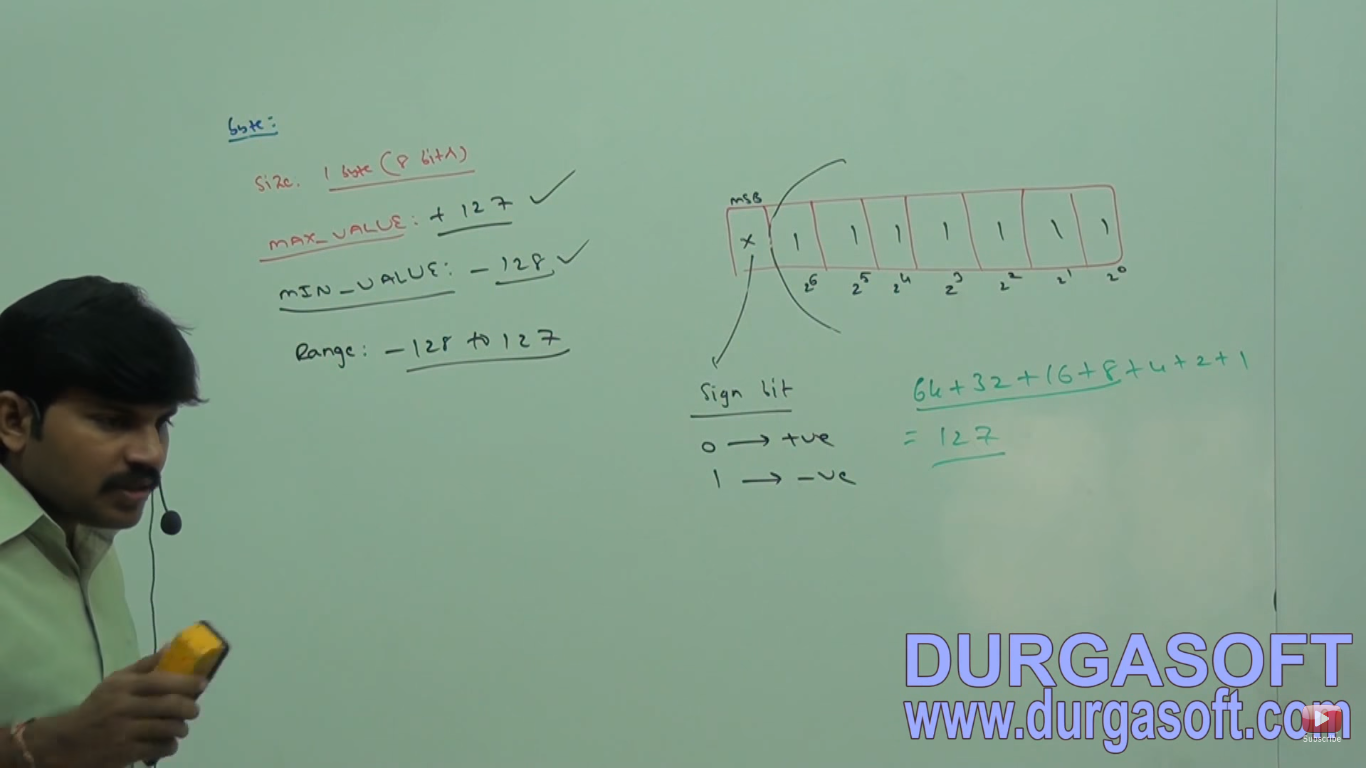
**Reserved Words:**

There are 53 reserved words.

50 keywords and 3 reserved literals



Except Boolean and char the remaining data types are signed.



Positive numbers will be represented directly in the memory and -ve numbers are represented in 2’s compliment.

File supported form and network supported form is in byte.

There are two types of streams.

1. Character stream
2. Byte stream

Byte is the best choice if we want to handle data in the streams.

short: 2 byte (16 bits): -2^15 to 2^15-1

Short data type is best suitable for 16 bits processor like 8085. But these processors are out dated and hence the corresponding short data types are also out data type.

int: 4 bytes (32 bits): -2^31 to 2^31-1

long: 8 bytes (64 bits): -2^63 to 2^63-1

java is Unicode based. Older languages were ascii based.

Number of allowed different ascii characters are <=256. To represent these 256 characters 8bits are enough. Therefore, the size oy character in older lang. was 1 byte (8 bits). But in java there are 2 byte(16 bits).

For integral literals. For integral data types(byte, short, int, long), we can represent literals(values ) in three ways.

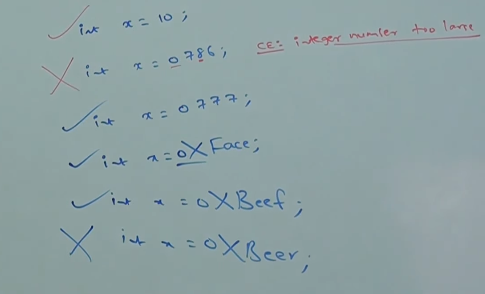
**Floating point literals can only be represented in decimal form.**

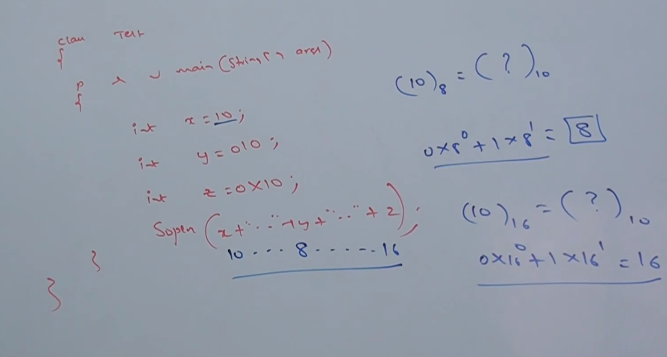
1. Decimal form (base 10).
2. Octal form (base 8): literal value should be prefixed with 0 (010).
3. Hexadecimal form (base 16): allowed digits are 0-9a-f or 0-9A-Z

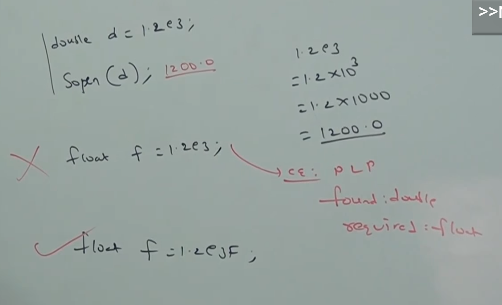
Represented as: 0X10 or 0x10.

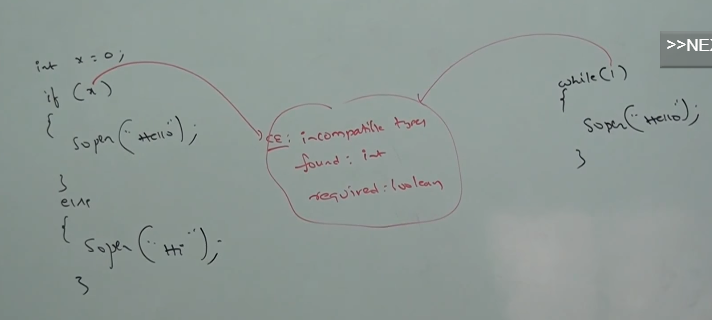
After 1.7

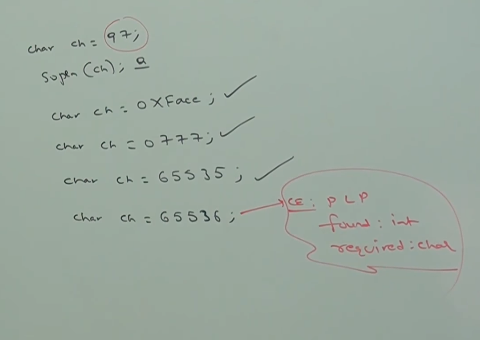
1. Binary form: 0B1111 or 0b1111 (15)

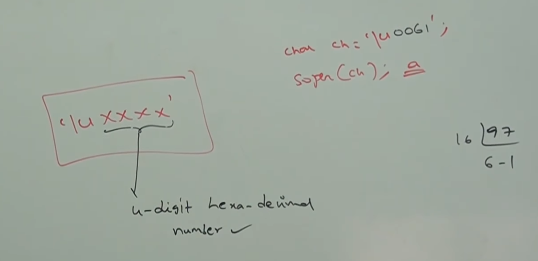








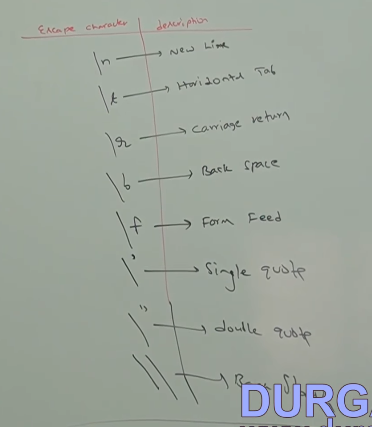




Every escape character is a valid character literal.

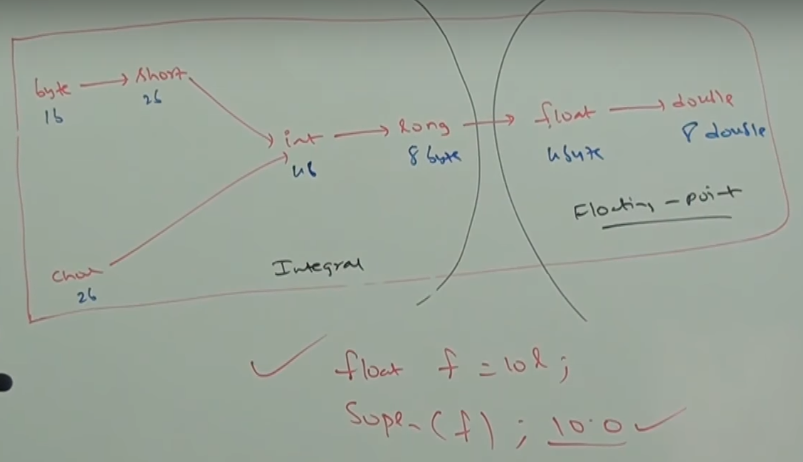
Example:

There are eight escape characters.



char ch = ’/n’;

char ch = ’/t’;



8byte long value we can assign to 4 byte float variables because the both are following different memory representation internally.