

Compiler is a software which takes the instructions from source code(code written by developer), scans the code only once and generates the instructions in MLL.

Interpreter is a software which takes the instructions from source code(code written by developers), scans the code line by line and generate the instructions in MLL.

How does java program executes?

```

           Compilation(compiled mode)           Execution(interpreted mode)
java -----compiler -----> .class file(byte code)-----JVM-----
>Output
```

+++++  
Reserve words  
+++++

In java some identifiers are reserved to associate some functionality or meaning such type of reserved identifiers are called as "reserve words".

Reserve words for Datatypes(8)  
byte, short, int, long, float, double, char, boolean

Reserve words for flow control(11)  
if, else, switch, case, default, for, while, do, break, continue, return

Reserve words for identifiers(11)  
private, public, protected, final, abstract, native, static, strictfp, synchronized, transient, volatile

Reserve words for ExceptionHandling(6)  
try, catch, finally, throw, throws, assert(1.4 version)

Reserve words for Class types(7)  
class, package, import, extends, implements, interface, enum

Reserve words for Object types(4)  
new, instanceof, super, this

Reserve word for return type of methods(1)  
void

Conclusions

1. All reserve words in java contains only lower case alphabets.
2. In java to create an object for a class we have "new" keyword, but we don't have delete keyword to destroy the object where as destroying the object is taken care by a program(thread) called "GarbageCollector".

Which of the following list contains only java reserve words

- a. final, finally, finalize(invalid)
- b. throw, throws, thrown(invalid)
- c. break, continue, return, exit(invalid)
- d. goto, constant(invalid)
- e. byte, short, Integer(invalid), long
- f. extends, implements, imports(invalid)
- g. finalized(invalid), synchronized
- h. instanceof, sizeof(invalid)
- i. new, delete(invalid)

j. instanceof, instanceof(invalid)  
k. public, static, void, main(invalid), String(invalid), args(invalid)

identifier :: name of class, methodname, variable name, label name.  
reservewords :: special meaning given for few identifiers (known to Compiler and JVM)

+++++

Software requirements to write java program

+++++

1. Download editplus from the following link ::

<https://www.editplus.com/download.html>

2. Download jdk software from the following link ::

<https://www.oracle.com/java/technologies/javase/jdk17-archive-downloads.html>

Steps to install jdk software

+++++

1. Double click on the downloaded setup file

2. click on next, next, next and finish the process.

3. Upon installation, the jdk software will be installed in the following location

C:\Program Files\Java\jdk-18.0.1.1

4. set environmental variable called "PATH" to jdk/bin folder

5. Now our machine is ready to run java programs.

Steps to install editplus

+++++

1. Just double click on editplus and install the software

2. Create one folder in any drive of your choice (D:\octbatchmicroservices)

3. Open editplus with this folder

+++++

Datatypes

+++++

Every variable has a type, every expression has a type and all types are Strictly defined moreover every assignment should be checked by the compiler for the type compatibility, so we say java is "Strictly typed programming language".

refer: diagram

++++

byte

++++

byte(size: 1 bytes (8 bits))

minrange: -128 (MIN\_VALUE)

maxrange: +127 (MAX\_VALUE)

Corresponding wrapper class is java.lang.Byte

When to use byte datatype?

=> byte datatype is suitable only when we work with handling data in terms of streams either from the file or from the network.

Note::

If the no is positive number then first bit will be zero.

If the no is negative number then first bit will be one.

Negative number will be stored in 2's complement.

eg

byte b=127;

byte b=130; //CE::loss of precision

```
byte b=10.5; //CE::Incompatible types
byte b=true; //CE::Incompatible types
byte b="ABC"; //CE::Incompatible types
```

#### short

It is the rarely used data type in java language.

Corresponding Wrapper class is java.lang.Short.

size:: 2 bytes

minrange::-32768(MIN\_VALUE)

maxrange::+32767(MAX\_VALUE)

short data type is mostly suitable only for 8086mp, since it is outdated we don't use

short data type in java language

```
eg:: short s=32456;
      short s=32768//CE
      short s=true; //CE
```

#### int

It is the most commonly used datatype to store numbers in java language

Corresponding wrapper class is java.lang.Integer

size:: 4bytes

minrange::-2147483648(MIN\_VALUE)

maxrange::2147483647(MAX\_VALUE)

```
eg:: int i=130;
      int i=10.5//CE:incompatible types
      int i=true; //CE::incompatible types
```

#### long

Whenever int is not enough to hold the value then we opt for long datatype

Corresponding wrapper class is java.lang.Long

size:: 8bytes

minrange::-9223372036854775808(MIN\_VALUE)

maxrange::9223372036854775807(MAX\_VALUE)

When to use long datatype?

=> It is preferred when we work with file whose size is in terms of GB.

```
eg:: long l=35;
      long l=35L;
      int x =10L;
```

To specify the long type we can prefix it with 'L' or 'l'.

\*\*\*\*\*Tomorrow session from 10.00AM to  
1.30PM\*\*\*\*\*

