

1.

Statically Type

- This means, it check the types of inputs at the compile time

Dynamically Type

- This means, it check the types of the inputs at the running time

Strongly type

- This means, it strongly consider about the types of inputs. According to that, it store the numbers where, that suitable location for numbers and we can't store other type of input in there.

Loosely Type

- It do not strongly consider about the types before store their data.

Java is a statically type, dynamically type, strongly type language.

2.

Case sensitive

Case sensitive languages have ability to distinguish between upper and lower letters in the character set. That means, the letter 'a' considered different than the letter 'A'.

ex -

```
String name="Samith Dulanga";
```

```
System.out.println(Name); // Show an Error
```

Case insensitive

The language with case insensitive ability, neglect the lower and upper case of the characters.

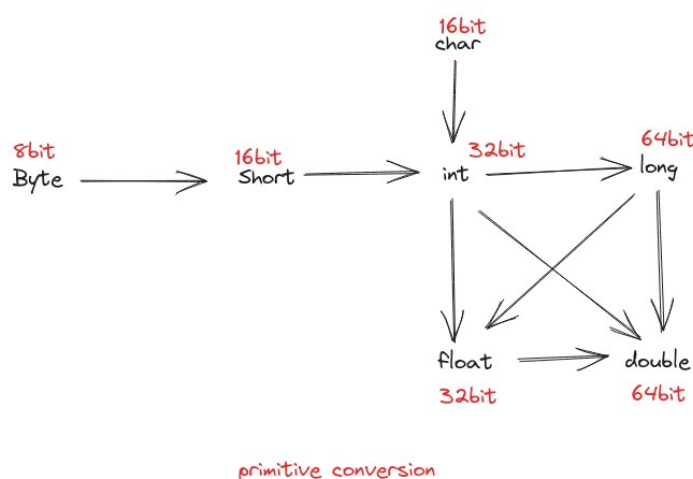
Java is a case sensitive language. Because it has different values for there characters. For an instance, ASCII value of 'A' is 65 and ASCII value of 'a' is 97.

3. In this concept it convert in to the same type using different identifiers.

Ex – `int x=30;`

`int y=x;`

4.



According to the direction that showed in the diagram, we can execute the primitive conversion. In this case, we can assign the value from small data type to higher data type because of the bit size. In other case, any number can assign to float and double because of the data structure they have.

Ex - `byte a=20;`
`int b=a;`

5.

run time constant – value assign to the given variable while program is running

Ex – `int piValue = Math.PI;`

compile time constant – value assign to the given variable at the time of compiling

Ex – `int x=10;`

6.

Implicitly narrowing primitive conversion

In this case assign the value in higher data type to a lower data type. But, to be an implicitly narrowing primitive conversion, it must follow the following rules.

- Assign value should be within the range of small data type.
`Byte newNum=20;`
- Assign value should be constant
`final int x=10;`
`byte myByte =x;`

Explicitly narrowing primitive conversion

In this case also assign the larger data type value into smaller type. But that value is not in a range of smaller data type. So that, we can occur some difference in between input and output value.

7.

long and float have different data structures to store their values. That means, float has allocated one bit for sign, 8 bits for exponent and 23 bits for fraction. Because of this reason, long value can store in the float although long has allocated the higher bits to store their value.