JRE Vs JDK → feed source code & produces Java code

(Java Run-time Environment)
↓
→ Creating & Running Java Apps.
→ End-users requires only JRE

[Java Development kit]
↳ tools are provided platform Independent
→ Developers requires JDK

Java App
↓
JRE
Host Environment

IDE [Integrated Development Environment]
↳ code to compile & run it.

Package → all lowercase

Variables | Datatypes | Mathoperations

Primitive
↳ int, float, Char, Boolean

↓
int mem;   name → first character → not number

double → 64 bit 32 ← float

++ applies before returning value.

Operator Precedence → Postfix, Prefix, Multiplicative Additive

Type Conversion:
long lval = 50
↓
int lval = (int)lval)

for(loop-variable-declaration : array)
statement;

Classes → allows storage.
→ Method returns a single value.

Access modifiers
Public void SetA (double A)
{ this. a = a;
}
public ~~double~~ double get A
{ return a;}

Setvalue { ~~this.setValue = setvalue}~~
(double a)
get value { return a value}

Field Initial State → zero

Field Initializers → method call.
↳ Mathround( )

Field Initialization
↳ Initialization Block
↳ Constructor

Constructor → No. of Parameters
↳ Type of Parameters.

Clone → create new object Instance duplicates current Instance

getClass → return type Information

Constructors are not inherited.

Wrapper class
↳ treat as Object

== & Equals
same output
↓
for no's if it is lengthe b/w 128 ... -128

Runtime Exceptions → Unchecked → Null Pointer Exception

Compile time Exceptions → Checked → IO Exception

java.lang.Comparable → Interface

compareTo → Method → used for determining relative order

Static members & fields are shared classwide

Static Initialization blocks → one time Initialization.

↳ cannot access instance keywords.