Core Java Cheat Sheet

Java Programming

Java is a high level, general purpose programming language that produces software for multiple platforms. It was developed by James Gosling in 1991 and released by Sun Microsystems in 1996 and is currently owned by Oracle.

Primitive Data Types

Types	Size	Min	Max
byte	8	-128	127
char	16	$2^{16}-1$	All Unicode characters
short	16	$2^{15}-1$	From +32,767 to -32,768
int	32	$2^{31}-1$	From +2,147,483,647 to -2,147,483,648
long	64	$2^{63}-1$	From +9,223,372,036,854,775,807 to -9,223,372,036,854,775,808
float	32	2^{-149}	$(2-2^{-23})\cdot 2^{127}$
double	64	$(2-2^{-52})\cdot 2^{1023}$	From 1.797,693,134,862,315,7 E+308 to 4.9 E-324
boolean	1	--	--

Java Operators

Operator	Category	Precedence
Unary Operator	postfix	expression++ expression--
	prefix	++expression --expression +expression -expression ~!
Arithmetic Operator	multiplication	* / %
	addition	+ -
Shift Operator	shift	<< >> >>>
Relational Operator	comparison	< > <= >= instanceof
	equality	== !=
Bitwise Operator	bitwise AND	&
	bitwise exclusive OR	^
	bitwise inclusive OR	|
Logical Operator	logical AND	&&
	logical OR	||
Ternary Operator	ternary	? :
Assignment Operator	assignment	= += -= *= /= %= &= ^= |= <<= >>= >>>=

Java Variables

```
{public|private} [static] type name [= expression|value];
```

Java Methods

```
{public|private} [static] {type | void} name(arg1, ..., argN ){statements}
```

Data Type Conversion

```
// Widening (byte<short<int<long<float<double)
int i = 10; //int--> long
long l = i; //automatic type conversion
// Narrowing
double d = 10.02;
long l = (long)d; //explicit type casting
// Numeric values to String
String str = String.valueOf(value);
// String to Numeric values
int i = Integer.parseInt(str);
double d = Double.parseDouble(str);
```

User Input

```
// Using BufferReader
BufferedReader reader = new BufferedReader(new
InputStreamReader(System.in));
String name = reader.readLine();
// Using Scanner
Scanner in = new Scanner(System.in);
String s = in.nextLine();
int a = in.nextInt();
// Using Console
String name = System.console().readLine();
```

Basic Java Program

```
public class Demo
 {
   public static void main(String[] args)
  {
     System.out.println("Hello from edureka!");
  }
 }
```

Iterative Statements

```
// for loop
for (condition) {expression}
// for each loop
for (int i: someArray) {}
// while loop
while (condition) {expression}
// do while loop
do {expression} while(condition)
```

Fibonacci Series

```
for (i = 1; i <= n; ++i)
{
System.out.print(t1 + " + ");
int sum = t1 + t2; t1 = t2;
t2 = sum;
}
```

Pyramid Pattern

```
k = 2*n - 2;
for(i=0; i<n; i++)
{
  for(j=0; j<k; j++){System.out.print(" ");}
  k = k - 1;
  for(j=0; j<=i; j++ ){System.out.print("* ");}
  System.out.println();
}
```

Core Java Cheat Sheet

Decisive Statements

```
//if statement
if (condition) {expression}
//if-else statement
if (condition) {expression} else {expression}
//switch statement
switch (var) { case 1: expression; break; default:
expression; break; }
```

Prime Number

```
if (n < 2)
{
   return false;
}
For (int i=2; i <= n/i; i++)
 {
   if (n%i == 0) return false;
 }
return true;
```

Factorial of a Number

```
int factorial(int n)
{
   if (n == 0)
      {return 1;}
   else
      {
       return(n * factorial(n-1));
      }
}
```

Arrays in Java

1-Dimensional

```
//Initializing
type[] varName= new type[size];
// Declaring
type[] varName= new type[]{values1, value2,...};
```

Array with Random Variables

```
double[] arr = new double[n];
for (int i=0; i<n; i++)
{a[i] = Math.random();}
```

Maximum Value in Array

```
double max = 0;
for (int i=0; i<arr.length(); i++)
{ if(a[i] > max) max = a[i]; }
```

Reversing an Array

```
for(int i=0; i<(arr.length())/2; i++)
{ double temp = a[i];
  a[i] = a[n-1-i];
  a[n-1-i] = temp; }
```

Multidimensional Arrays

```
// Initializing
datatype[][] varName = new dataType[row][col];
// Declaring
datatype[][] varName = {{value1, value2....},{value1,
value2....}..};
```

Transposing a Matrix

```
for(i = 0; i < row; i++)
 { for(j = 0; j < column; j++)
  { System.out.print(array[i][j]+" "); }
  System.out.println(" ");
}
```

Multiplying Two Matrices

```
for (i = 0; i < row1; i++)
{ for (j = 0; j < col2; j++)
 { for (k = 0; k < row2; k++)
   { sum = sum + first[i][k]*second[k][j]; }
  multiply[i][j] = sum;
sum=0; }}
```

Java Strings

```
// Creating String using literal
String str1 = "Welcome";
// Creating String using new keyword
String str2 = new String("Edureka");
```

String Methods

```
str1==str2 //compare the address;
String newStr = str1.equals(str2); //compares the values
String newStr = str1.equalsIgnoreCase() //
newStr = str1.length() //calculates length
newStr = str1.charAt(i) //extract i'th character
newStr = str1.toUpperCase() //returns string in ALL CAPS
newStr = str1.toLowerCase() //returns string in ALL LOWERCASE
newStr = str1.replace(oldVal, newVal) //search and replace
newStr = str1.trim() //trims surrounding whitespace
newStr = str1.contains("value"); //Check for the values
newStr = str1.toCharArray(); //Convert into character array
newStr = str1.IsEmpty(); //Check for empty String
newStr = str1.endsWith(); //Checks if string ends with the
given suffix
```