# Assignments Solution for Week -2

Akash Pal - akash.majida@gmail.com

1. **List the Operators in Java?**

**Operators in Java:**

**->** Arithmetic Operators
-> Relational Operators
-> Logical Operators
-> Assignment Operators
-> Unary Operators
-> Bit-wise Operators

**2. Explain about Increment and Decrement operators and give an examples**

Increment and Decrement Operators in Java and used to increase or decrease the value by 1. For example, Incremental operator ++ is useful to increase the existing variable value by 1(i=i+1). Moreover, the decrement operator - - is useful to decrease or subtract the current value by 1(i=i-1). The syntax of both increment and decrement operators in Java Programming to prefix or postfix is

Increment Operator : ++x or x++
Decrement Operator : --x or x- -

**3.What are the Conditional Operators in Java?**

They are used when a condition comprises more than one boolean expression. For instance, if we want to print a number only if it is greater than 2 and less than 5, then we will use conditional operators to combine the 2 expressions. We have 3 types of conditional operators - **logical-and, logical-or ternary operator.**

**Logical-and operator(&&)**

It is used when we want the condition to be true if both the expression and true

**Syntax**

```
if(condition - 1 && condition -2){
Statement;
}
```

**Logical-or operator(||)**

This operator is used when we are satisfied as long as any one of the boolean expression is evaluated as true.

**Syntex**

```
if(condition -1 || condition -2){
Statement;
}
```

**Ternary operator(?:)**

It is a smaller version for the if-else statement. If the condition is true then the statement - 1 is executed else the statement -2 is executed.

**Syntex**

```
Condition? Statement -1 : statement -2;
```

**4. what are the types of operators based on the number of operands?**

There are three types of operators in java based on the number of operands. They are:

Unary operator
Binary operator

**5. What are the conditional Statements and use of conditional statements in Java?**

-> If to specify a block of code to be executed, if a specified condition is true
-> else to specify a block of code to be executed , if the same condition is false
-> else if to specify a new condition to test, if the first condition is false

-> switch to specify many alternative blocks of code to be executed

**Q6 What is the syntax of if else statement?**

```
if(condition){
    //block of code to be executed if condition is  true
} else if (condition) {
  // block of code to be executed if the condition1 is false and condition2 is
true
} else{
  //block of code to be executed if the condition1 is false and condition1 is
false
}
```

**7. What are the 3 types of iterative statements is java?**

**3 types of iterative statements is java**

for loop
while loop
do while loop

**8. What do you mean by an Array?**

It refers to index collection of fixed no of homogeneous data elements.
Single variable holding multiple values which improves readability of the program.

**Q9. How to create an Array?**

**Ans ->**

Array declarations
Single Dimension Array
Two Dimensional Array
Jagged Array

Declaration of array
->int[] a; //recommended to use as variable is separated from type.
-> int a[];
-> int[6] a; //compile time error: we can not specify the size;
Array Construction
Every array in java is an object hence we create using a new operator.
Example
Int[] a;
a=new int[5];
Or
Int[] a = new int[5];
-> For every type corresponding classes are available but the these classes are part of java language but not application at the programmer level.
int [] [i
Float[] [F
Double[] [D

**10. State the difference StringBuffer and StringBuilder in Java?**

StringBuffer and StringBuilder are two Java classes for manipulating strings. There are mutable objects, i.e, They can be modified, and provide various methods such as insert(), substring(),delete(),and append(),for String manipulation.

StringBuffer: The StringBuffer class was created by the Java Team When they realized the need for an editable string object. Nevertheless, StringBuffer has all methods synchronized, meaning they are thread-safe. Therefore , StringBuffer allows only one thread to access a method at once, so it is not possible to call StringBuffer methods from two threads simultaneously, which means it takes more time to access. The Stringbuffer class has synchronized methods, making it thread-safe, slower, and less efficient than StringBuilder. The StringBuffer class was introduced in Java 1.0.

Syntax:

StringBuffer var = new StringBuffer(str);

StringBuilder: it was at that point that java team realized that making all methods of StringBuffer Synchronized methods. Unlike StringBuffer, StringBuilder does not offer synchronized methods, which makes it less thread-safe, faster, and more efficient. StringBuilder was introduced in Java 1.5 in response to StringBuffer’s shortcomings.

Syntax:

StringBuilder var = new StringBuilder(str);