BHAVANS COLLEGE AUTONOMOUS, ANDHERI WEST

PRACTICAL JOURNAL

INDEX

Sr No.	Practical No.	Topic	Date	Signature

// Print statement

```
fun main() {
    println("Hello World...!")
}
```

```
Hello World...!
```

// Variable declaration - <u>val</u> and <u>var</u>

```
fun main() {
    val a = 5
    var b = 5
    a = 10
    b = 10
    println(a)
    println(b)
}
```

```
'val' cannot be reassigned.
```

// Variable usage

```
fun main() {
    val str = "Shinchan"
    println("Hello, You got a message from $str")
}
```

```
Hello, You got a message from Shinchan
```

// Assignment Options

```
fun main() {
    var test = 0
    println("var test = 0 --> $test")
    test = 5
    println("test = 5 -->test")
    test = test + 5
    println("test = test + 5 --> $test")
    test+=5
    println("test += 5 --> $test")
    test-=5
    println("test -= 5 --> $test")
    test*=5
    println("test *= 5 --> $test")
    test/=5
    println("test /= 5 --> $test")
}
```

```
var test = 0 --> 0
test = 5 --> 5
test = test + 5 -->10
test += 5 --> 15
test -= 5 --> 10
test *= 5 --> 50
test /= 5 --> 10
```

// Variable Declaration methods [either value/initialise or type mentioning]

```
fun main() {
    val a: Int
    a = 10                  // 1st declared then assigned
    println("a = $a")
    val b: Int = 20         // declared and assigned at the same time
    println("b = $b")
}
```

```
a = 10
b = 20
```

// List – allows duplicate values

```
fun main() {
    val lst: List<String> = listOf("Red","Yellow","Green") // :List<String> is not necessary
    println(lst)
}
```

```
[Red, Yellow, Green]
```

// Mutable List

```
fun main() {
    val lst: MutableList<String> = mutableListOf("Red","Yellow","Green")
    println(lst)                                    // :MutableList<String> is not necessary
    lst.add("Blue")
    println(lst)
}
```

```
[Red, Yellow, Green]
[Red, Yellow, Green, Blue]
```

// Set – doesn't allow duplicate values

```
fun main() {
   val st: Set<String> = setOf("Monday","Tuesday","Wednesday")
   println(st)
}
```

```
[Monday, Tuesday, Wednesday]
```

// Mutable Set

```
fun main() {
   val st: MutableSet<String> = mutableSetOf("Monday","Tuesday","Wednesday")
   println(st)
   st.add("Sunday")
   println(st)
   st.remove("Monday")
   println(st)
}
```

```
[Monday, Tuesday, Wednesday]
[Monday, Tuesday, Wednesday, Sunday]
[Tuesday, Wednesday, Sunday]
```

// Set Count

```
fun main() {
   val st: Set<String> = setOf("Monday","Tuesday","Wednesday","Monday")
   println("Set has ${st.count()} values.") // Counts only unique records
}
```

```
Set has 3 values.
```

```
fun main() {
   val st: List<String> = listOf("Monday","Tuesday","Wednesday","Monday")
   println("List has ${st.count()} values.") // Counts all records
}
```

```
List has 4 values.
```

// in keyword

```
fun main() {
   val st: Set<String> = setOf("Sunday","Monday","Tuesday","Wednesday")
   println("Sunday" in st)
}
```

```
true
```

// if-else statements

```
fun main() {
   val a = 5
   val b = 10
   val max: Int
   if (a > b) {max = a} else {max = b}
   println("Maximum value is $max")
   val new_max = if (a > b) a else b
   println("Maximum value is $new_max")
}
```

```
Maximum value is 10
Maximum value is 10
```

```
fun main() {
   val a = 10
   val b = 10
   val max: String
   if (a > b) {
      max = "a has maximum value"
   } else if (b > a) {
      max = "b has maximum value"
   } else {
      max = "Both have same value."}
   println(max)
}
```

```
Both have same value.
```

// for loop

```
fun main() {
   for (i in 1..5){
      println("Value of i = $i")}}
```

```
Value of i = 1
Value of i = 2
Value of i = 3
Value of i = 4
Value of i = 5
```