Programs on Array

Write a Scala program to sum values of an given array.

object Scala\_Array {

def main(args: Array[String]): Unit = {

var nums = Array(1.2, 1.7, 1.12, 1.16, 1.81, 1.99)

println("Original Array elements:")

// Print all the array elements

for ( x <- nums ) {

//print(x+" ")

print(s"$x, ")

}

println("\nUsing sum():")

val result = nums.sum

println(s"Result: ${result}");

println("\nUsing for loop:")

var total = 0.0;

for ( i <- 0 to (nums.length - 1)) {

total += nums(i);

}

println(s"Result: ${total}");

}

}

Write a Scala program to remove a specific element from an given array.

object Scala\_Array

{

def main(args: Array[String]): Unit =

{

val colors = Array("Red","Blue","Black","Green","White")

println("Original Array elements:")

// Print all the array elements

for ( x <- colors ) {

print(s"${x}, ")

}

println("\nReplace some elements with ''/null etc.:")

colors(0) = ""

colors(3) = null

println("Now the Original Array becomes:")

// Print all the array elements

for ( x <- colors ) {

print(s"${x}, ")

}

}

}

Write a Scala program to reverse an array of integer values.

object Scala\_Array {

def test(nums: Array[Int]): Array[Int] = {

var temp1 = 0

var temp2 = 0

var index\_position = 0

var index\_last\_pos = nums.length - 1

while (index\_position < index\_last\_pos) {

temp1 = nums(index\_position)

temp2 = nums(index\_last\_pos)

nums(index\_position) = temp2

nums(index\_last\_pos) = temp1

index\_position += 1

index\_last\_pos -= 1

}

nums

}

def main(args: Array[String]): Unit = {

var nums1 = Array(1789, 2035, 1899, 1456, 2013)

println("Orginal array:")

for ( x <- nums1) {

print(s"${x}, ")

}

var result1= test(nums1)

println("\nReversed array:")

for ( x <- result1) {

print(s"${x}, ")

}

var nums2 = Array(1789, 2035, 1899, 1456)

println("\nOrginal array:")

for ( x <- nums2) {

print(s"${x}, ")

}

var result2= test(nums2)

println("\nnReversed array:")

for ( x <- result2) {

print(s"${x}, ")

}

}

}

Programs on List

Write a Scala program to delete element(s) from a given List.

object Scala\_List

{

def main(args: Array[String]): Unit =

{

val nums = List(1, 3, 5, 7, 9, 11, 14, 12)

println("Original list:")

println(nums)

//As scala List is immutable, so we can’t delete elements from it, but

//filter out element(s) as per requirement.

println("Filter out 3 from the above list:")

val nums1 = nums.filter(\_ != 3)

println(nums1)

println("Filter out numbers which are greater than 10:")

val nums2 = nums.filter(\_ > 10)

println(nums2)

}

}

Write a Scala program to iterate over a list to print the elements and calculate the sum and product of all elements of this list

object Scala\_List

{

def main(args: Array[String]): Unit =

{

//Iterate over a list

val nums = List(1, 3, 5, 7, 9)

println("Iterate over a list:")

for( i <- nums)

{

println(i)

}

println("Sum all the items of the said list:")

//Applying sum method

val result = nums.sum

println(result)

println("Multiplies all the items of the said list:")

val result1 = nums.product

println(result1)

}

}

Write a Scala program to find the largest and smallest number from a given list.

object Scala\_List

{

def main(args: Array[String]): Unit =

{

//Iterate over a list

val nums = List(1, 3, 5, 7, 9, 11, 14, 12)

println("Original list:")

println(nums)

println("Largest number of the said list:")

println(nums.max)

println("Smallest number from the said list:")

println(nums.min)

}

}

 Write a Scala program to remove duplicates from a given list.

object Scala\_List

{

def main(args: Array[String]): Unit =

{

val nums = List(1, 3, 5, 2, 7, 9, 11, 5, 2, 14, 12, 3)

println("Original list:")

println(nums)

val result1 = nums.distinct

println("Unique elements of the said list:")

println(result1)

val chars = List("a", "a", "b", "c", "d", "c", "e", "f")

println("Original list:")

println(chars)

val result2 = chars.distinct

println("Unique elements of the said list:")

println(result2)

}

}

Write a Scala program to find the first and last element of given list.

object Scala\_List

{

def main(args: Array[String]): Unit =

{

val colors = List("Red", "Blue", " Black ", "Green", " White", "Pink")

println("Original list:")

println(colors)

println("First element of the said list: " + colors.head)

println("Last element of the said list: " + colors.last)

}

}

Write a Scala program to find the index of an element in a given list.

object Scala\_List

{

def main(args: Array[String]): Unit =

{

val colors = List("Red","Blue","Black","Green","White")

println("Original lists:")

println(colors)

println("Index of 'Red':", colors.indexOf("Red"))

println("Index of 'Blue':", colors.indexOf("Blue"))

println("Index of 'Black':", colors.indexOf("Black"))

println("Index of 'Green':", colors.indexOf("Green"))

println("Index of 'White':", colors.indexOf("White"))

}

}

Write a Scala program to find the even and odd numbers from a given list.

object Scala\_List

{

def main(args: Array[String]): Unit =

{

val nums = List(1, 2, 3, 4, 5, 7, 9, 11, 14, 12, 16)

println("Original list:")

println(nums)

val even\_nums = nums.filter(\_ % 2 ==0)

println("Even number of the said list:")

println(even\_nums)

val odd\_nums = nums.filter(\_ % 2 != 0)

println("Odd number of the said list:")

println(odd\_nums)

}

}

Write a Scala program to reverse a given list.

object scala\_basic {

def main(args: Array[String]): Unit = {

val nums = List(1,2,3,4,5,6,7,8,9,10)

println("Original List")

println(nums)

println("Reversed the said list:")

println("Using reverse() function:")

println(nums.reverse)

println("Using for loop:")

for(n<-nums.reverse)

{

print(n)

print(" ")

}

}

}

Write a Scala program to check a given list is a palindrome or not

object Scala\_List {

def is\_Palindrome[A](list\_nums: List[A]):Boolean = {

list\_nums == list\_nums.reverse

}

def main(args: Array[String]): Unit = {

println("Result: " + is\_Palindrome(List(1,2,3,4,3,2,1)));

println("Result: " + is\_Palindrome(List(1,2,3)));

}

}

Write a Scala program to count the number of occurrences of each element in a given list.

object Scala\_List {

def list\_elemnt\_occurrences[A](list1:List[A]):Map[A, Int] = {

list1.groupBy(el => el).map(e => (e.\_1, e.\_2.length))

}

def main(args: Array[String]): Unit = {

println(list\_elemnt\_occurrences(List(1,1,1,2,2,3,6,4,4,6,1,6,2)))

println(list\_elemnt\_occurrences(List("Red", "Green", "White", "Black", "Red", "Green", "Black")))

}

}