|  |
| --- |
|  |
| Scala Programming Assignment |
| 20-PBD-002 |
| Shraddha P Jain |
|  |
| **11-Nov-21** |

|  |
| --- |
|  |

Scala Programs

# Scala Basic Programs

1. **Write a Scala program to print "Hello, world" and version of the Scala language.**

object Hello extends App {

println("Hello, world")

println("Scala language: "+util.Properties.versionString)

}

1. **Write a Scala program to compute the sum of the two given integer values. If the two values are the same, then return triples their sum.**

object scala\_basic {

def test(x:Int, y:Int) : Int =

{

if (x == y) (x + y) \* 3 else x + y

}

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

println("Result: " + test(1, 2));

println("Result: " + test(2, 2));

}

}

1. **Write a Scala program to get the absolute difference between n and 51. If n is greater than 51 return triple the absolute difference.**

object scala\_basic {

def test(x:Int) : Int =

{

val abs\_Diff = Math.abs(x - 51)

if (x > 51) 3 \* abs\_Diff else abs\_Diff

}

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

println("Result: " + test(60));

println("Result: " + test(40));

}

}

1. **Write a Scala program to check two given integers, and return true if one of them is 30 or if their sum is 30.**

object scala\_basic {

def test(x:Int, y:Int) : Boolean =

{

x == 30 || y == 30 || x + y == 30

}

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

println("Result: " + test(30, 0));

println("Result: " + test(25, 5));

println("Result: " + test(30, 20));

println("Result: " + test(25, 20));

}

}

1. **Write a Scala program to check a given integer and return true if it is within 20 of 100 or 300.**

object scala\_basic {

def test(x:Int) : Boolean =

{

Math.abs(100 - x) <= 20 || Math.abs(300 - x) <= 20

}

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

println("Result: " + test(115));

println("Result: " + test(200));

println("Result: " + test(250));

println("Result: " + test(70));

}

}

## Scala String Exercises

1. **Write a Scala program to get the character at the given index within a given String. Also print the length of the string.**

**Sample Output:**

**Original String = Scala Exercises!**

**The character at position 0 is S**

**The character at position 10 is c**

**The character at position 15 is !**

**Length of the string: 16**

object Scala\_String {

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

var str = "Scala Exercises!";

println("Original String = " + str);

// Get the character at positions 0 and 10.

var index1 = str.charAt(0);

var index2 = str.charAt(10);

var index3 = str.charAt(15);

// Print out the results

println(s"The character at position 0 is ${index1}");

println(s"The character at position 10 is ${index2}");

println(s"The character at position 15 is ${index3}");

println(s"Length of the string: ${str.length}")

}

}

1. **Write a Scala program to concatenate a given string to the end of another string.   
   Sample Output:**

**Original strings:**

**String 1: Scala Exercises and**

**String 2: Python Exercises**

**The concatenated string: Scala Exercises and Python Exercises**

object Scala\_String {

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

val str1 = "Scala Exercises and ";

val str2 = "Python Exercises";

println("Original strings:")

println("String 1: " + str1);

println("String 2: " + str2);

// Concatenate the two strings together.

val str3 = str1.concat(str2);

// Display the new String.

println("The concatenated string: " + str3);

}

}

1. **Write a Scala program to check whether two String objects contain the same data.   
   Sample Output:**

**"Stephen Edwin King" equals "Stephen Edwin  King"? false**

**"Stephen Edwin King" equals "Stephen Edwin King"? true**

object Scala\_String {

def test(str1: String, str2: String): Boolean = {

str1.equals(str2)

}

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

val columnist1 = "Stephen Edwin King";

val columnist2 = "Stephen Edwin King";

val columnist3 = "Stephen Edwin King";

// Are any of the above Strings equal to one another?

val equals1 = test(columnist1,columnist2)

val equals2 = test(columnist1,columnist3)

// Display the results of the equality checks.

System.out.println("\"" + columnist1 + "\" equals \"" +

columnist2 + "\"? " + equals1);

System.out.println("\"" + columnist1 + "\" equals \"" +

columnist3 + "\"? " + equals2);

}

}

**4. Write a Scala program to compare a given string to another string, ignoring case considerations.   
Sample Output:**

**"Stephen Edwin King" equals "Stephen Edwin  King"? false**

**"Stephen Edwin King" equals "Stephen edwin king"? true**

object Scala\_String {

def test(str1: String, str2: String): Boolean = {

str1.equalsIgnoreCase(str2)

}

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

val columnist1 = "Stephen Edwin King";

val columnist2 = "Stephen Edwin King";

val columnist3 = "Stephen edwin king";

// Are any of the above Strings equal to one another?

val equals1 = test(columnist1,columnist2)

val equals2 = test(columnist1,columnist3)

// Display the results of the equality checks.

System.out.println("\"" + columnist1 + "\" equals \"" +

columnist2 + "\"? " + equals1);

System.out.println("\"" + columnist1 + "\" equals \"" +

columnist3 + "\"? " + equals2);

}

}

**5. Write a Scala program to replace a specified character with another character.   
Sample Output:**

**Original string: The quick brown fox jumps over the lazy dog.**

**New String: The quick brown fox jumps over the lazy fog.**

object Scala\_String {

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

val str = "The quick brown fox jumps over the lazy dog.";

// Replace all the 'd' characters with 'f' characters.

val new\_str = str.replace('d', 'f');

// Display the strings for comparison.

println("Original string: " + str);

println("New String: " + new\_str);

}

}

# Scala List Exercises

1. **Write a Scala program to get the difference between two given lists.**

object Scala\_List

{

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

{

val list1 = List("Red","Blue","Blue","Green","Black")

val list2 = List("Blue","White")

println("Original lists")

println(list1)

println(list2)

println("Difference of the said two lists(list1-list2):")

val temp = list2.toSet

val result = list1.filterNot(temp)

println(result)

println("Difference of the said two lists(list2-list1):")

val temp1 = list1.toSet

val result1 = list2.filterNot(temp1)

println(result1)

}

}

1. **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)

}

}

1. **Write a Scala program to merge (concatenate) given lists.**

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"))

}

}

1. **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 nums1 = List(1,3,5,7,9)

val nums2 = List(2,4,6,8,10)

println("Original Lists:")

println(nums1)

println(nums2)

println("Merge the said two lists using the ++ method:")

val nums\_1 = nums1 ++ nums2

println(nums\_1)

println("Using ::: way:")

val nums\_2 = nums1 ::: nums2

println(nums\_2)

println("Using concat method:")

val nums\_3 = List.concat(nums1, nums2)

println(nums\_3)

}

}

1. **Write a Scala program to find the nth element of 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) } }

# Scala Array Exercises

1. **Write a Scala program to find the common elements between two arrays of strings.**

object Scala\_Array {

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

var nums1 = Array(2,4,5,7,9)

var nums2 = Array(2,3,5,6,9)

//Call the following Java class for some array operation

import java.util.Arrays;

println("Original Array1 : "+Arrays.toString(nums1));

println("Original Array2 : "+Arrays.toString(nums2));

println("Common elements of the said two arrays:")

var i = 0

var j =0;

for (i <- 0 to nums1.length-1)

{

j=0

for (j <- 0 to nums2.length-1)

{

if(nums1(i) == nums2(j))

{

print(s"${nums1(i)}, ")

}

}

}

}

}

1. **Write a Scala program to remove duplicate elements from an array of strings.**

object Scala\_Array {

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

var my\_array = Array("bcd", "abd", "jude", "bcd", "oiu", "gzw", "oiu");

println("Orginal array:")

for ( x <- my\_array) {

print(s"${x}, ")

}

var f =0

for (i <- 0 to my\_array.length-1)

{

var x = f+1;

for (j <- x to my\_array.length-1)

{

if(my\_array(f) == my\_array(x) && (f != x) )

{

println("\nDuplicate Element: "+my\_array(x));

}

x=x+1;

}

f=f+1;

}

}

}

1. **Write a Scala program to find the number of even and odd integers in a given array of** **integers.**

object scala\_basic {

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

var array\_nums = Array(5, 7, 2, 4, 9)

println("Original array:")

for (x <- array\_nums) {

print(s"${x}, ")

}

var ctr = 0;

for (i <- 0 to array\_nums.length - 1) {

if (array\_nums(i) % 2 == 0)

ctr=ctr+1

}

println("\nNumber of even numbers : " + ctr);

println("Number of odd numbers : " + (array\_nums.length - ctr));

}

}

1. **Write a Scala program to compute the average value of an array element except the largest and smallest values.**

object scala\_basic {

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

var array\_nums = Array(5, 7, 2, 4, 9);

println("Original array:")

for (x <- array\_nums) {

print(s"${x}, ")

}

var max = array\_nums(0)

var min = array\_nums(0)

var sum: Double = 0

for (i <- 0 to array\_nums.length - 1)

{

sum = sum + array\_nums(i);

if(array\_nums(i) > max)

max = array\_nums(i);

else if(array\_nums(i) < min)

min = array\_nums(i);

}

val x: Double = ((sum-max-min) / (array\_nums.length - 2));

println(s"\nAverage value the said array elements except the largest and smallest values: ${x}");

}

}

1. **Write a Scala program to remove the duplicate elements of a given sorted array and return the new length of the array.**

object Scala\_Array {

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

var index = 1;

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

if (nums(i) != nums(index))

{

index += 1

nums(index) = nums(i)

}

}

index;

}

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

val nums = Array(20, 20, 30, 40, 50, 50, 50, 50, 60, 60);

println(s"Original array length: ${nums.length}");

println("Array elements are: ");

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

{

print(s"${nums(i)} ");

}

println(s"\nThe new length of the array after removing the duplicate elements is: ${test(nums)}");

}

}