start-master.sh

source ~/.profile

ls -l /opt/spark

source ~/.bashrc

spark-shell

object Factorial{

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

val num = 5

val factorial = 1

for (i<-1 to num){

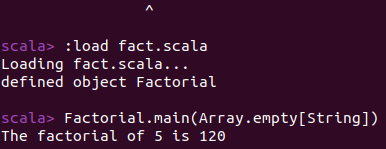
factorial\*=i

}

println(s”the factorial of $num is $factorial”)

}

}



Reverse the String

object ReverseString {

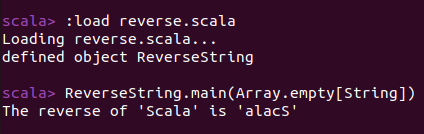
def main(args: Array[String]): Unit = {

val str = "Scala"

val reversed = str.reverse

println(s"The reverse of '$str' is '$reversed'")

}



object FindLargest {

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

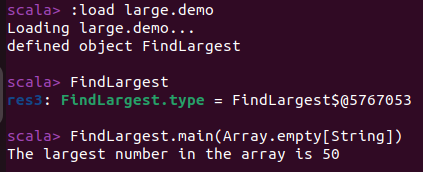
val numbers = Array(10, 20, 30, 40, 50)

val largest = numbers.max

println(s"The largest number in the array is $largest")

}

}



object SumOfTwoNumbers {

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

val num1 = 10

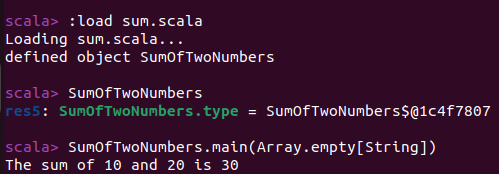
val num2 = 20

val sum = num1 + num2

println(s"The sum of $num1 and $num2 is $sum")

}

}



object AddTwoNumbers {

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

println("Enter the first number:")

val num1 = StdIn.readInt()

println("Enter the second number:")

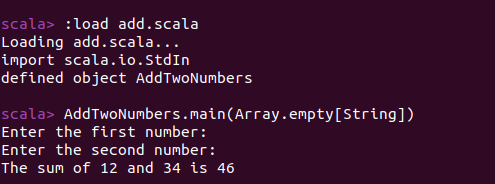
val num2 = StdIn.readInt()

val sum = num1 + num2

println(s"The sum of $num1 and $num2 is $sum")

}

}



import scala.io.StdIn

object SimpleCalculator {

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

println("Enter the first number:")

val num1 = StdIn.readDouble()

println("Enter an operator (+, -, \*, /):")

val operator = StdIn.readChar()

println("Enter the second number:")

val num2 = StdIn.readDouble()

val result = operator match {

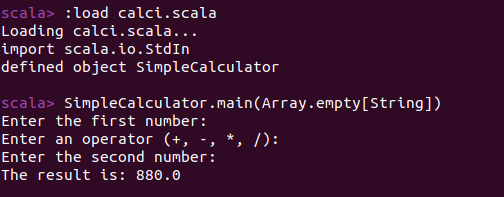
case '+' => num1 + num2

case '-' => num1 - num2

case '\*' => num1 \* num2

case '/' => if (num2 != 0) num1 / num2 else "undefined (division by zero)"

case \_ => "Invalid operator"

}