Quiz 4- HIMABINDU MELACHURU

1. Ruby Program that prints the Time in different time zones

irb(main):073:0> def time

irb(main):074:1> time = Time.new

irb(main):075:1> puts "current time: " + time.inspect

irb(main):076:1> puts time.zone

irb(main):077:1> t = time.to\_a

irb(main):078:1> puts Time.utc(\*t)

irb(main):079:1> puts Time.gm(\*t)

irb(main):080:1> end

=> :time

irb(main):081:0> time

current time: 2015-11-20 19:08:06 -0800

PST

2015-11-20 19:08:06 UTC

2015-11-20 19:08:06 UTC

2. Write a program that iterates over an array and builds a new array that is the result of incrementing each value in the original array by a value of 2. You should have two arrays at the end of this program, The original array and the new array you've created.Print both arrays to the screen using the p method instead of puts.

irb(main):070:0> def array

irb(main):071:1> a = [1,2,3,4,5,6]

irb(main):072:1> new = a.map {|x| x += 2}

irb(main):073:1> p a

irb(main):074:1> p new

irb(main):075:1> end

=> :array

irb(main):076:0> array

[1, 2, 3, 4, 5, 6]

[3, 4, 5, 6, 7, 8]

=> [3, 4, 5, 6, 7, 8]

3. Ruby program to find the leap year when start and end year are given.

irb(main):038:0> def year

irb(main):039:1> puts "Enter starting year:"

irb(main):040:1> starting\_year = gets.chomp.to\_i

irb(main):041:1> puts "Enter ending year:"

irb(main):042:1> ending\_year = gets.chomp.to\_i

irb(main):043:1> year = starting\_year

irb(main):044:1> while true

irb(main):045:2> if (year%4==0)

irb(main):046:3> if (year%100!=0) || (year%400 ==0)

irb(main):047:4> puts year.to\_s + "is a Leap Year"

irb(main):048:4> end

irb(main):049:3> end

irb(main):050:2> year = year +1

irb(main):051:2> break if year >= ending\_year

irb(main):052:2> end

irb(main):053:1> end

=> :year

irb(main):054:0> year

Enter starting year:

1990

Enter ending year:

2015

1992is a Leap Year

1996is a Leap Year

2000is a Leap Year

2004is a Leap Year

2008is a Leap Year

2012is a Leap Year

=> nil

4. Ruby program that takes a numerical value and give the output as Roman number

irb(main):001:0> def roman

irb(main):002:1> conversion = Hash[ 1000 => "M", 900 => "CM", 500 => "D", 400 => "CD", 100 => "C", 90 => "XC", 50 => "L", 40 => "XL", 10 => "X", 9 => "IX", 5 => "V", 4 => "IV", 1 => "I"]

irb(main):003:1> puts "Enter number "

irb(main):004:1> num = gets.to\_i

irb(main):005:1> if (num > 0) || (num < 3999)

irb(main):006:2> conversion.keys.sort{ |a,b| b <=> a }.each do

irb(main):007:3\* |n|

irb(main):008:3\* while num >= n

irb(main):009:4> num = num - n

irb(main):010:4> print conversion[n]

irb(main):011:4> end

irb(main):012:3> end

irb(main):013:2> end

=> :roman

irb(main):017:0> roman

Enter number 3789

MMMDCCLXXXIX=> [1000, 900, 500, 400, 100, 90, 50, 40, 10, 9, 5, 4, 1]

5. Write a your own ruby program that uses a Queue

require 'thread'

test = Queue.new

test.enq(10)

puts "enqueing 10"

test.enq(12)

puts "enqueing 12"

test.enq(20)

puts "enqueing 20"

while ! test.empty?

popped = test.deq

puts "Popped: " + popped.to\_s

end

OUTPUT:

enqueing 10

enqueing 12

enqueing 20

popped: 10

popped: 12

popped: 20

6. Write your own ruby program that uses each\_with\_index method to iterate through an array that prints each index and value

irb(main):040:0> def array

irb(main):041:1> array = ["cat", "dog", "peacock", "monkey"]

irb(main):042:1> array.each\_with\_index {|val, index| puts "#{val} => #{index}" }

irb(main):043:1> end

=> :array

irb(main):044:0> array

cat => 0

dog => 1

peacock => 2

monkey => 3

=> ["cat", "dog", "peacock", "monkey"]

7. Ruby Program that prints if duplicates existing in a array

irb(main):016:0> def duplicate

irb(main):017:1> a = ["A", "B", "C", "B", "A", "F", "Z", "P", "Z"]

irb(main):018:1> a.select{ |e| a.count(e) > 1 }.uniq

irb(main):019:1> end

=> :duplicate

irb(main):020:0> duplicate

=> ["A", "B", "Z"]

8. Write a Ruby program that prints pascal triangle

9. Write a Ruby program that prints the length of the common string when two strings are compared.

irb(main):094:0> def equals

irb(main):095:1> puts "Enter a string value: "

irb(main):096:1> string1 = gets.chomp

irb(main):097:1> puts "Enter a second string value: "

irb(main):098:1> string2 = gets.chomp

irb(main):099:1> if (string1 == string2)

irb(main):100:2> puts " The length of the string is ",string1.length

irb(main):101:2> puts "They both are same"

irb(main):102:2> else

irb(main):103:2\* puts "They are different"

irb(main):104:2> end

irb(main):105:1> end

=> :equals

irb(main):106:0> equals

Enter a string value:

hima

Enter a second string value:

hima

The length of the string is

4

They both are same

=> nil

irb(main):107:0> equals

Enter a string value:

hima

Enter a second string value:

bindu

They are different