**Explorations in Ruby: Practical**

**Practical 2**

**Student Name- Vipin Hans**

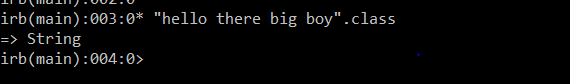
**Student Number- 17201230**

1) In irb and each of the primitives ***class*** and ***instance\_of?*** test the following to see

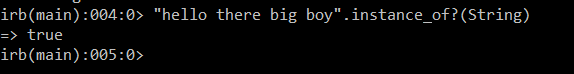
what types of object they are and explain why you get the answers you do:

Answer-

a. “hello there big boy”

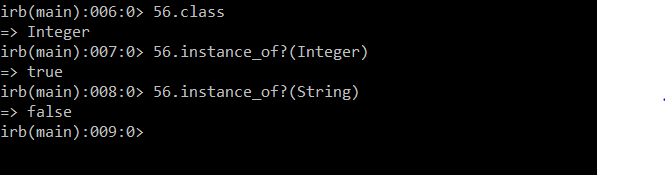


The .class or .class.name returns the Class to which an object belongs.

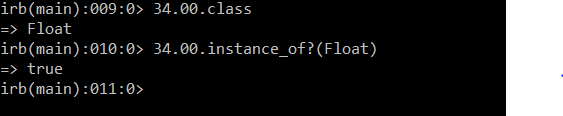


if the object is an instance of that exact class, not a subclass.

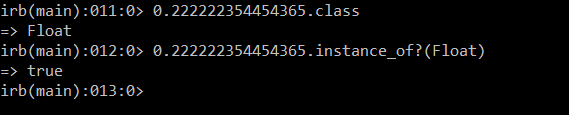
b. 56



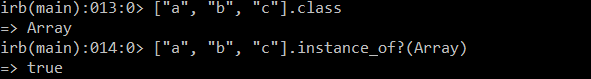
c. 34.00

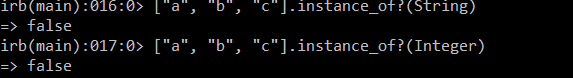


d. 0.222222354454365

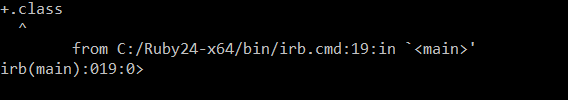


e. [“a”, “b”, “c”]

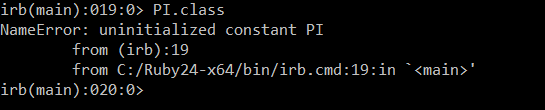




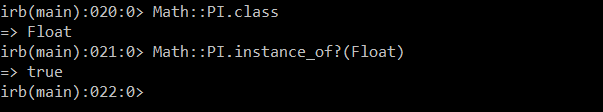
f. +



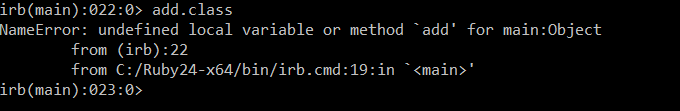
g. PI



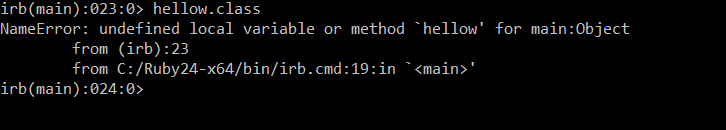
h. Math::PI



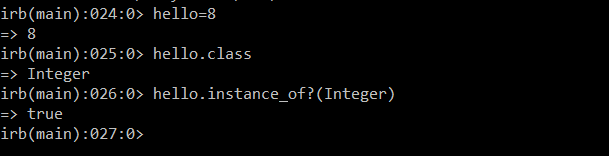
i. add



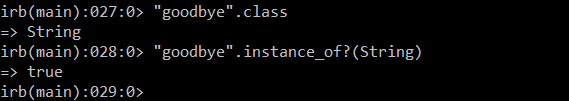
j. hellow



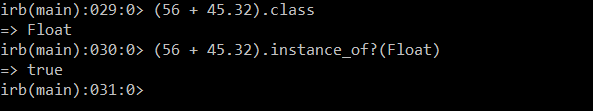
k. hello = 8 and then check hello with ***class***



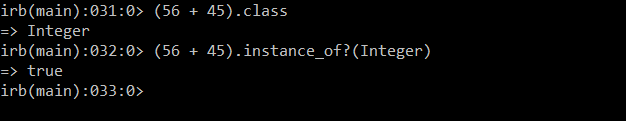
l. “goodbye”



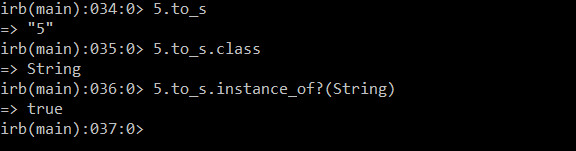
m. (56 + 45.32)



n. (56 + 45)

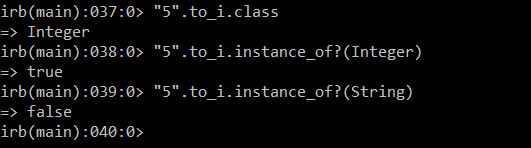


o. 5.to\_s

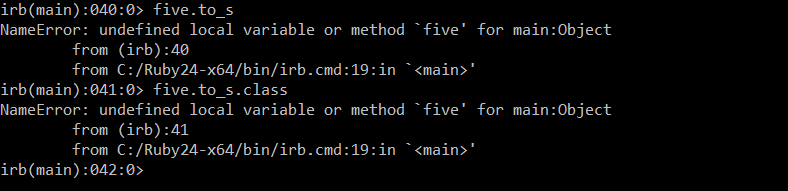


5.to\_s converted this to String data type and then performing .class on it returned the new primitive data type class which is String. Instance\_of?(String) returned Boolean value true or false.

p. “5”.to\_i



q. five.to\_s



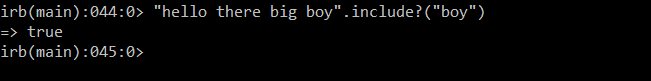
2) In irb what happens when you evaluate the following. Try to predict it before

trying them:

Answer-

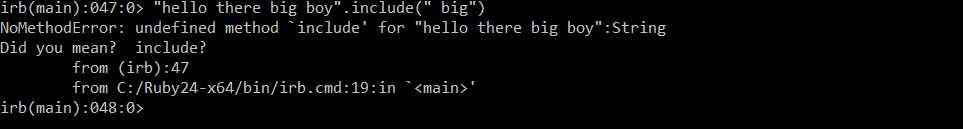
1. “hello there big boy”.include?(“boy”)

Include? Returns true or false based upon if the part of string “boy” is included in String object “hello there big boy”. Therefore, in such case the value returned should be true.



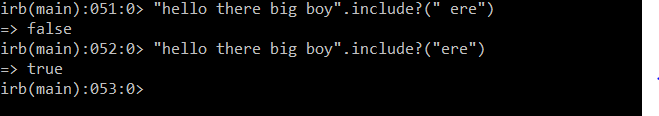
1. “hello there big boy”.include(“ big”)

It should return some syntax error or method not defined since built-in method is include? And not include

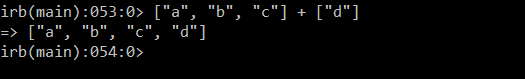


1. “hello there big boy”.include?(“ ere”)

It will return **false** as there is a space “ ere” in the string to be compared. If there were no space “ere” it should have been matched up with “there” in the String object “hello there big boy” and would have returned true in this scenario.



1. What happens when you evaluate: [“a”, “b”, “c”] + [“d”]



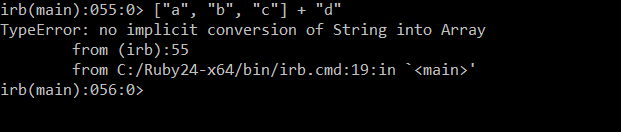
Added new element to array. Array concatenation.

Array1 + Array2

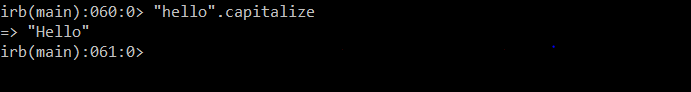
Array1.concat Array2

1. What happens when you evaluate: [“a”, “b”, “c”] + “d”

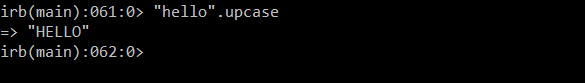
It won’t be added as the “d” object is of string type and not enclosed in [].



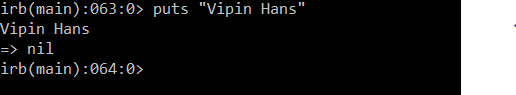
1. Is there an easy way to capitalise words, so “hello” becomes “Hello” ?



1. In the same vein, make “hello” “HELLO”.



1. Write a command to print out your name.



1. Write a method to print out your name.

**//**Method to print my name

def printname

puts “Vipin Hans”

end

1. Write a method to print out any name.

def inputname

puts "Enter your name: "

name=gets

puts "Your name is: "+ name

end

k. Set up the varibles, *maxi*, *dick* and *twinko* so that they are all assigned

numbers but two of them are assigned to the same numbers. Then

show with a series of equality tests which ones actually have the same

value.

def equalitytest

maxi=10

dick=10

twinko=100

if (maxi==dick)

print "maxi and dick contains same value"

elsif (dick==twinko)

print "dick and twinko contains same value"

elsif (twinko==maxi)

print "twinko and maxi contains same value"

else

print "All values are different"

end

end

l. If you change the variables with the same number to be a Float and

Fixnum does it change the results of the equality tests ?

It doesn’t really changes the result of program.

def equalitytest

maxi=10

dick=10.0

twinko=100

if (maxi==dick)

print "maxi and dick contains same value"

elsif (dick==twinko)

print "dick and twinko contains same value"

elsif (twinko==maxi)

print "twinko and maxi contains same value"

else

print "All values are different"

end

end

m. Do a version of these test using strings rather than numbers.

def equalitytest

maxi="maxi"

dick="maxi"

twinko="twinko"

if (maxi==dick)

print "maxi and dick contains same value"

elsif (dick==twinko)

print "dick and twinko contains same value"

elsif (twinko==maxi)

print "twinko and maxi contains same value"

else

print "All values are different"

end

end

3) What’s a predicate ?

It returns boolean values which is either true or false such as include? And instance\_of? In ruby such methods end with a question mark.

4) Define your own adding method that always adds 5 and 6 together.

So, my\_add\_five\_to\_six => 11.

def addnumbers

firstnumber=5

secondnumber=6

finalanswer=firstnumber+secondnumber

puts "Addition of 5 and 6 is: "+finalanswer.to\_s

end

5) Put this defined method in a file and call it using the ruby command outside of

Irb

* Save the file in any directory.
* Add Ruby installation bin path to environment variables.
* Open command prompt, and navigate to the respective directory where program is located.
* Execute command ruby filename.rb