Explorations in Ruby: Practicals
Practical 1

Name:

Date:

It is assumed that you have been through all the examples mentioned in the lecture, you need to have typed all of the stuff mentioned there (at least once) and be familiar with what was done.

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:
    a. "hello there big boy" String, and it return true.
    b. 56
    c. 34.00
    d. 0.222222354454365
    e. ["a", "b", "c"]
    f. +
    g. PI
    h. Math::PI
    i. add
    j. hello
    k. hello = 8 and then check hello with ***class***
    l. "goodbye"
    m. (56 + 45.32)
    n. (56 + 45)
    o. 5.to_s
    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:
    a. "hello there big boy".include?("boy")
    b. "hello there big boy".include(" big")
    c. "hello there big boy".include?(" ere")

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

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

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

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

h. Write a command to print out your name.

i. Write a method to print out your name.

j. Write a method to print out any name.

k. Set up the varibles, *maxi*, *dick* and *twink* 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.

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 ?

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

3) What’s a predicate ?

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

    So, my_add_five_and_six => 11.

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