Quiz 0.2b Rubric

1. (3 points) What will Scheme print in response to the following expressions or group of

expressions? Assume that they are typed in sequence, so definitions affect later interactions.If an expression produces an error message, you may just say “error''; you don't have to provide the exact text of the message.

(bf '(yowza))

()

(bf 'yowza)

owza

(define sent '(cs is fun))

(bf sent)

(is fun)

sent

(cs is fun)

( + ‘math sent)

error

(define (three-equals-five?)

(= 3 5))

(three-equals-five?)

#f

Half a point for each part all or nothing.

1. (3 points) Write a procedure hyphenate that has takes two words as arguments and hyphenates them.

> (hyphenate ‘fast ‘tracked)

fast-tracked

> (hyphenate ‘one ‘two)

one-two

(define (hyphenate wd1 wd2)

(word wd1 `- wd2))

1 point – uses word not sentence

1 point – quotes the hyphen

1 point has the correct number of arguments

-1 point – for every syntax error not listed above

1. (4 points) Write a procedure called singular that takes a word and makes it singular. We

will write a simple version where we make a word ending in “s” or “es” singular by removing the “s” or “es” and if the word doesn’t end in either of those then we assume it is already singular and leave it unchanged. This won’t work for everything. For example “places” would become “plac” but you don’t need to worry about those cases. Here are more examples:

> (singular ‘cans)

can

> (singular ‘boxes)

box

> (singular ‘car)

car

(define (singular wd)

(if (equal? (last wd) ‘s)

(if (equal? (last (bl wd)) ‘e)

(bl (bl wd))

(bl wd))

wd))

2 points – correctly checking for ending in s or es.

1 point – using equal? vs =

1 point – removing the correct amount from the input

-1 point – for every syntax error not listed above