DrRacket Tips and Traps

Common errors

- **Too many parentheses** - Since an open parenthesis, "(", indicates the beginning of a function, too many paranthesis will cause DrRacket to try to evaluate the result of the inner paranthesis as if it were a function, causing the error message: illegal application: first term in application must be a function name
- **Too few parenthesis** - Such as leaving off the correct number of parenthesis at the end of your function definition. The parenthesis-matching capabilities of DrRacket are designed to minimize this error, so pay attention to the highlighting that takes place as you type in parenthesis. The error message generated is syntax error: missing close paren or syntax error: too many close parens
- **Misspellings** - Typographical errors will give a reference to undefined identifier: [xxxx] error.
- **Improper define syntax** - An incorrect syntax in a define statement may lead to a define: malformed definition error.

Error Message Interpretations

- **illegal application: first term in application must be a function name** - The first argument after the parenthesis must be a function. This error is generated by ((+ 3 4)), where there is an extra set of parentheses.
- **syntax error: missing close paren** - The closing paranthesis is missing. The mis-matched opening paranthesis is highlighted by DrRacket. This error is generated by (+ 5 (* 3 4)).
- **syntax error: too many close parens** - The opening paranthesis is missing. The mis-matched closing paranthesis is highlighted by DrRacket. This error is generated by + 5 (* 3 4)).
- **illegal application: first term in application is a function-bound identifier** - This error occurs in the "Beginning Student" level because function input parameters that are functions is not allowed. For example, it is generated by (define (fn x y) (x y)) because x needs to be a function.
- **reference to undefined identifier: [xxxx]** - This means that [xxxx] was misspelled. The misspelled word is highlighted by DrRacket. Note that this may mean that there is a space missing between two words. For example, the error is generated by (+ 4 5) where *4 is "misspelled" due to a missing space.
- **define: malformed definition** - Incorrect syntax used in a define statement. For instance, (define fn(x) (* x 5)) will generate the error. It should be (define (fn x) (* x 5)).

What does it mean when DrRacket says...?

- *#i[number]* - the number is an inexact value. That is, math errors may result from it because an approximation to the correct value has been used. For example pi prints out as #i3.141592653589793.