Discussion Questions for "Lambda Expressions"

1. Does Java fully support first class functions with Lambda Expressions? In what ways are Lambda expressions first class functions, and in what ways are they still second class?
2. A Lambda expression can use the lambda expression arguments but can also use some other values.
    a. Lambda expressions can use field values. When the lambda expression is evaluated, how does the expression know how to read the field values?
    b. Lambda expressions can use final local variables. How does the expression know how to read the local variable when it is executed in an entirely different context?
    c. Lambda expressions cannot use local variables which are mutable. Why not?
    d. Can lambda expressions use static variables? If so, how does the expression know how to access the static variable's value?
    e. Lambda expressions cannot use local variables that are available when the expression is evaluated. Why not?
3. The concept of capturing the environment in which a first class expression is defined so that the environment can be used when the expression is evaluated is a concept which makes first class functions extremely powerful, but can also lead to counter-intuitive results at times. The Java language developers chose to implement a fairly limited capture, keeping only the value of "this", as well as constants. What else could have been captured? What are the advantages and disadvantages of capturing more or less program state when the lambda expression is defined?