1. (7 pts) Reduce the following, listing when you are using an α or β reduction at each step.   
   **λxy.(y x) y λy.2y** (where 2y = 2 times y)

λxy.(yx) y λy.(2y) [START]

λxy.(yx) u λy.(2y) [ α ]

λy.(yu) λy.(2y) [ β ]

λy.(yu) λt.(2t) [ α ]

λt.(2t) u [ β ]

2u [ β ]

1. (7 pts) Reduce the following, listing when you are using an α or β reduction at each step.  
   **λxy.(y x x) 5 λs.( s + s)**

λxy.( y x x ) 5 λs.(s + s) [START]

λy.( y 5 5 ) λs.(s + s) [ β ]

λs.(s + s) 5 5 [ β ]

5 + 5 5 [ β ]

1. (7 pts) Let T = λxy.x, F = λxy.y, and the OR operator,   
   ∨ = λxy.x(λuv.u) y = λxy.xTy. Show that ∨FF = F.

V F F = F  λxy.(x T y) λxy.(y) λxy.(y) = λxy.(y)

* λxy.(x T y) λxy.(y) λxy.(y) [α]  λxy.(x T y) λuv.(v) λxy.(y)    
  [β]  λy.( λuv.(v) T y) λxy.(y)  *\*\*now work inside parentheses of leftmost λy function…\*\**
  + λuv.(v) T y  T = λst.(s)  λuv.(v) λst.(s) y  [β] *\*\*the λu just throws away first given expression (* λst.(s) *)\*\**  λv.(v) y  [β]  y  thus: λuv.(v) T y = y…
* λy.(y) λxy.(y)  [α]  λm.(m) λxy.(y)  [β]  λxy.(y)    
  thus: V F F = λxy.(x T y) λxy.(y) λxy.(y) = λxy.(y) which when plugged into VFF = F you get: **λxy.(y) = λxy.(y)**; which proves VFF = F is **true.**

1. (4 pts) Is this an example of application? Justify your answer.  *x    y*

Yes, this is an example of an application. It is *applying* expression 1 (x) to expression 2 (y).

1. (5 pts) Provide an example of a function with no bound variables and at least one free variable. Is this a function according to the Lambda calculus definition of a function? Justify your answer.

**λx.(y)**

Yes, this is a function according to λ calculus because it has the λ followed by a variable parameter (x) which attempts to bind a variable (but no variable is bound because x is not used in the expression) followed by an expression, y (which is free).