Logan McCaul

2.

a. Pi at line 4 is bound by line 3 because that version of pi is declared as a val inside of the function circumference therefore is restricted to the functions scope. The pi at line 7 is bounded by line 1 because it is declared outside of the function therefore it's scope is the entire function itself.
b. X at line 3 is bounded at line 2 because it is a variable that is passed into the function therefore it is restricted to that functions scope. X at line 6 is also bounded by line 2 because the scope of the x is the function f. X at line 10 is bounded by line 8 because x is declared inside that function and the x in line 10 is interacting with the declaration of val x = y+1. X at line 13 is bound by x at line one because it is using the originally declared x and passing it into function f, but that x's scope is the whole process.

3. The return statement for function g would have to be something that could handle coordinates, potentially an array of ints. I determined this by looking at what the function was doing and how it was using it's passed in values.

val(a,b): arrays because

1: int

(x,3): multiple ints