I think a simple concrete example will help. Suppose we have a functor

data F a = One a | Two a a | Two' a a | Three Int a a a

with the obvious fmap. Then Free F a is the type of trees whose leaves have type a and whose nodes are tagged with One, Two, Two' and Three. One-nodes have one child, Two- and Two'-nodes have two children and Three-nodes have three and are also tagged with an Int.

Free F is a monad. return maps x to the tree that is just a leaf with value x. t >>= f looks at each of the leaves and replaces them with trees. When the leaf has value y it replaces that leaf with the tree f y.

A diagram makes this clearer, but I don't have the facilities for easily drawing one!