

Guards

When you want to inspect the value of an input instead of just pattern matching on its constructors you should use guards.

```
compare :: Eq a => a -> a -> String
compare x y
  | x == y      = "They are the same!"
  | otherwise   = "Different"
```

So the syntax. As ever you put the name of the function so that it knows this part of the definition is for it. Then you give names to the parameters you are taking in, here we have just called them `x` and `y`. Unlike pattern matching you don't then have an equals, instead you go to a newline and place a guard (`|`). Here we return to the familiar pattern of the LHS of the equals describing the case in which you want the code on the RHS to execute. The LHS can be any logical expression involving your variables you want and each guard represent a different case.