How can you load files in `ghci`?

Use `:load` (`:l`) or `:reload` (`:r`).
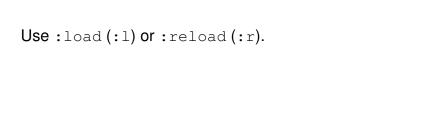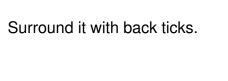
What is the key restriction of `ghci`?

It can only evaluate expressions not declarations.

How can you use a function in infix style?

Surround it with back ticks.

How can you convert an integral type into a floating type?

And vice-versa?

```
i :: Integer
i = 1
f :: Float
f = fromIntegral i
```

`round`, `floor`, `ceiling` can be used to convert the other way.

What's the difference between / and `div`?

`/` only does floating point division. `div` does integer division.

What is `otherwise`?

The value `True`. It's just a way of matching all cases in a more readable way.

How are lists implemented?

As singly linked lists. not arrays. So don't forget random access is $O(n)$.

Why is it okay to write many small functions that transform whole data structures and then combine them?

Haskell's lazy evaluation strategy allows it to interleave the code, only generating and storing elements as needed.