Concatenate strings.

```
str1 ++ str2
```

What are strings, really?

Lists of characters.

How is negation done?

/= and unary `not`

Give the conditional syntax.

What is unusual about Haskell conditionals?

```
if bool then expr1 else expr2
```

The else clause is required, to ensure an interesting result for the whole `if` expression.

Like in Java, Boolean types are ...

... strict.

Find the type of an expression

Make `ghci` print all types.

`:t expr` shows a value's type in `ghci`.

`:set +t` makes it permanent for that session.

What are the two types of a function?

The optional type declaration and the function declaration.

Create a module `MyMath` with the function `my_max`.

```
module MyMath where
    my_max :: Integer -> Integer -> Integer
    my_max x y = if x > y then x else y
```

How can you control flow of control to a function?

- Multiple function definitions.
- Guards.

Give the syntax of guards.

What is often used as the last guard?

Delay the equals sign, splitting up the argument lists.

```
func args
    | boolean1 = ...
    | boolean2 = ...
```

`otherwise` is an alias for true, often used as the last guard.

Give the syntax of tuples and lists.

Tuples use round parens, lists use brackets.