**Module 1 Knowledge**

When we communicate as humans and with humans, we use our shared knowledge of the world, on past conversations and context to make sense of what is being said. Computers on the other hand are machines, how do we capture instructions in a mechanical process?

That leads to an even deeper philosophical question which is if we want a computer to compute something, we can ask, well, what's the knowledge that it's going to use to do that computation?

This leads to an interesting distinction between what we consider knowledge, because we can divide knowledge into two types.

**Declarative knowledge**

Declarative knowledge-- those are statementsof fact, statements of truth, it is.

For example, if I were lecturing here at GW, I might have gone in ahead of time and strapped or taped some candy under one of the chairs inside the lecture hall.

A declarative piece of information simply says there's candy taped to the underside of a chair.

It doesn't tell you anything about how to find it. It doesn't tell you where to look for it.

You'd either do it in parallel by having every student reach under the chair or search the entire lecture hall. It's simply a statement of fact.

**Imperative knowledge is a recipe.**

It's how to gain knowledge or how-to information, and this gives us a sequence of steps to find a solution. Again, if I stuck candy under a chair inside the lecture hall, I might say to the lecturer, do the following—

* face the students at the front of the room.
* Count one, two, three rows.
* Start in the middle section on the left side, go in one chair.
* Look underneath it. Candy is there.

It's a recipe. It's a sequence of how to steps, very mechanical, to get things done, this is the knowledge a computer uses.

**Recipe**

* It is a sequence of simple steps.
* Flow of control, steps to tells us in which order to execute the steps.
* A means of determining when to stop.