

Summary – Variables, Constants and Types

What a variable, a constant and a type are, the four types we have met so far, and how to read a line of the Guessing Game's code word by word.

Things a program remembers

While a program runs, it has to remember things. The Guessing Game remembers the secret number, the guess on the screen, how many guesses you have made, the message it shows you, and whether the game is over.

Each thing it remembers is a **named value**: a name, such as `guessCount`, that holds a value, such as 0.

Variables and constants

A name that holds a value is a **variable** if its value can change, and a **constant** if it cannot.

- A variable's line starts with **var**.
- A constant's line starts with **let**.

```
let highestNumber: Int = 100
@State var guessCount: Int = 0
```

`highestNumber` is one of the game's settings, and it never changes while the game runs, so it is a constant. `guessCount` goes up with every guess, so it is a variable.

Types

The word after the colon is the name's **type**: what kind of value it holds. Here are the four types we have met so far.

Type	What it holds	A line from the Guessing Game
<code>Int</code>	a whole number	<code>@State var guessCount: Int = 0</code>
<code>String</code>	text	<code>@State var messageSymbol: String = "questionmark.circle.fill"</code>
<code>Color</code>	a colour	<code>@State var messageColor: Color = Color.indigo</code>
<code>Bool</code>	a yes-or-no answer: <code>true</code> or <code>false</code>	<code>@State var gameIsOver: Bool = false</code>

Reading a line of code

Example 1

Read this line word by word.

`@State var guessCount: Int = 0`

`@State` the screen depends on this, so the screen updates when it changes

`var` it is a variable: its value can change

`guessCount` its name

`: Int` its type: it holds a whole number

`= 0` the value it starts with

Put together: `guessCount` is a variable that holds a whole number, it starts at 0, and the screen updates when it changes.

Example 2

Why must this line start with `var`, and not `let`?

```
@State var secretNumber: Int = Int.random(in: lowestNumber...highestNumber)
```

A new game needs a new secret number, so `startNewGame()` must change `secretNumber`. A `let` can never change, so if this line started with `let`, Xcode would refuse to run the game.

Every name at the top of the Guessing Game

Name	let or var	Type	What it holds
lowestNumber, highestNumber	let	Int	the range of numbers
secretNumber	var	Int	the number to find
guess	var	Int	the guess on the screen, before you click Guess
guessCount	var	Int	how many guesses in this game
message, messageSymbol	var	String	what the card says, and the name of its symbol
messageColor	var	Color	the card's colour
gameIsOver	var	Bool	has the number been found?