So you're stuck on a problem

Congratulations! You're now a real programmer! Getting stuck during the problem-solving process is a core part of programming. But fear not: we have some troubleshooting methods to help you through it.

GOOGLING AN ERROR

One of the most important tools you need to be comfortable with as a programmer is Google. Copy-pasting errors into Google can quickly lead you to a fix, and Googling a concept can help you reach the information you're looking for. Use keywords of tools that you are using and the language you are working in like "JavaScript for loop" or "Python syntax for if statement" to hone your search.

PLAYING CHICKEN WITH A PROBLEM

Sometimes a problem is so specific that Google can't really help you -- at least not yet. The go-to here is to sit and stare down your problem: ***try something, run the code, repeat***. You'll be surprised how often you can straight-up elbow-grease your way to a solution. Remember, you can run your code as many times as you need without consequence, so add in variables to see what is popping up at different parts of your code and work through it step-by-step to see what might be going on.

WHEN ALL ELSE FAILS, ASK FOR HELP

One of the great things about programming is it is basically one massive team effort: you can ask questions and find answers online, as well as help answer other programmers' questions.

When asking a question online (on a forum, or even on Discord), in order to save everyone's time, there is specific information you need to supply in order for someone to replicate your problem and help you find a solution. These are:

1. **A summary of the problem.** This gives the person a general idea of what to expect. This also includes a brief description of the greater context of the code so the person can understand what this code is supposed to do.
2. **What computer you are using, its operating system, what IDE you used and what version of software you used.** This helps the person to see if the problem comes from the setup, rather than the code itself.

3. **The code itself.** This is an obvious one. The person needs to see your code in order to recreate the problem. Make sure your code includes descriptive variable names and comments where applicable.
4. **The error message.** Paste the error message you are getting when you run this code.
5. **What you've tried to fix it.** This is a super important one that not many people do, but it really helps for people to take your question seriously! You need to show that you have tried to fix the problem yourself before asking for help. This could include links to websites, other snippets of code, the suggestion your friend gave you, whatever. As long as you show that you have tried, people will be more than happy to help you out.

EXAMPLES

Below are two examples: one good, one bad. Both examples are asking for help with the same problem.

Example 1

My code is not working, can you help?

```
let var1 = "Peter";
let var2 = "Parker";
let var3 = `{var1} {var2}`;
if (var3  === "Peter Parker") {
    console.log("You're Spiderman!");
}
else {
    console.log("You're not Spiderman");
}
```

Example 2

Hi, I am creating a basic programme to practise if-else statements. I want the console to output "You're Spiderman!" if the name and surname together make "Peter Parker".

Computer: Dell Inspiron 13 7000 Series
Operating system: Windows
IDE: Visual Studio Code version 1.47.3
Node.js version 12.8.1

See the code below:

```
let name = "Peter";
let surname = "Parker";
let fullname = `{name} {surname}`;
if (fullname  === "Peter Parker") {
    console.log("You're Spiderman!");
}
else {
    console.log("You're not Spiderman");
}
```

The code runs so I am not getting an error message, but the console outputs "You're not Spiderman" when I need it to output "You're Spiderman!"

I have tried changing the triple equals (===) to a double equals (==), but that has not helped. I also tried to use straight concatenation for the variable **`fullname`** and changed the line to **`let fullname = name + " " + surname`** which worked, but I'm trying to get comfortable with template literals. I suspect my problem is on line 3, but I'm not sure how to fix it. Any help would be really appreciated!

Did you notice how detailed the second example is? It is much easier to help someone with a solution when they give that level of detail in the information. But also notice that, while the question is asked well with a lot of detail, a quick Google search of template literals would show that the format requires a dollar sign (**`$`**) before the curly braces (**`{}`**), and so line 3 should be **`let fullname = `${name} ${surname}`;`**. So often Google can help us solve our problems without needing to ask for help!

This is the structure if you want to pose a question on StackOverflow or any other forum site, but it would also be good practice to structure your questions to your mentor like this. It will help you get into good habits and allow them to help you much faster.