Running your first program: Hello World!

Introduction

Welcome back, coding explorers! Today is an exciting day - you're going to write your very first Python program! Programmers have a tradition of making their first program display the message "Hello, World!" This tradition goes back decades and is like saying "Hi!" to the world of programming.

Learning Goals

By the end of this lesson, you will:

- Write a simple Python program
- Run your code and see the results
- Learn how to save your program
- Make your program more interactive

Your First Python Program

Step 1: Open Thonny

Find the Thonny icon on your computer and click to open it. You should see a blank editor window at the top and a shell at the bottom.

Step 2: Write Your Code

In the top editor window, type this exactly as you see it:

```
print("Hello, World!")
```

That's it! Just one line of code. The `print()` function tells Python to display something on the screen. The message inside the quotation marks is what will be displayed.

Step 3: Run Your Program

Press the green "Run" button at the top of Thonny (or press F5 on your keyboard). Look at the shell window at the bottom. You should see:

```
Hello, World!
```

Congratulations! You just wrote and ran your first Python program! 🎉

Understanding Your Code

Let's understand what just happened:

- `print` is a Python command (or function) that displays text
- The parentheses `()` tell Python what to print
- The quotation marks `""` show the beginning and end of the text
- The exclamation mark `!` is just part of our message

Saving Your Program

Let's save your work so you can show it to others or come back to it later.

1. Click on "File" in the menu
2. Select "Save" (or "Save as...")
3. Choose where you want to save the file
4. Name your file `hello_world.py` (the `.py` part tells the computer it's a Python file)
5. Click "Save"

Now your program is saved on your computer!

Making It More Interesting

Let's make our program a bit more exciting:

```
print("Hello, World!")
print("My name is...")  # Put your name here
print("I'm learning to code with Python!")
```

Run this program to see what happens. You should see three lines of text now!

Adding Comments

In programming, we can add notes to ourselves that the computer ignores. These are called "comments" and they start with the # symbol:

```
# This is my first Python program!
print("Hello, World!")  # This displays a greeting
# The computer ignores this line
```

Comments help you remember what your code does and can help others understand your code too.

Making Your Program Interactive

Let's make the program ask for the user's name:

```
# My interactive Hello World program
name = input("What is your name? ")
print("Hello, " + name + "!")
print("Welcome to the world of Python programming!")
```

Run this program and see what happens. The program will wait for you to type your name and press Enter!

Explaining the Interactive Code

- `input()` is a function that asks the user to type something
- `name =` stores what the user types in a variable called "name"
- The `+` signs combine (concatenate) different pieces of text together

Common Mistakes to Watch For

- Missing quotation marks: `print(Hello, World!)`
- Missing parentheses: `print "Hello, World!"`
- Spelling errors: `prnit("Hello, World!")`

If your program doesn't work, check these common errors first!

Wrap-up Challenge

Try to modify your program to:

1. Ask for the user's age
2. Make the program tell the user how old they will be next year
3. Add a comment explaining how your code works

Hint: You'll need to use `input()` again and convert the age from text to a number with `int()`.

Extra Fun

Try changing the message to make the computer say something funny or draw a simple picture using text characters!

Remember, programming is about experimenting and having fun. Don't be afraid to try new things!