1. Create a Mojo Project with `magic`

We'll start by using the `magic` CLI to create a virtual environment and generate our initial project directory.

Step 1: Install the `magic` CLI

If you don't have the `magic` CLI yet, you can install it on macOS and Ubuntu Linux with this command:

```
bash
CopyEdit
curl -ssL https://magic.modular.com/deb16ff5-c2d2-4314-8120-0
033dd0446f3 | bash
```

Then run the `source` command that's printed in your terminal.

Step 2: Create the Project Directory

Navigate to the directory in which you want to create the project and execute:

```
bash
CopyEdit
magic init life --format mojoproject
```

This creates a project directory named `life`.

Step 3: Check the Project Contents

Let's go into the directory and list its contents:

```
bash
CopyEdit
cd life
```

```
ls -A
```

You should see the following files in the project directory:

- `mojoproject.toml`: Defines the project dependencies and other features.
- `magic.lock`: A lock file specifying transitive dependencies and the actual package versions.
- `.magic`: A subdirectory containing the conda virtual environment for the project.
- `.gitignore` and `.gitattributes`: Optionally used if you plan to use Git version control with the project.

The `magic` command automatically adds the `max` package as a dependency, which includes Mojo.

Step 4: Verify Mojo Installation

To verify that Mojo is correctly installed in the project's virtual environment, execute:

```
bash
CopyEdit
magic run mojo --version
```

You should see a version string indicating the latest released version of Mojo.

---

2. Create a "Hello, World" Program

In the project directory, create a file named `life.mojo` and add the following code:

```
mojo
CopyEdit
# My first Mojo program!
def main():
```

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

Explanation:

- `def main():`: Defines the `main()` function, which is the entry point for the program.
- **Indentation**: Mojo uses indentation (like Python), and it's common to use 4 spaces.
- `print()`: Mojo has a built-in `print()` function, so there's no need to import anything.

Step 1: Start the Virtual Environment

Start a shell session in the virtual environment:

```
bash
CopyEdit
magic shell
```

To exit the virtual environment later, just type `exit`.

Step 2: Run the Program

To run your program, use:

```
bash
CopyEdit
mojo life.mojo
```

You should see the output:

```
CopyEdit
Hello, World!
```

Step 3: Build the Executable

Mojo is a compiled language, so you can also compile the program into an executable file. Run:

```
bash
CopyEdit
mojo build life.mojo
```

By default, this saves an executable file named `life` in the current directory.

To execute the compiled program:

```
bash
CopyEdit
./life
```

You should see:

```
CopyEdit
Hello, World!
```