Part A: Environment Setup

This chapter gets your machine ready to run every example in the book. Work through it once, top to bottom, and you will not need to revisit it.

You will need:

- An Anthropic API key (get one at console.anthropic.com under API Keys)
- Python 3.11
- Node.js 18 or higher

Why Do We Even Need to Do This?

Think of your computer like a kitchen. It has basic equipment — a stove, fridge, a few pots — fine for everyday meals. But a professional Italian dinner needs the right ingredients, the right tools, and a dedicated workspace so nothing gets mixed up. Setting up a development environment is exactly that: a clean, dedicated workspace for this project, with only what it needs.

Here is what each piece does and why we need it:

- **Python 3.11** is the programming language used throughout this book. Your computer may already have a version of Python installed, but it might be the wrong version. MCP requires Python 3.10 or higher. We install a specific version so that everything behaves consistently.
- **uv** is a tool that manages Python versions and packages for you. Think of it as an organizer. Instead of manually hunting down the right Python version or worrying about packages clashing, uv handles all of that cleanly.
- **A virtual environment** is your isolated workspace. Imagine you are working on two different school projects that need completely different sets of supplies. You would not throw them all into one box. A virtual environment does the same thing for code: it keeps all the packages for this project separate from everything else on your machine. This prevents conflicts and keeps things predictable.
- **The packages** (`mcp`, `anthropic`, `pillow`, and others) are the actual libraries we will use to build MCP servers and clients. These get installed inside your virtual environment, not globally on your machine.
- **Node.js** is a separate runtime that some MCP tooling depends on. You will not write Node.js code in this book, but certain MCP tools run on it under the hood, so it needs to be present.
- **The API key** is your credential for accessing Claude through the Anthropic API. Every request you make to the model is authenticated with this key. Without it, none of the code in this book will run.

The whole setup takes around ten minutes. Do it once, and you will not need to think about it again.

---

macOS Setup

> The link below provides an easy way to install uv, however we will also be covering the uv installation guide in the macOS portion below: https://docs.astral.sh/uv/getting-started/installation/

Step 1 — Install uv

```
curl -LsSf https://astral.sh/uv/install.sh | sh
```

> Restart your terminal after installation.

Step 2 — Verify the installation

```
uv --version
```

Step 3 — Install Python 3.11

```
uv python install 3.11
```

Step 4 — Create a virtual environment

```
uv venv
```

Or, to name it explicitly:

```
uv venv .venv
```

Step 5 — Activate the environment

```
source .venv/bin/activate
```

Step 6 — Install required packages

```
uv pip install "mcp[cli]" ipykernel pillow anthropic uvicorn
```

Step 7 — Verify MCP is installed

```
python -c "import mcp; print(mcp.__file__)"
```

You should see a file path printed. If you see an error, repeat Step 6.

Step 8 — Install Node.js (LTS)

Using Homebrew:

```
brew install node@18
```

Or download directly from nodejs.org. Minimum version is Node 18.

Step 9 — Verify Node.js

```
node -v
npm -v
```

Windows Setup

Step 1 — Install uv

Open PowerShell (not CMD) and run:

```
powershell -ExecutionPolicy ByPass -c "irm https://astral.sh/uv/install.ps1
| iex"
```

Restart your terminal after installation.

Step 2 — Verify the installation

```
uv --version
```

Step 3 — Install Python 3.11

```
uv python install 3.11
```

Step 4 — Create a virtual environment

```
uv venv
```

Or, to name it explicitly:

```
uv venv .venv
```

Step 5 — Activate the environment

```
.venv\Scripts\activate
```

> If you get an execution policy error, first run:
> `Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope CurrentUser`

Step 6 — Install required packages

```
uv pip install "mcp[cli]" ipykernel pillow anthropic uvicorn
```

Step 7 — Verify MCP is installed

```
python -c "import mcp; print(mcp.__file__)"
```

You should see a file path printed. If you see an error, repeat Step 6.

Step 8 — Install Node.js (LTS)

Download directly from nodejs.org and run the installer. Minimum version is Node 18.

Or, if you have winget (built into Windows 11):

```
winget install OpenJS.NodeJS.LTS
```

Step 9 — Verify Node.js

```
node -v
npm -v
```

Setting Your API Key

Before running any code in this book, you need to make your Anthropic API key available as an environment variable.

Windows (PowerShell):

```
$env:ANTHROPIC_API_KEY = "your-key-here"
```

macOS:

```
export ANTHROPIC_API_KEY="your-key-here"
```

> Both of the above set the key for the current terminal session only. When you close the terminal, it is gone. This is fine for getting started.

Permanent API Key Setup (Recommended)

For a more permanent solution, store your key in a `.env` file at the root of your project:

```
ANTHROPIC_API_KEY=your-key-here
```

Then install the `python-dotenv` package:

```
uv pip install python-dotenv
```

And load it at the top of any script:

```
from dotenv import load_dotenv

load_dotenv()
```
