

The Wasmer JavaScript SDK

The `@wasmer/sdk` package is Wasmer's SDK for JavaScript.

Some functionality supported by `@wasmer/sdk` :

- Executing WebAssembly modules compiled to [WASIX](#)
- Includes **full** WASI support (filesystem access, environment variables, command-line arguments, etc.)
- Running packages from [the Wasmer registry](#)
- Mounting directories into a WASIX instance and communicating with it while the instance is running
- Running multi-threaded WASIX programs
- Spawning sub-processes using commands from the same package, side-loaded packages (`uses`), or arbitrary packages from the Wasmer registry using the virtual `wasmer run` command

Quickstart

First, add the [@wasmer/sdk](#) to your project.

[npm](#) yarn pnpm

```
npm install -S @wasmer/sdk
```

We'll be running version `3.12` of the `python/python`

```
index.ts

1  import { init, Wasmer } from "@wasmer/sdk";
2
3  await init();
4
5  const pkg = await Wasmer.fromRegistry("python/python@3.12");
6  const instance = await pkg.entrypoint.run({
7    args: ["-c", "print('Hello, World!')"],
8  });
9
10 const { code, stdout } = await instance.wait();
11 console.log(`Python exited with ${code}: ${stdout}`);
```

Breaking this down line-by-line...

Line 1 - imports the `Wasmer` class and `init()` function from the `@wasmer/sdk` package.

Line 3 - here, the `init()` function is called and awaited. This function must be called once before using any functionality from `@wasmer/sdk` It sets up the necessary environment for running WebAssembly modules and instantiates the WebAssembly runtime.

Line 5 - uses `Wasmer.fromRegistry()` to fetch the `python/python@3.12` package from the Wasmer registry and load it into memory, alongside any dependencies it may have.

Line 6 - runs the `python/python@3.12` package's entrypoint (the `python` command) with the arguments `-c "print('Hello, World!')"` and gets an `Instance` which can be used to communicate with the running process.

Line 10 - waits until the python interpreter has exited, then extracts the `code` and `stdout` values from the output so they can be printed to the console on line 11.

What Next?

API Docs



wasmerio/wasmer-js

Troubleshooting Common Problems

Last updated on June 10, 2024

< Support

Building a Web-Based Markdown Editor
with WASIX and WebAssembly >

On This Page

Quickstart

What Next?

Question? Give us feedback →

Edit this page on GitHub →