

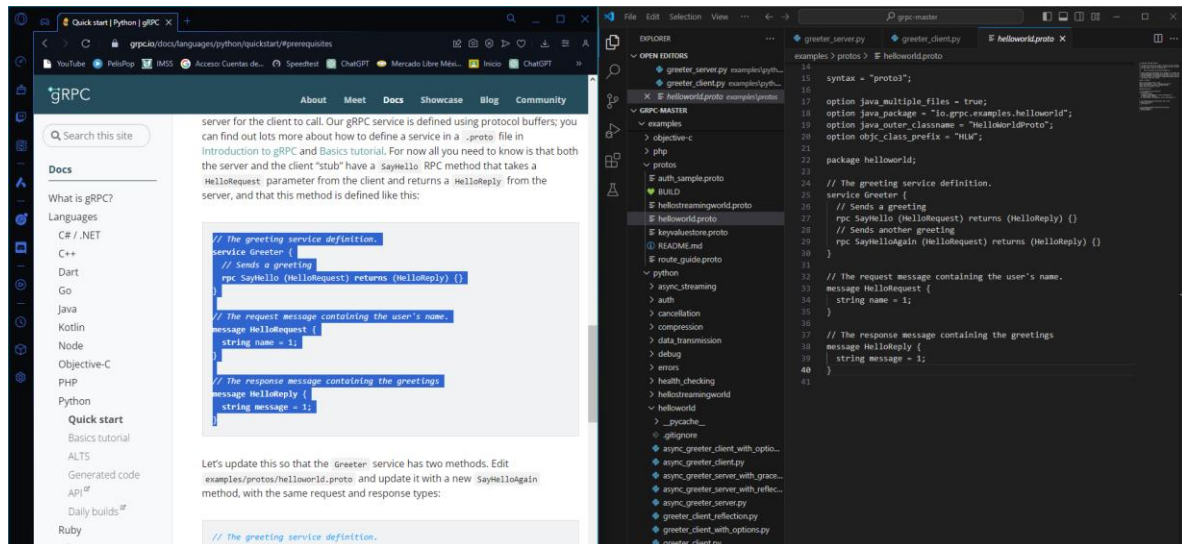
```

Microsoft Windows [Versión 10.0.22631.3155]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\victo>python -m pip show grpcio
Name: grpcio
Version: 1.62.0
Summary: HTTP/2-based RPC framework
Home-page: https://grpc.io
Author: The gRPC Authors
Author-email: grpc-io@googlegroups.com
License: Apache License 2.0
Location: C:\Users\victo\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.12_qbz5n2kfra8p0\LocalCache\local-pack
ages\Python312\site-packages
Requires:
Required-by: grpcio-tools

C:\Users\victo>python -m pip show grpcio-tools
Name: grpcio-tools
Version: 1.62.0
Summary: Protobuf code generator for gRPC
Home-page: https://grpc.io
Author: The gRPC Authors
Author-email: grpc-io@googlegroups.com
License: Apache License 2.0
Location: C:\Users\victo\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.12_qbz5n2kfra8p0\LocalCache\local-pack
ages\Python312\site-packages
Requires: grpcio, protobuf, setuptools
Required-by:

```



Remember to save the file!

Generate gRPC code

Next we need to update the gRPC code used by our application to use the new service definition.

From the `examples/python/helloworld` directory, run:

```

$ python -m grpc_tools.protoc -I../protos --python_out=. --pyi_out=. --grpc_out=.

```

```

.
├── .gitignore
├── async_greeter_client_with_options.py
├── async_greeter_client.py
├── async_greeter_server_with_graceful_shutdown.py
├── async_greeter_server_with_reflection.py
├── async_greeter_server.py
├── greeter_client_reflection.py
├── greeter_client_with_options.py
├── greeter_client.py
├── greeter_server_with_reflection.py
├── greeter_server.py
├── helloworld_pb2_grpc.py
├── helloworld_pb2.py
└── helloworld_pb2.pyi

```

