# **Add a producer for Kinesis Video Streams**

Write and examine

In this section of the [Java Producer Library procedure](https://docs.aws.amazon.com/kinesisvideostreams/latest/dg/producer-sdk-javaapi.html), you write and examine the Java example code you downloaded in the previous section.

The Java test application (DemoAppMain) shows the following coding pattern:

* Create an instance of KinesisVideoClient.
* Create an instance of MediaSource.
* Register the MediaSource with the client.
* Start streaming. That is, start the MediaSource, and it starts sending data to the client.

The following sections provide details.

**Creating an Instance of KinesisVideoClient**

You create the KinesisVideoClient object by calling the createKinesisVideoClient operation.

final KinesisVideoClient kinesisVideoClient = KinesisVideoJavaClientFactory

.createKinesisVideoClient(

Regions.US\_WEST\_2,

AuthHelper.getSystemPropertiesCredentialsProvider());

For KinesisVideoClient to make network calls, it needs credentials to authenticate. You pass in an instance ofSystemPropertiesCredentialsProvider, which reads AWSCredentials for the default profile in the credentials file:

[default]

aws\_access\_key\_id = ABCDEFGHIJKLMOPQRSTU

aws\_secret\_access\_key = AbCd1234EfGh5678IjKl9012MnOp3456QrSt7890

**Creating an Instance of MediaSource**

To send bytes to your Kinesis video stream, you need to produce the data. Amazon Kinesis Video Streams provides the MediaSource interface, which represents the data source.

For example, the Kinesis Video Streams Java library provides the ImageFileMediaSource implementation of the MediaSource interface. This class only reads data from a series of media files rather than a Kinesis video stream, but you can use it for testing the code.

final MediaSource bytesMediaSource = createImageFileMediaSource();

**Registering the MediaSource with the Client**

Register the media source that you created with the KinesisVideoClient so that it knows about the client (and can then send data to the client).

kinesisVideoClient.registerMediaSource(STREAM\_NAME, bytesMediaSource);

**Starting the Media Source**

Start the media source so that it can begin generating data and sending it to the client.

bytesMediaSource.start();