We'll take a closer look at the format of entry IDs (`1526985054069-0` and `1526985054079-0` in the example above) in the upcoming stages.

The XADD command

The XADD command appends an entry to a stream. If a stream doesn't exist already, it is created.

Here's how it works:

```
$ redis-cli XADD stream_key 1526919030474-0 temperature 36 humidity 95
"1526919030474-0" # (ID of the entry created)
```

The return value is the ID of the entry created, encoded as a bulk string.

`XADD` supports other optional arguments, but we won't deal with them in this challenge.

`XADD` also supports auto-generating entry IDs. We'll add support for that in later stages. For now, we'll only deal with explicit IDs (like `1526919030474-0` in the example above).

Tests

The tester will execute your program like this:

```
$ ./spawn_redis_server.sh
```

It'll then send an `XADD` command to your server and expect the ID as a response.

```
$ redis-cli XADD stream_key 0-1 foo bar
"0-1"
```

Your server should respond with `$3\r\n0-1\r\n` , which is `0-1` encoded as a