

# Pruebas de escritura

Puede usar el `zeebe-test` módulo para escribir pruebas JUnit para su trabajador de trabajo y flujo de trabajo BPMN. Proporciona una regla JUnit para arrancar el intermediario y algunas afirmaciones básicas.

## Uso en un proyecto Maven

Agregue `zeebe-test` como dependencia de prueba Maven a su proyecto:

```
<dependency>
  <groupId>io.zeebe</groupId>
  <artifactId>zeebe-test</artifactId>
  <scope>test</scope>
</dependency>
```



## Bootstrap the Broker

Use el `ZeebeTestRule` en su caso de prueba para iniciar un corredor incrustado. Contiene un cliente que se puede usar para implementar un flujo de trabajo BPMN o crear una instancia.



```
import io.zeebe.client.ZeebeClient;
import io.zeebe.client.api.response.WorkflowInstanceEvent;
import org.junit.Before;
import org.junit.Rule;
import org.junit.Test;

public class MyTest {

    @Rule public final ZeebeTestRule testRule = new ZeebeTestRule();

    private ZeebeClient client;


    @Test
    public void test() {
        client = testRule.getClient();

        client
            .newDeployCommand()
            .addResourceFromClasspath("process.bpmn")
            .send()
            .join();

        final WorkflowInstanceEvent workflowInstance =
            client
                .newCreateInstanceCommand()
                .bpmnProcessId("process")
                .latestVersion()
                .send()
                .join();
    }
}
```

## Verificar el resultado

El `ZeebeTestRule` también proporciona algunas afirmaciones básicas en el estilo AssertJ. El punto de entrada de las afirmaciones es `ZeebeTestRule.assertThat(...)`.



```
final WorkflowInstanceEvent workflowInstance = ...

ZeebeTestRule.assertThat(workflowInstance)
    .isEnded()
    .hasPassed("start", "task", "end")
    .hasVariable("result", 21.0);
```