Crear instancias de flujo de trabajo sin bloqueo

Prerrequisitos

1. Ejecución de Zeebe Broker con punto final `localhost:26500` (predeterminado)
2. Ejecute el ejemplo Implementar un flujo de trabajo

NonBlockingWorkflowInstanceCreator.java

Fuente en github

```
/*
 * Copyright Camunda Services GmbH and/or licensed to Camunda Services GmbH
under
 * one or more contributor license agreements. See the NOTICE file distributed
 * with this work for additional information regarding copyright ownership.
 * Licensed under the Zeebe Community License 1.0. You may not use this file
 * except in compliance with the Zeebe Community License 1.0.
 */
package io.zeebe.example.workflow;

import io.zeebe.client.ZeebeClient;
import io.zeebe.client.ZeebeClientBuilder;
import io.zeebe.client.api.ZeebeFuture;
import io.zeebe.client.api.response.WorkflowInstanceEvent;

public class NonBlockingWorkflowInstanceCreator {
  public static void main(final String[] args) {
    final String broker = "127.0.0.1:26500";
    final int numberOfInstances = 100_000;
    final String bpmnProcessId = "demoProcess";

    final ZeebeClientBuilder builder =

ZeebeClient.newClientBuilder().brokerContactPoint(broker).usePlaintext();

    try (ZeebeClient client = builder.build()) {
      System.out.println("Creating " + numberOfInstances + " workflow
instances");

      final long startTime = System.currentTimeMillis();

      long instancesCreating = 0;

      while (instancesCreating < numberOfInstances) {
        // this is non-blocking/async => returns a future
        final ZeebeFuture<WorkflowInstanceEvent> future =

client.newCreateInstanceCommand().bpmnProcessId(bpmnProcessId).latestVersion().s
end();

        // could put the future somewhere and eventually wait for its completion

        instancesCreating++;
      }

      // creating one more instance; joining on this future ensures
      // that all the other create commands were handled

client.newCreateInstanceCommand().bpmnProcessId(bpmnProcessId).latestVersion().s
end().join();

      System.out.println("Took: " + (System.currentTimeMillis() - startTime));
    }
  }
}
```