

<b>Método sobre el que se realiza la prueba</b>	<b>Nombre significativo del test propuesto</b>	<b>Breve descripción del objetivo del test diseñado</b>	<b>Configuración y parámetros de entrada al test</b>	<b>Estado/valor esperado después de la ejecución</b>
pujar(Usuario usuario, double cantidad)	testPujaNoAbierta1	Pujar sin tener la subasta abierta.	Subasta.abierta == false.	IllegalStateException
pujar(Usuario usuario, double cantidad)	testUsuarioEsPropietarioPuja1	Pujar siendo el propietario de la subasta.	Subasta.propietario == usuario.	False
pujar(Usuario usuario, double cantidad)	testCreditoMenorCantidad	Pujar una cantidad teniendo un crédito inferior.	Usuario.credito < cantidad.	False
pujar(Usuario usuario, double cantidad)	testCantidadMenorQueUltimaPuja	Pujar una cantidad inferior a la puja más alta.	Cantidad < getPuyaMayor	False
pujar(Usuario usuario, double cantidad)	testPujaCorrecta1	Pujar con los requisitos.	Subasta.propietario != usuario y usuario.credito > cantidad	True
pujar(Usuario usuario)	testPujaNoAbierta2	Pujar sin tener la subasta abierta.	Subasta.abierta == false.	IllegalStateException
pujar(Usuario usuario)	testUsuarioEsPropietarioPuja2	Pujar siendo el propietario de la subasta.	Subasta.propietario == usuario.	False
pujar(Usuario usuario)	testPujaCorrecta2	Pujar con los requisitos.	Subasta.propietario != usuario y usuario.credito > cantidad	True
ejecutar()	testEjecutarPujaNoAbierta	Ejecutar una subasta que no está abierta.	Subasta.abierta == false.	IllegalStateException
ejecutar()	testEjecutarSinPuja	Ejecutar una subasta donde no se ha realizado ninguna puja.	Subasta.getPuyaMayor == null	False

ejecutar()	testEjecutarCorrectamente	Ejecutar la subasta con los requisitos.	Subasta.getPujaMayor != null y subasta.abierta == true	True
	testCuatroPujasCantidadEjecutar	Se crea la subasta y se hacen 4 pujas estableciendo la cantidad, seguidamente se ejecuta.	Subasta.propietario != usuarios y usuarios.credito > cantidad	Cerrado
	testDosPujasDiferentesEjecutar	Se crea la subasta, se realiza una puja estableciendo la cantidad y luego una puja sin la cantidad, seguidamente se ejecuta.	Subasta.propietario != usuarios y usuario.credito > cantidad	Cerrado
	testCuatroPujasSinCantidadEjecutar	Se crea la subasta, se realiza cuatro pujas sin establecer la cantidad, seguidamente se ejecuta.	Subasta.propietario != usuarios y usuario.credito => 1	Cerrado
	testDosPujasDiferentesSinEjecutar	Se crea la subasta, se realiza una puja estableciendo la cantidad y luego una puja sin la cantidad y no se ejecuta	Subasta.propietario != usuarios y usuario.credito > cantidad	Abierto