

# Pruebas de Cobertura

Proyecto GestionAlmacen

Entornos de Desarrollo

Andrés Serrano

# Repositorio:

[AndresFp2024/GestionAlmacen.CodigoJava: Código java del proyecto Gestión de Almacén](#)

## Plan de pruebas

- Objetivo: verificar que los métodos principales funcionan según lo esperado.
- Ámbito: clases del modelo (productos, pedidos, usuarios, almacén).
- Técnicas: caja blanca (pruebas unitarias).
- Criterios de aceptación: todas las pruebas deben pasar sin errores.

## Herramientas de depuración y prueba:

En el entorno Visual Studio Code se han utilizado:

- Java Test Runner → para ejecutar pruebas JUnit
- JUnit JAR Downloader → para importar JUnit 5 rápidamente
- Debugger Java → para análisis paso a paso
- Consola integrada → para verificar salidas y trazas de errores

## Pruebas automáticas implementadas

- Se utilizan @Test de JUnit 5
- Se ejecutan con Java Test Runner
- Verificación inmediata del estado de cada test
- Integración con VS Code permite ejecutar todos los tests de forma simultánea

## Pruebas unitarias realizadas

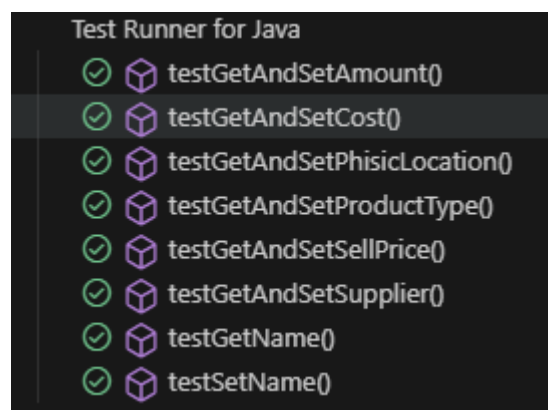
Se han implementado pruebas automáticas para:

Clase	Test implementado
Product	ProductTest
PhisicLocation	PhisicLocationTest
Order	OrderTest
Delivery	DeliveryTest
Admin	AdminTest
Client	ClientTest
ExternalAgent	ExternalAgentTest
Supplier	SupplierTest

WareHouse	WareHouseTest
-----------	---------------

## ProductTest.java

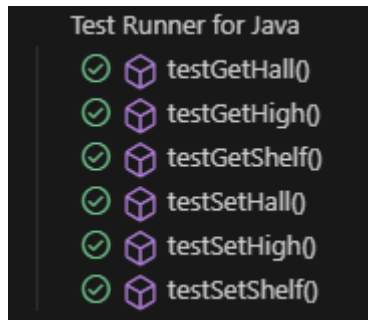
Método de prueba	Valor de entrada	Valor esperado
testGetName		assertEquals("Teclado", product.getName())
testSetName	product.setName("Ratón")	assertEquals("Ratón", product.getName())
testGetAndSetPhisicLocation	product.setPhisicLocation(...)	assertEquals(nuevaUbicacion, product.getPhisicLocation())
testGetAndSetProductType	product.setProductType("Periférico")	assertEquals("Periférico", product.getProductType())
testGetAndSetSellPrice	product.setSellPrice("30")	assertEquals("30", product.getSellPrice())
testGetAndSetCost	product.setCost("18")	assertEquals("18", product.getCost())
testGetAndSetSupplier	product.setSupplier("ProveedorX")	assertEquals("ProveedorX", product.getSupplier())
testGetAndSetAmount	product.setAmount("200")	assertEquals("200", product.getAmount())



## PhisicLocationTest.java

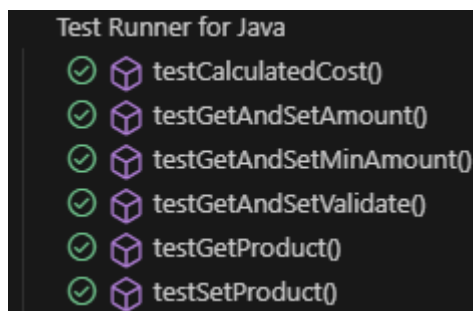
Método de prueba	Valor de entrada	Valor esperado
testGetShelf		assertEquals("Estante 1", ubicacion.getShelf())
testSetShelf	ubicacion.setShelf("Estante 2")	assertEquals("Estante 2", ubicacion.getShelf())
testGetHall		assertEquals("Pasillo A", ubicacion.getHall())

testSetHall	ubicacion.setHall("Pasillo B")	assertEquals("Pasillo B", ubicacion.getHall())
testGetHigh		assertEquals("Alta", ubicacion.getHigh())
testSetHigh	ubicacion.setHigh("Media")	assertEquals("Media", ubicacion.getHigh())



## OrderTest.java

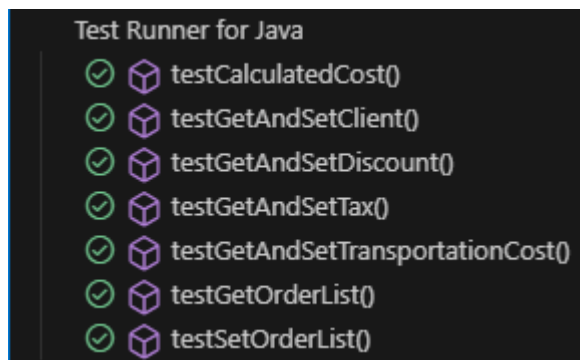
Método de prueba	Valor de entrada	Valor esperado
testGetProduct		assertEquals(product, order.getProduct())
testSetProduct	order.setProduct(nuevo)	assertEquals(nuevo, order.getProduct())
testGetAndSetAmount	order.setAmount(5)	assertEquals(5, order.getAmount())
testGetAndSetMinAmount	order.setMinAmount(2)	assertEquals(2, order.getMinAmount())
testGetAndSetValidate	order.setValidate(false)	assertFalse(order.getValidate())
testCalculatedCost		assertEquals(300.0, order.CalculatedCost())



## DeliveryTest.java

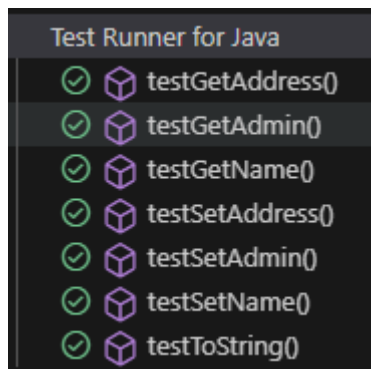
Método de prueba	Valor de entrada	Valor esperado
testGetOrderList		assertEquals(2, delivery.getOrderList().size())

testSetOrderList	delivery.setOrderList(nuevaLista)	assertEquals(0, delivery.getOrderList().size())
testGetAndSetTransportationCost	delivery.setTransportationCost(30)	assertEquals(30.0, delivery.getTransportationCost())
testGetAndSetTax	delivery.setTax(15.0)	assertEquals(15.0, delivery.getTax())
testGetAndSetDiscount	delivery.setDiscount(5.0)	assertEquals(5.0, delivery.getDiscount())
testGetAndSetClient	delivery.setClient("Cliente B")	assertEquals("Cliente B", delivery.getClient())
testCalculatedCost		assertEquals(565.0, delivery.CalculatedCost())



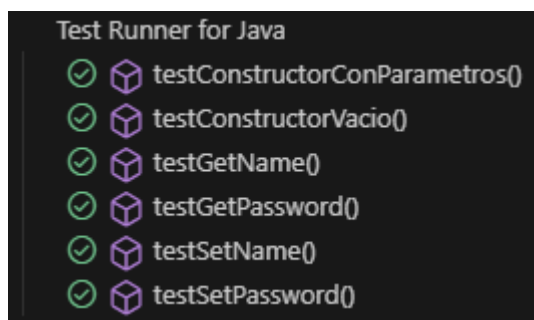
## WarehouseTest.java

Método de prueba	Valor de entrada	Valor esperado
testGetName		assertEquals("Central", almacen.getName())
testSetName	almacen.setName("Almacén Norte")	assertEquals("Almacén Norte", almacen.getName())
testGetAddress		assertEquals("Calle Mayor 15", almacen.getAddress())
testSetAddress	almacen.setAddress("Avenida Principal 45")	assertEquals("Avenida Principal 45", almacen.getAddress())
testGetAdmin		assertEquals(admin, almacen.getAdmin())
testSetAdmin	almacen.setAdmin(nuevoAdmin)	assertEquals(nuevoAdmin, almacen.getAdmin())
testToString		assertEquals("Name : Central\nAddress : Calle Mayor 15\nAdmin : Carlos", almacen.toString())



## AdminTest.java

Método de prueba	Valor de entrada	Valor esperado
testConstructorConParametros		assertEquals("María", admin.getName())
testConstructorVacio		assertNull(adminVacio.getName())
testSetName	admin.setName("Luis")	assertEquals("Luis", admin.getName())
testSetPassword	admin.setPassword("nuevoPass")	assertEquals("nuevoPass", admin.getPassword())
testGetName		assertEquals("María", admin.getName())
testGetPassword		assertEquals("admin123", admin.getPassword())



## ClientTest.java

Método de prueba	Valor de entrada	Valor esperado
testConstructor		assertEquals("Andrés", client.getName())
testSetName	client.setName("Cristina")	assertEquals("Cristina", client.getName())
testSetId	client.setId("CL999")	assertEquals("CL999", client.getId())
testGetName		assertEquals("Andrés", client.getName())
testGetId		assertEquals("CL001", client.getId())



## ExternalAgentTest.java

Método de prueba	Valor de entrada	Valor esperado
testGetName		assertEquals("Javier López", agent.getName())
testSetName	agent.setName("Laura Ruiz")	assertEquals("Laura Ruiz", agent.getName())
testGetPhoneNumber		assertEquals("987654321", agent.getPhoneNumber())
testSetPhoneNumber	agent.setPhoneNumber("123456789")	assertEquals("123456789", agent.getPhoneNumber())
testGetAddress		assertEquals("Calle Falsa 123", agent.getAddress())
testSetAddress	agent.setAddress("Av. Nueva 456")	assertEquals("Av. Nueva 456", agent.getAddress())
testGetId		assertEquals(1001, agent.getId())
testSetId	agent.setId(2002)	assertEquals(2002, agent.getId())
testGetEmail		assertEquals("javier@example.com", agent.getEmail())
testSetEmail	agent.setEmail("laura@example.com")	assertEquals("laura@example.com", agent.getEmail())
testGetFuntion		assertEquals("Distribuidor", agent.getFuntion())
testSetFuntion	agent.setFuntion("Proveedor")	assertEquals("Proveedor", agent.getFuntion())

```
Test Runner for Java
✓ testGetAddress()
✓ testGetEmail()
✓ testGetFuntion()
✓ testGetId()
✓ testGetName()
✓ testGetPhoneNumber()
✓ testSetAddress()
✓ testSetEmail()
✓ testSetFuntion()
✓ testSetId()
✓ testSetName()
✓ testSetPhoneNumber()
```

SupplierTest.java

Método de prueba	Valor de entrada	Valor esperado
testConstructor		assertEquals("ProveedorX", supplier.getName())
testSetName	supplier.setName("ProveedorY")	assertEquals("ProveedorY", supplier.getName())
testSetId	supplier.setId("SUP999")	assertEquals("SUP999", supplier.getId())
testGetName		assertEquals("ProveedorX", supplier.getName())
testGetId		assertEquals("SUP001", supplier.getId())

```
Test Runner for Java
✓ testConstructor()
✓ testGetId()
✓ testGetName()
✓ testSetId()
✓ testSetName()
```



## Incidencias detectadas y solucionadas

Incidencia	Solución aplicada
Error al convertir String a int	Añadido Integer.parseInt(...)
Métodos sin @Test	Anotación agregada para que sean reconocidos
Pruebas fallaban por valores incorrectos	Corregidos valores en los tests
Imports ausentes	Añadidas importaciones necesarias (import static ...)