

Constructor

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Lista de materiales	{lista != null} 1	{lista == null} 2
Descripción	{descripción != null} 3	{descripción == null} 4

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{lista != null, descripción != null}	nueva instancia de la clase	1,3		
Clases erróneas	{lista == null}	Lista de materiales nula	2		
	{descripción == null}	descripción nula	4		

setListaMat()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Lista de materiales	{listaMat != null} 1	{listaMat == null} 2

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{listaMat != null}	setea Lista Materiales	1		
Clases erróneas	{listaMat == null}	Lista de materiales nula	2		

Constructor

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Pedido	{pedido != null} 1	{pedido == null} 2

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{pedido != null}	nueva instancia de la clase	1		
Clases erróneas	{pedido == null}	El parametro es null	2		

agregarObservacion()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Observacion	{obs != null} 1	{obs == null} 2

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{obs != null}	observación agregada	1		
Clases erróneas	{obs == null}	Observacion nula	2		

aceptarPedido()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
fechaProduccion	{fechaProduccion != null} 1	{fechaProduccion == null} 2

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{fechaProduccion != null}	pedido aceptado	1		
Clases erróneas	{fechaProduccion == null}	Fecha propuesta por produccion nula	2		

Constructor

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Mensaje	{mensaje != null} 1	{mensaje == null} 2
Lista de materiales	{faltantes != null} 3	{faltantes == null} 4

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{mensaje != null, faltantes != null}	nueva instancia de la clase	1,3		
Clases erróneas	{mensaje == null}	null	2		
	{faltantes == null}	Lista nula	4		

Constructor

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Mensaje	{mensaje != null} 1	{mensaje == null} 2

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{mensaje != null}	nueva instancia de la clase	1		
Clases erróneas	{mensaje == null}	null	2		

agregarLote()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Lote	{nuevo != null} 1	{nuevo == null} 2

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{nuevo != null}	agrega Lote	1		
Clases erróneas	{nuevo == null}	Lote nulo	2		

borrarLote()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Lote	{lot != null} 1	{lot == null} 2

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{lot != null}	borra Lote	1		
Clases erróneas	{lot == null}	Lote nulo	2		

agregarNuevo()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Lote	{nuevo != null} 1	{nuevo == null} 2

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{nuevo != null}	agrega Material	1		
Clases erróneas	{nuevo == null}	Material nulo	2		

verificarExistencias()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Formato tipo	{tipo = TIPXXXXXX && 0 <= X <= 9} 1	{tipo != TIPXXXXXX} 2
Intervalo	{0 < cantidad <= 999} 3	{cantidad = 0} 4.1 {cantidad > 999} 4.2

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{TIP098212, 300}	verifica si los materiales son Suficientes para la construcción	1,3	{TIP999999}	verifica si los materiales son suficientes para la Construcción
				{TIP000000}	
				{1}	
				{999}	
Clases incorrectas	{0}	Cantidad invalida	4.1		
	{1005}	Cantidad invalida	4.2	{1000}	Cantidad invalida
	{TIP1122}	Tipo invalido	2	{TIP000000}	Tipo invalido
	{TIP100000000}	Tipo invalido	2	{TIP9999999}	Tipo invalido

getCodigo()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Codigo	{"Flipper" "Consola individual" "Consola grupal" "Simulador"}1	{!("Flipper" "Consola individual" "Consola grupal" "Simulador")}2

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{"Flipper"}	retorna codigo del producto	1		
Clases erróneas	{"Hola"}	Tipo producto invalido	2	{" "}	Tipo producto invalido

getProducto()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Formato codigo	{codigo = TIPXXXXXX && 0 <= X <= 9} 1	{codigo != TIPXXXXXX} 2

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{"TIP213431"}	retorna producto	1	{TIP999999}	retorna producto
				{TIP000000}	retorna producto
Clases erróneas	{TIP1122}	Codigo invalido	2	{TIP000000}	Codigo invalido
	{TIP100000000}	Codigo invalido	2	{TIP9999999}	Codigo invalido

actualizarExistencias()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
-----------	------------------	-----------------

Clase ListaMaterialesStock

Tipo Producto	{tipo != null} 1	{tipo == null} 2
---------------	------------------	------------------

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{tipo != null}	actualiza las existencias del producto	1		
Clases erróneas	{tipo == null}	Producto nulo	2		

Constructor

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Formato legajo	{legajo = LEGXXXXXX && 0 <= X <= 9} 1	{legajo != LEGXXXXXX} 2
Longitud nya	{0 < length nya <= 100}3	{length nya = 0} 4.1 {length nya > 100} 4.2
Sector	{"Ventas" "Produccion" "Contabilidad" "Inspeccion y calidad"} 5	{!("Ventas" "Produccion" "Contabilidad" "Inspeccion y calidad")}6

Batería de pruebas

	Entradas	Salidas	Clases Cubiertas	Valores Límite	Salidas
Clases correctas	{ LEG213431,"Rodrigo Cassanelli", "Ventas"}	nueva instancia de la clase	1,3,5	{LEG999999}	nueva instancia De la clase
				{LEG000000}	
				{nya.length = 1}	
				{nya.length = 100}	
Clases incorrectas	{LEG1122}	Legajo invalido	2	{LEG000000}	Legajo invalido
	{LEG100000000}	Legajo invalido	2	{LEG9999999}	
	{nya.length = 0}	Nombre y apellido invalidos	4.1		
	{nya.length = 200}	Nombre y apellido invalidos	4.2	{nya.length =101}	Nombre y Apellido invalidos
	{"Hola"}	Sector invalido	6	{" "}	Sector invalido

Constructor

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Formato numeroPedido	{numeroPedido = PEDXXXXXX && 0 <= X <= 9} 1	{numeroPedido != PEDXXXXXX} 2
Formato codigoMaquina	{codigoMaquina = TIPXXXXXX && 0 <= X <= 9} 3	{codigoMaquina != PEDXXXXXX} 4
Maquina	{"Flipper" "Consola individual" "Consola grupal" "Simulador"}5	{!("Flipper" "Consola individual" "Consola grupal" "Simulador")}6
Intervalo	{0 < cantidad <= 999}7	{cantidad = 0}8.1 {cantidad > 999}8.2
fechaPedido	{fechaPedido != null}9	{fechaPedido == null}10
fechaEntregaVentas	{fechaEntregaVentas != null}11	{fechaEntregaVentas == null}12

Batería de pruebas

	Entradas	Salidas	Clases Cubiertas	Valores Límite	Salidas
Clases correctas	{PED102111, fechaPedido != null, fechaEntregaVentas != null, "Flipper", TIP213431, 200}	nueva instancia de la clase	1,3,5,7,9,11	{PED999999}	nueva instancia De la clase
				{PED000000}	
				{TIP999999}	
				{TIP000000}	
				{1}	
				{999}	
Clases erróneas	{PED1110}	Numero pedido invalido	2	{PED000000}	Numero pedido Invalido
	{PED10000000}	Numero pedido invalido	2	{PED99999999}	
	{"Hola"}	Tipo de maquina invalido	6	{" "}	Tipo de maquina Invalido
	{0}	Cantidad invalida	8.1		
	{1005}	Cantidad invalida	8.1	{1000}	Cantidad invalida
	{TIP1122}	Codigo de maquina invalido	4	{TIP000000}	Codigo de Maquina invalido
	{TIP100000000}	Codigo de maquina invalido	4	{TIP99999999}	
	{fechaPedido == null}	Fecha de pedido invalido	10		
	{fechaEntregaVentas == null}	Fecha de entrega invalida	12		

agregarObservacion()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Observacion	{obs != null} 1	{obs == null} 2

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{obs != null}	observación agregada	1		
Clases erróneas	{obs == null}	Observacion nula	2		

aceptarPedido()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
fechaProduccion	{fechaProduccion != null} 1	{fechaProduccion == null} 2

Batería de pruebas

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{fechaProduccion != null}	pedido aceptado	1		
Clases erróneas	{fechaProduccion == null}	Fecha propuesta por produccion nula	2		

setEstadoActual()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
estadoActual	{estadoActual != null} 1	{estadoActual == null} 2

Batería de pruebas

Clase Pedido

	Entradas	Salidas	Clases cubiertas	Valores Límite	Salidas
Clases correctas	{estadoActual != null}	estado seteado	1		
Clases erróneas	{estadoActual == null}	Estado nulo	2		

Constructor

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Formato	{codigo = MATXXXXX && 0 <= X <= 9} 1	{codigo != MATXXXXX} 2
Longitud	{0 < length descripcion <= 100} 3	{length descripcion > 100} 4.1 {length descripcion = 0} 4.2
Intervalo	{000,0000 < cantidad <= 999,9999} 5	{cantidad <= 0} 6.1 {cantidad > 999,999} 6.2

Batería de pruebas

	Entradas	Salidas	Clases Cubiertas	Valores Límite	Salidas
Clases correctas	{MAT10211, descripcion.lenght = 50, 300}	nueva instancia de la clase	1,3,5	{MAT999999}	nueva instancia De la clase
				{MAT000000}	
				{descripcion.length = 100}	
				{descripcion.length = 1}	
				{999,9999}	
				{000,0001}	
Clases erróneas	{MAT101}	Codigo invalido	2	{MAT0000}	Codigo invalido
	{MAT1000000}	Codigo invalido	2	{MAT999999}	Codigo invalido
	{descripcion.length = 0}	Descripcion invalida	4.1		
	{descripcion.length = 200}	Descripcion invalida	4.2	{descripcion.length =101}	Descripcion invalida
	{000,0000}	Cantidad invalida	6.1		
	{1001,0000}	Cantidad invalida	6.2	{1000,0000}	Cantidad invalida

setCodigo()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Formato	{codigo = MATXXXXX && 0 <= X <= 9} 1	{codigo != MATXXXXX} 2

Batería de pruebas

Clase Material

	Entradas	Salidas	Clases Cubiertas	Valores Límite	Salidas
Clases correctas	{MAT10211}	seteaCodigo	1	{MAT999999}	seteaCodigo
				{MAT000000}	seteaCodigo
Clases erróneas	{MAT101}	Codigo invalido	2	{MAT00000}	Codigo invalido
	{MAT1000000}	Codigo invalido	2	{MAT999999}	Codigo invalido

setDescripcion()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Longitud	{0 < length descripcion <= 100} 1	{length descripcion > 100} 2.1 {length descripcion = 0} 2.2

Batería de pruebas

	Entradas	Salidas	Clases Cubiertas	Valores Límite	Salidas
Clases correctas	{descripcion.length = 50}	seteaDescripcion	1	{descripcion.length = 100}	nueva instancia De la clase
				{descripcion.length = 1}	
Clases erróneas	{descripcion.length = 0}	Descripcion invalida	2.1		
	{descripcion.length = 200}	Descripcion invalida	2.2	{descripcion.length = 101}	Descripcion invalida

setCantidad()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Intervalo	{000,0000 < cantidad <= 999,9999} 1	{cantidad <= 0} 2.1 {cantidad > 999,999} 2.2

Batería de pruebas

	Entradas	Salidas	Clases Cubiertas	Valores Límite	Salidas
Clases correctas	{300}	seteaCantidad	1	{999,9999}	nueva instancia De la clase
				{000,0001}	

Clase Material

Clases erróneas	{000,0000}	Cantidad invalida	6.1		
	{1001,0000}	Cantidad invalida	6.2	{1000,0000}	Cantidad invalida

Constructor

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Formato pedido	{pedido != null} 1	{pedido == null} 2
Formato numero de lote	{numeroLote = LOTXXXXXX && 0 <= X <= 9} 3	{numeroLote != LOTXXXXXX} 4

Batería de pruebas

	Entrada	Salidas	Clases Cubiertas	Valores limite	Salidas
Clases correctas	(pedido != null , LOT123456)	nueva instancia de la clase	1,3	(pedido != null, LOT000000) (pedido != null, LOT999999)	Nueva instancia de la clase
Clases erróneas	(pedido == null, LOT123456)	el pedido es null	2,3		
	(pedido != null, LOT001)	el numero de lote no cumple con El formato	1,4	(pedido != null, LOT000)	el numero de lote no cumple con el formato
	(pedido != null, LOT0000000)	el numero de lote no cumple con El formato	1,4	(pedido != null, LOT999999)	el numero de lote no cumple con el formato

Constructor

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Tema	{tema != null} 1	{tema == null} 2
Fecha de observacion	{fechaObservacion != null} 3	{fechaObservacion == null} 4
Formato legajo	{legajoEmpleado = LEGXXXXXX && 0 <= X <= 9} 5	{LegajoEmpleado != LEGXXXXXX} 6
Formato texto observación	{0 < length texto <= 500} 7	{length texto <= 0 } 8.1 {length texto > 500} 8.2

Batería de pruebas

	Entrada	Salidas	Clases Cubiertas	Valores limite	Salidas
Clases correctas	(tema != null , fechaObservacion != null, LEG123456, texto.length = 250)	nueva instancia de la clase	1,3,5,7	LEG000000 LEG999999 Texto.length = 1 Texto.length = 500	Nueva instancia De la clase
Clases erróneas	tema == null	el tema es null	2		
	FechaObservacion == null	la fecha de observacion es null	4		
	LEG123	el legajo no cumple con el formato	6	LEG000 LEG999	el legajo no cumple con El formato
	texto.length = 600	el texto no cumple con el formato	8.2		
	texto.length = 0	el texto no cumple con el formato	8.1		

Constructor

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Pedido	{pedido != null} 1	{pedido == null} 2

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
Clases correctas	{pedido != null}	nueva instancia de la clase	1		
Clases erróneas	{pedido == null}	el pedido es null	2		

agregarObservacion

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Observación	{obs != null} 1	{obs == null} 2

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
Clases correctas	{obs != null}	Imposible agregar, pedido ya aceptado	1		
Clases erróneas	{obs == null}	la observacion es null	2		

Constructor

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Pedido	{pedido != null} 1	{pedido == null} 2

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
Clases correctas	{pedido != null}	nueva instancia de la clase	1		
Clases erróneas	{pedido == null}	el pedido es null	2		

Constructor

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Texto	{texto != null} 1	{texto == null} 2
Mensaje	{mensaje != null} 3	{mensaje == null} 4

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
Clases correctas	{texto != null, mensaje != null}	nueva instancia de la clase	1,3		
Clases erróneas	texto == null	el texto es null	2		
	mensaje == null	el mensaje es null	4		

Buscar

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Legajo a buscar	{legajo = LEGXXXXXX && 0<= X < 9} 1	{legajo != LEGXXXXXX} 2

Batería de pruebas

	Entrada	Salidas	Clases Cubiertas	Valores limite	Salidas
Clases correctas	(LEG123456)	Referencia al empleado buscado si este existe, aviso de inexistencia en caso contrario	1	LEG000000 LEG999999	Referencia al empleado buscado si este existe, aviso de inexistencia En caso contrario
Clases erróneas	(LEG123) (LEG1234567)	Aviso de inexistencia del empleado buscado ya que no es un legajo valido	2		

agregarMaterial()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Material a agregar	{nuevo != null} 1	{nuevo == null} 2

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
Clases correctas	{nuevo != null}	Agregar el nuevo material a la lista	1		
Clases erróneas	{nuevo == null}	Error ya que es null	2		

getMaterial()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Codigo a buscar	{codigo = MATXXXXX && 0 <= X <= 9} 1	{codigo != MATXXXXX} 2

Batería de pruebas

	Entrada	Salidas	Clases Cubiertas	Valores limite	Salidas
Clases correctas	(MAT12345)	Devuelve una referencia la material indicado, o un aviso de inexistencia	1	MAT00000 MAT99999	Devuelve una referencia al material indicado, O un aviso de inexistencia
Clases erróneas	MAT123	El codigo del material no respeta el formato	2	MAT000	El codigo del material no respeta el formato
	MAT123345	El codigo del material no respeta el formato	2	MAT999999	El codigo del material no respeta el formato

modificarMaterial()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Codigo del material a modificar	{codigo = MATXXXXX && 0 <= X <= 9} 1	{codigo != MATXXXXX} 2
Nueva cantidad del material	{cantidad >= 0} 3.1 {cantidad ∈ R} 3.2	{cantidad < 0} 4.1 {cantidad ∉ R} 4.2

Batería de pruebas

	Entrada	Salidas	Clases Cubiertas	Valores limite	Salidas
Clases correctas	(MAT12345 , 300)	Cambios en la cantidad del material indicado por el codigo	1, 3.2	MAT00000 MAT99999 Cantidad = 0	Cambios en la cantidad del material indicado por el codigo
Clases erróneas	Cantidad = -1	Error en la cantidad de material	4.1		
	MAT123	El codigo del material no respeta el formato	2	MAT000	El codigo del material no respeta el formato
	MAT123345	El codigo del material no respeta el formato	2	MAT999999	El codigo del material no respeta el formato

borrarMaterial()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Codigo del material a borrar	{codigo = MATXXXXX && 0 <= X <= 9} 1	{codigo != MATXXXXX} 2

Batería de pruebas

	Entrada	Salidas	Clases Cubiertas	Valores limite	Salidas
--	---------	---------	------------------	----------------	---------

Clases correctas	(MAT12345)	Borrar el material indicado por el codigo si existe, si no avisa de la inexistencia	1	MAT00000 MAT99999	Borrar el material indicado por el codigo si existe, si no avisa de la inexistencia
Clases erróneas	MAT123	El codigo del material no respeta el formato	2	MAT000	El codigo del material no respeta el formato
	MAT123345	El codigo del material no respeta el formato	2	MAT999999	El codigo del material no respeta el formato

agregarMaterial()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Código del material a agregar	{codigo = MATXXXXX && 0 <= X <= 9} 1	{codigo != MATXXXXX} 2
Descripción del nuevo material	{descripcion != null} 3	{descripcion == null} 4
Cantidad del nuevo material	{cantidad >= 0} 5.1 {cantidad ∈ R} 5.2	{cantidad < 0} 6.1 {cantidad ∉ R} 6.2

Batería de pruebas

	Entrada	Salidas	Clases Cubiertas	Valores limite	Salidas
Clases correctas	(MAT12345, descripcion.length = 50, 45.2)	Agregado el nuevo material	1,3,5.1,5.2		
Clases erróneas	MAT123	El codigo del material no respeta el formato	2	MAT000	El codigo del material no respeta el formato
	MAT123345	El codigo del material no respeta el formato	2	MAT999999	El codigo del material no respeta el formato
	Descripcion == null	La descripcion es null	4		
	Cantidad = -3	La cantidad es negativa	6.1		

agregarNuevo()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Pedido a agregar	{nuevo != null} 1	{nuevo == null} 2

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
Clases correctas	{nuevo != null}	Agregar el nuevo pedido a la lista	1		
Clases erróneas	{nuevo == null}	Error ya que es null	2		

borrarPedido()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Pedido a eliminar	{ped != null} 1	{ped = null} 2

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
Clases correctas	(ped != null)	Borrar el pedido indicado por el codigo si existe, si no avisa de la inexistencia	1		
Clases erróneas	(ped == null)	Error ya que el pedido es null	2		

setEmpleadoActual()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Referencia al empleado actual	{actual != null} 1	{actual == null} 2

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
Clases correctas	(actual != null)	Setea el empleado actual	1		
Clases erróneas	(actual == null)	El empleado actual es null	2		

setPedidoActual()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Referencia al pedido actual	{pedido != null} 1	{pedido == null} 2

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
Clases correctas	(pedido != null)	Setea el pedido actual	1		
Clases erróneas	(pedido == null)	El pedido actual es null	2		

setLoteActual()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Referencia al lote actual	{lote != null} 1	{lote == null} 2

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
Clases correctas	(lote != null)	Setea el lote actual	1		
Clases erróneas	(lote == null)	El lote actual es null	2		

setProductoActual()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Referencia al producto actual	{producto != null} 1	{producto == null} 2

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
Clases correctas	(producto != null)	Setea el producto actual	1		
Clases erróneas	(producto == null)	El producto actual es null	2		

buscarEmpleado()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Legajo a buscar	{legajo = LEGXXXXXX && 0 <= X < 9} 1	{legajo != LEGXXXXXX} 2

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
--	---------	---------	------------------	----------------	---------

Clases correctas	(LEG123456)	Referencia al empleado buscado si este existe, aviso de inexistencia en caso contrario	1	LEG000000 LEG999999	Referencia al empleado buscado si este existe, aviso de inexistencia en caso contrario
Clases erróneas	(LEG123) (LEG1234567)	Aviso de inexistencia del empleado buscado ya que no es un legajo valido	2		

cambiarAceptado()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Fecha propuesta por produccion	{fechaProduccion != null} 1	{fechaProduccion == null} 2

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
Clases correctas	(fechaProduccion != null)	Acepta el pedido con la fecha propuesta por produccion	1		
Clases erróneas	(fechaProduccion == null)	Error	2		

crearNuevoPedido()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Fecha de pedido	{fechaProduccion != null} 1	{fechaProduccion == null} 2
Tipo de maquina	{tipoMaquina == TIPXXXXXX && 0<= X <= 9} 3	{tipoMaquina != TIPXXXXXX} 4
Cantidad a producir	{cantProducir >= 0} 5	{cantProducir < 0} 6

Fecha solicitada por ventas	{fechaVentas != null} 7	{fechaVentas == null} 8
-----------------------------	-------------------------	-------------------------

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
Clases correctas	(fechaProduccion != null, TIP123456, 200, FechaVentas != null)	Nuevo pedido creado	1,3,5,7	TIP000000 TIP999999	Nuevo pedido creado
Clases erróneas	fechaProduccion == null FechaVentas == null CantProducir = -1 TIP123 TIP1234567	Error	2 8 6 4 4		

crearObservacion()

Tabla de particiones

Condición	Clases correctas	Clases erróneas
Tema	{tema != null} 1	{tema == null} 2
Formato texto observación	{0 < length texto <= 500} 3	{length texto <= 0 } 4.1 {length texto > 500} 4.2

Batería de pruebas

	Entrada	Salidas	Clases cubiertas	Valores limite	Salidas
Clases correctas	(tema != null, texto.length = 50)	Nueva observación agregada	1,3	exto.length = xto.length = 5	Nueva observación agregada
Clases erróneas	Tema == null Texto.length = 0 Texto.length = 501	Informa del error y no agrega la observación	2 4.1 4.2		