```java
package edu.cpp.cs5800.VendingMachine;

import edu.cpp.cs5800.VendingMachine.snacks.Snack;
import edu.cpp.cs5800.VendingMachine.states.Idle;
import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.assertEquals;

public class IdleTest {
    VendingMachine vendingMachine = new VendingMachine();
    Idle idle = new Idle(vendingMachine);

    @Test
    public void testSelectSnack() {
        int snackId = 1;
        Snack snack = vendingMachine.getSnack(snackId);
        String expected = "You have selected: " + snackId + " (" + snack.getName() + ")";
        String actual = idle.selectSnack(snackId);
        assertEquals(expected, actual);
    }

    @Test
    public void testInsertAmount() {
        String expected = "Invalid request: Please first select a snack!";
        assertEquals(expected, idle.insertMoney(6));
    }

    @Test
    public void testDispenseSnack() {
        String expected = "Invalid request: Please first select a snack!";
        assertEquals(expected, idle.dispenseSnack());
    }
}

```

Part 1 (Vending Machine) Unit Test

```java
package edu.cpp.cs5800.VendingMachine;

import edu.cpp.cs5800.VendingMachine.snacks.Snack;
import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.assertEquals;

class SnackDummy extends Snack {

    public SnackDummy(int quantity, double price) {
        super("SnackDummy", quantity, price);
    }
}

public class SnackTest {
    Snack snack = new SnackDummy(3, 1.5);

    @Test
    public void TestGetName() {
        assertEquals("SnackDummy", snack.getName());
    }

    @Test
    public void TestGetQuantity() {
        assertEquals(3, snack.getQuantity());
    }

    @Test
    public void TestGetPrice() {
        assertEquals(1.5, snack.getPrice());
    }

    @Test
    public void TestDispense() {
        assertEquals("SnackDummy", snack.dispense());
        assertEquals(2, snack.getQuantity());
    }
}
```

```java
package edu.cpp.cs5800.VendingMachine;

import edu.cpp.cs5800.VendingMachine.states.Idle;
import edu.cpp.cs5800.VendingMachine.states.WaitingForMoney;
import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.assertEquals;

public class VendingMachineTest {
    VendingMachine vendingMachine = new VendingMachine();

    @Test
    public void testGetSnack() {
        assertEquals("Coke", vendingMachine.getSnack(1).getName());
    }

    @Test
    public void testGetAndSetState() {
        assertEquals((new Idle(vendingMachine)).toString(), vendingMachine.getState().
toString());
        vendingMachine.setState(new WaitingForMoney(vendingMachine));
        assertEquals((new WaitingForMoney(vendingMachine)).toString(),
                vendingMachine.getState().toString());
    }

    @Test
    public void testSetAndGetSelectedSnack() {
        vendingMachine.setSelectedSnack(2);
        assertEquals(2, vendingMachine.getSelectedSnack());
    }
}

```

Part 1 (Vending Machine) Unit Test

```java
package edu.cpp.cs5800.VendingMachine;

import edu.cpp.cs5800.VendingMachine.states.DispensingSnack;
import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.assertEquals;

public class DispensingSnackTest {
    VendingMachine vendingMachine = new VendingMachine();
    DispensingSnack dispensingSnack = new DispensingSnack(vendingMachine);

    @Test
    public void testSelectSnack() {
        String expected = "Invalid request: Currently dispensing snack!";
        assertEquals(expected, dispensingSnack.selectSnack(1));
    }

    @Test
    public void testInsertAmount() {
        String expected = "Invalid request: Currently dispensing snack!";
        assertEquals(expected, dispensingSnack.insertMoney(6));
    }

    @Test
    public void testDispenseSnack() {
        String expected = "Completed transaction!";
        assertEquals(expected, dispensingSnack.dispenseSnack());
    }
}

```

```java
package edu.cpp.cs5800.VendingMachine;

import edu.cpp.cs5800.VendingMachine.states.WaitingForMoney;
import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.assertEquals;

public class WaitingForMoneyTest {
    VendingMachine vendingMachine = new VendingMachine();
    WaitingForMoney wfm = new WaitingForMoney(vendingMachine);

    @Test
    public void testSelectSnack() {
        String expected = "Invalid request: Please inserted money!";
        assertEquals(expected, wfm.selectSnack(1));
    }

    @Test
    public void testInsertAmount() {
        vendingMachine.setSelectedSnack(1);
        vendingMachine.setState(wfm);
        String expected = "You have inserted: $" + 6.0;
        assertEquals(expected, wfm.insertMoney(6));
    }

    @Test
    public void testDispenseSnack() {
        String expected = "Invalid request: Please inserted money!";
        assertEquals(expected, wfm.dispenseSnack());
    }
}

```

```java
package edu.cpp.cs5800.VendingMachine;

import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.assertFalse;
import static org.junit.jupiter.api.Assertions.assertTrue;

public class SnackDispenseHandlerTest {
    VendingMachine vendingMachine = new VendingMachine();
    SnackDispenseHandler handler = vendingMachine.getSnackDispenseHandler();

    @Test
    public void testDispenseWithEnoughMoneyAndQty() {
        assertTrue(handler.dispense(6, 10));
    }

    @Test
    public void testDispenseWithNotEnoughQty() {
        assertTrue(handler.dispense(6, 10));
        assertFalse(handler.dispense(6, 10));
    }

    @Test
    public void testDispenseWithNotEnoughMoney() {
        assertFalse(handler.dispense(1, 1));
    }

    @Test
    public void testDispenseSnackNotFound() {
        assertFalse(handler.dispense(7, 1));
    }
}

```