```java
/*****************************************************************************
*
*  File: CoffeeOrders.java
*
*  Author: Joshua Wiley
*
*  Description:  Finds all calculations for a coffee shop
*
*  Date: 4-24-15
*
*  Comments: It is cheaper to send 1 medium box then 2 small boxes i.e. 8 bags
*            ordered would be shipped in 1 medium box not 2 small boxes.
*
******************************************************************************/

public class CoffeeOrders
{
    /*** Class Constants ***/

    private static final int LARGE_BOX  = 20;
    private static final int MEDIUM_BOX = 10;
    private static final int SMALL_BOX  = 5;

    public  static final double LARGE_BOX_PRICE  = 1.80;
    public  static final double MEDIUM_BOX_PRICE = 1.00;
    public  static final double SMALL_BOX_PRICE  = 0.60;
    private static final int   MAX_QUANTITY = 1500;

    private static final double PRICE_PER_BAG  = 5.50;

    /*** Class Methods ***/

    public static double calculatePurchasePrice( int bagsOrdered )
    {
        /*** local Variables ***/

        double purchasePrice = 0.0;

        /*** Calculates the purchase price before shipping ***/

        purchasePrice = PRICE_PER_BAG * bagsOrdered;

        return purchasePrice;
    }

    public static double totalPrice( int bagsOrdered, int large, int medium, int small )
    {
        /*** local Variables ***/

        double totalPrice = 0.0;

        /*** Calculates the total purchase price with shipping ***/

        totalPrice += calculatePurchasePrice( bagsOrdered );
        totalPrice += boxSizeCost( large, LARGE_BOX_PRICE );
        totalPrice += boxSizeCost( medium, MEDIUM_BOX_PRICE );
        totalPrice += boxSizeCost( small, SMALL_BOX_PRICE );

        return totalPrice;
    }

    public static int largeBoxesNeeded( int bagsToBox )
    {
        /*** local Variables ***/

        int boxes = 0;

        boxes = bagsToBox / LARGE_BOX;

        return boxes;
    }

    public static int mediumBoxesNeeded( int bagsToBox )
    {
        /*** local Variables ***/

        int boxes = 0;

        boxes = ( bagsToBox % LARGE_BOX ) / MEDIUM_BOX;

        if ( ( bagsToBox % LARGE_BOX ) % MEDIUM_BOX > SMALL_BOX )
            boxes ++;  //It is cheaper to send 1 medium then 2 small

        return boxes;
    }

    public static int smallBoxesNeeded( int bagsToBox )
    {
        /*** local Variables ***/

        int boxes = 0;

        boxes = ( ( ( bagsToBox % LARGE_BOX )  % MEDIUM_BOX ) );
        if ( boxes > SMALL_BOX )
            boxes = 0;  //It is cheaper to send 1 medium then 2 small
        else if ( boxes > 0 )
            boxes = 1;
        else
            boxes = 0;

        return boxes;
    }

    public static double boxSizeCost( int numberOfBoxes, double boxPrice )
    {
        /*** local Variables ***/

        double price = 0;

        /*** Calculates shipping ***/

        price = numberOfBoxes * boxPrice;

        return price;
    }

    public static boolean validateString( String newString )
    {
        boolean status = false;
                       //Makes sure the string is not empty
        if ( newString.trim().length() > 1 )
            status = true;

        return status;
    }

    public static boolean validateState( String newString )
    {
        boolean status = false;
        int i = 0;  //Initializes for loop

        String states[] = { "AL","AK","AZ","AR","CA",
                            "CO","CT","DE","FL","GA",
                            "HI","ID","IL","IN","IA",
                            "KS","KY","LA","MA","MI",
                            "ME","MD","MN","MS","MO",
                            "MT","NE","NV","NH","NJ",
                            "NM","NY","NC","ND","OH",
                            "OK","OR","PA","RI","SC",
                            "SD","TN","TX","UT","VT",
                              "VA","WA","WV","WI","WY" };

        for ( i = 0; i < states.length; i++ )
        {                  //Makes sure 2 letter abbr. is an actual state
            if ( newString.equals( states[ i ] ) )
                status = true;
        }

        return status;
    }

    public static boolean validateQuantity( int Quantity )
    {
        boolean status = false;

        if ( ( Quantity >= 1 )  && ( Quantity <= MAX_QUANTITY ) )
            status = true;

        return status;
    }

    public static boolean validateZipcode( int Zipcode )
    {
        boolean status = false;

        if ( ( Zipcode >= 01001 )  && ( Zipcode <= 99950 ) )
            status = true;

        return status;
    }
}
```