

Credit Name: CSE2140 2nd Language Programming  
Assignment Name: Lunch Order Mastery

How has your program changed from planning to coding to now? Please explain?

Created a public class with a private double variables called prices, fats, carbs, fibers and total.

```
public class LunchOrder {  
    private double prices, fats, carbs, fibers, total;  
}
```

Prepared decimal format to shorten any long decimals to 2 decimal places.

```
//Shortens number to 2 decimal places  
DecimalFormat df = new DecimalFormat("#0.00");
```

Created constructor method with default values for the class variables.

```
//Constructor method with default values  
public LunchOrder() {  
    prices = 0;  
    fats = 0;  
    carbs = 0;  
    fibers = 0;  
    total = 0;  
}
```

Created method that overloads constructor method with a 4 double variables as the parameters. Overloading total gets a separate method with a double as the parameter in order to overload it more efficiently.

```
//Overloading constructor method  
public LunchOrder(double price2, double fat2, double carb2, double fiber2) {  
    prices = price2;  
    fats = fat2;  
    carbs = carb2;  
    fibers = fiber2;  
}  
public LunchOrder(double totalAmt) {  
    total = totalAmt;  
}
```

Created an access method for each class variable, will return the variable.

```
//Access method  
public double getPrice() {  
    return prices;  
}  
public double getFat() {  
    return fats;  
}  
public double getCarb() {  
    return carbs;  
}  
public double getFiber() {  
    return fibers;  
}  
public double getTotal() {  
    return total;  
}
```

Created a modifier method for each class variable with a double as the parameter that can you change/alter the variable. Doesn't return a value, only initializes the variable.

```
//Modifier method  
public void setPrice(double price2) {  
    prices = price2;  
}  
public void setFat(double fat2) {  
    fats = fat2;  
}  
public void setCarb(double carb2) {  
    carbs = carb2;  
}  
public void setFiber(double fiber2) {  
    fibers = fiber2;  
}
```

In the client code in the main method, declare the variables as int as we are dealing with whole numbers.

```
public static void main(String[] args) {  
    // TODO Auto-generated method stub  
    //Declaration area  
    int burgerAmt, saladAmt, fryAmt, sodaAmt;
```

Prepared for user to input an answer.

```
//Prepare for user input  
Scanner userInput = new Scanner(System.in);
```

Created a new object for each menu item with their respective cost, fats, carbs and fiber, as well as a new object for the total.

```
//Create new objects  
LunchOrder totalAmt = new LunchOrder(0);  
LunchOrder burgerInfo = new LunchOrder(1.85, 9, 33, 1);  
LunchOrder saladInfo = new LunchOrder(2, 1, 11, 5);  
LunchOrder friesInfo = new LunchOrder(1.3, 11, 36, 4);  
LunchOrder sodaInfo = new LunchOrder(0.95, 0, 38, 0);
```

Prompted the user to input the amount of each item ordered, initializing it. Then displayed the nutritional information using that item's object and the method that returns the nutritional information. For each item, it's cost is calculated and added to the total using the totalAmt object with the user input for the amount of each item and the items' price as the parameters.

```
//Prompt user to input an integer for each item and initialize it, display info about each item  
System.out.print("Enter number of hamburgers: ");  
burgerAmt = userInput.nextInt();  
System.out.println(burgerInfo.burgers());  
totalAmt.burgerCost(burgerAmt, burgerInfo.getPrice());
```

```
System.out.print("Enter number of salads: ");  
saladAmt = userInput.nextInt();  
System.out.println(saladInfo.salads());  
totalAmt.saladCost(saladAmt, saladInfo.getPrice());
```

```
System.out.print("Enter number of fries: ");  
fryAmt = userInput.nextInt();  
System.out.println(friesInfo.fries());  
totalAmt.fryCost(fryAmt, friesInfo.getPrice());
```

```
System.out.print("Enter number of sodas: ");  
sodaAmt = userInput.nextInt();  
System.out.println(sodaInfo.sodas());  
totalAmt.sodaCost(sodaAmt, sodaInfo.getPrice());
```

Display the total cost of the items ordered using the totalAmt object and the total method.

```
//Display total  
System.out.println(totalAmt.total());
```

```

    LAUNCH = LAUNCH,
}
public void setTotal(double totalAmt) {
    total = totalAmt;
}

```

Created a string method for each menu item that returns the nutrition information.

```

//Return display info for menu items
public String burgers() {
    return ("Each hamburger has " + fats + "g of fat, " + carbs + "g of carbs, and " + fibers + "g of fiber.");
}
public String salads() {
    return ("Each salad has " + fats + "g of fat, " + carbs + "g of carbs, and " + fibers + "g of fiber.");
}
public String fries() {
    return ("French fries have " + fats + "g of fat, " + carbs + "g of carbs, and " + fibers + "g of fiber.");
}
public String sodas() {
    return ("Each soda has " + fats + "g of fat, " + carbs + "g of carbs, and " + fibers + "g of fiber.");
}

```

Calculate the cost of each menu item using the number of items(int) ordered and its price(double) as the parameters.

Then add the cost of the item, the amount of items multiplied by the price, to the total.

```

//Calculate burger price
public void burgerCost(int burgerAmt, double price)
    total += (burgerAmt * price);
}

//Calculate salad price
public void saladCost(int saladAmt, double price) {
    total += (saladAmt * price);
}

//Calculate salad price
public void fryCost(int fryAmt, double price) {
    total += (fryAmt * price);
}

//Calculate salad price
public void sodaCost(int sodaAmt, double price) {
    total += (sodaAmt * price);
}

```

Created a string method the returns the total cost of all the items.

```

//Display total price
public String total() {
    return "Your order comes to: $" + df.format(total);
}

```