Task 1

#include<iostream>

```
#include <iomanip>
using namespace std;

double calculateDiscountedTotal(double totalPrice) {
return (totalPrice > 200) ? totalPrice * 0.9 : totalPrice;
}

int main() {
double totalPrice;

cout << "Enter the total price: $";
cin >> totalPrice;

double discountedTotal = calculateDiscountedTotal(totalPrice);

cout << "Discounted Total:$"
<<fixed <<setprecision(2)
<< discountedTotal <<endl;

return 0;
}
```

Task 2

```
#include <iostream>
#include <iomanip>
using namespace std;
```

```
double calculateDiscountedTotal(double totalPrice) {
    return (totalPrice > 200) ? totalPrice * 0.9 : totalPrice;
}

int main() {
    double totalPrice;

cout << "Enter the total price: $";
cin >> totalPrice;

    double discountedTotal = calculateDiscountedTotal(totalPrice);

cout << "Discounted Total:$"
    <<fixed <<setprecision(2)
    << discountedTotal <<endl;

    return 0;
}
```

Task 3

```
#include <iostream>
#include <iomanip>
using namespace std;

double calculateTotalWithShipping(double totalAfterDiscount) {
    const double shippingFee = 15.0;
    double totalWithShipping;


    if (totalAfterDiscount < 150) {
    totalWithShipping = totalAfterDiscount + shippingFee;
```

```
    } else
 {
    totalWithShipping = totalAfterDiscount;
    }

    return totalWithShipping;
}

int main() {
    double originalPrice;
    double discountPercentage;

    cout << "Enter the original price: $";
    cin >> originalPrice;

    cout << "Enter the discount percentage (%): ";
    cin >> discountPercentage;

    double discountAmount = originalPrice * (discountPercentage /
100);
    double totalAfterDiscount = originalPrice - discountAmount;

    double totalWithShipping =
calculateTotalWithShipping(totalAfterDiscount);

    cout << std::fixed << std::setprecision(2);
    cout << "Original Price: $" << originalPrice << std::endl;
    cout << "Discount Percentage: " << discountPercentage << "%"
<< std::endl;
    cout << "Discount Amount: $" << discountAmount << std::endl;
    cout << "Total After Discount: $" << totalAfterDiscount <<
std::endl;
    cout << "Total With Shipping: $" << totalWithShipping <<
std::endl;

    return 0;
```

```
}
```

Task 4

```
#include <iostream>
#include <iomanip>
using namespace std;
int calculateLoyaltyPoints(double totalWithShipping) {

    return (totalWithShipping > 300) ? 50 : 20;
}


int calculateLoyaltyPointsAlternative(double totalWithShipping) {
    if (totalWithShipping > 300) {
    return 50;
    } else {
    return 20;
    }
}

int main() {
    double originalPrice;
    double discountPercentage;

cout << "Enter the original price: $";
cin >> originalPrice;

cout << "Enter the discount percentage (%): ";
cin >> discountPercentage;

    double discountAmount = originalPrice * (discountPercentage /
100);
    double totalAfterDiscount = originalPrice - discountAmount;
```

```
    double totalWithShipping = (totalAfterDiscount < 150) ?
totalAfterDiscount + 15 : totalAfterDiscount;

    int loyaltyPoints = calculateLoyaltyPoints(totalWithShipping);

cout << std::fixed << std::setprecision(2);
cout << "Original Price: $" << originalPrice <<endl;
cout << "Discount Percentage: " << discountPercentage << "%"
<<endl;
cout << "Discount Amount: $" << discountAmount <<endl;
cout << "Total After Discount: $" << totalAfterDiscount <<endl;
cout << "Total With Shipping: $" << totalWithShipping <<endl;
cout << "Loyalty Points: " << loyaltyPoints <<endl;

    return 0;
}
```

Task 5

```
#include <iostream>
#include <iomanip>
using namespace std;

const double SHIPPING_FEE = 15.0;
const double DISCOUNT_THRESHOLD = 200.0;
const double DISCOUNT_PERCENTAGE = 10.0;
const double LOYALTY_POINTS_THRESHOLD = 300.0;
const int LOYALTY_POINTS_HIGH = 50;
const int LOYALTY_POINTS_LOW = 20;
```

```
double calculateDiscount(double subtotal) {
    return (subtotal > DISCOUNT_THRESHOLD) ? subtotal *
(DISCOUNT_PERCENTAGE / 100) : 0;
}

double calculateShippingCost(double subtotal) {
    return (subtotal < 150) ? SHIPPING_FEE : 0;
}

int calculateLoyaltyPoints(double total) {
    return (total > LOYALTY_POINTS_THRESHOLD) ?
LOYALTY_POINTS_HIGH : LOYALTY_POINTS_LOW;
}

int main() {

cout << "Product Selection:" << std::endl;
cout << "1. Product A ($100)" << std::endl;
cout << "2. Product B ($200)" << std::endl;
cout << "3. Product C ($300)" << std::endl;

    int productChoice;
cout << "Enter product choice (1/2/3): ";
cin >> productChoice;

    double productPrice;
    switch (productChoice) {
case 1:
 productPrice = 100;
break;
case 2:
  productPrice = 200;
break;
  case 3:
  productPrice = 300;
break;
```

```
default:
cerr << "Invalid product choice." << std::endl;
    return 1;
    }

    double subtotal = productPrice;

    double discount = calculateDiscount(subtotal);
    cout << "Discount: $" << std::fixed << std::setprecision(2)
<< discount << std::endl;


    double subtotalAfterDiscount = subtotal - discount;

    double shippingCost =
calculateShippingCost(subtotalAfterDiscount);
cout << "Shipping Cost: $" << std::fixed << std::setprecision(2)
<< shippingCost << std::endl;

    double total = subtotalAfterDiscount + shippingCost;


    int loyaltyPoints = calculateLoyaltyPoints(total);
cout << "Loyalty Points: " << loyaltyPoints << std::endl;


cout << "Final Total: $" << std::fixed << std::setprecision(2) <<
total << std::endl;

    return 0;
}
```