**RESEARCH**

**Loyalty Program Discount Calculator**

The tiered loyalty program is a membership program that rewards customers with different benefits based on their level of engagement or spending with a company. This program is structured into multiple tiers, with each tier offering increasingly valuable rewards as customers progress.

Tiered loyalty programs are different from other types of loyalty programs because of their accumulative structure. They can be an effective way for businesses to:

* Incentivize customer engagement:

Customers are motivated to take certain actions, such as making repeat purchases, engaging with the brand on social media, and leaving reviews.

* Increase customer loyalty:

Customers are more likely to visit a store and expand their purchase history over time

* Differentiate themselves from competitors:

Businesses can offer unique and memorable experiences to create a sense of exclusivity around their brand.

* Data collection and analysis

This program calculates discounts based on customer loyalty tiers and total purchase amounts. It prompts the user for their phone number, loyalty tier, and total purchase amount, then determines the applicable discount.

Here are some well-known examples of loyalty tier programs:

**1. Starbucks Rewards:**

* **Tiers**: Green (entry-level) and Gold (higher level)
* **Rewards**: Free in-store refills, free birthday drinks, personalized offers, and the ability to redeem points for food and beverages.
* **How it works**: Customers earn "Stars" for every purchase. Once a certain number of Stars are accumulated, they can redeem them for rewards. As they spend more, they move up to the Gold tier, which offers better rewards.
  1. **Amazon Prime:**
* **Tiers**: Prime (though it's a single tier, it's a membership program that works similarly to a tier system)
* **Rewards**: Free shipping, access to streaming services (Prime Video, Prime Music), Prime-exclusive deals, and more.
* **How it works**: Members pay an annual fee to access the benefits. While there's only one membership tier, the exclusivity and array of benefits mimic a high-tier loyalty program.

Loyalty tier programs can be used in a wide range of industries where customer retention and repeat business are important. Here are key areas where they can be implemented:

**1. Retail**

2. **Hospitality and Travel**

3. **Restaurants and Cafés**

4. Fitness & Wellness

5. Entertainment & Gaming

6. E-commerce

7. Healthcare & Pharmaceuticals

[**https://www.incentivesmart.com/blog/tiered-loyalty-programs**](https://www.incentivesmart.com/blog/tiered-loyalty-programs)

[**https://www.openloyalty.io/insider/effective-tiered-loyalty-programs**](https://www.openloyalty.io/insider/effective-tiered-loyalty-programs)

[**https://www.friendbuy.com/blog/tiered-rewards-programs**](https://www.friendbuy.com/blog/tiered-rewards-programs)

[**https://www.w3schools.com/c/c\_do\_while\_loop.php**](https://www.w3schools.com/c/c_do_while_loop.php)

[**https://www.programiz.com/c-programming/c-do-while-loops**](https://www.programiz.com/c-programming/c-do-while-loops)

**ANALYZE**

For this program we use C language in which we use

* If else loop

**if Statement**: The if statement evaluates a condition. If the condition is true, the block of code inside the if statement is executed

if (condition) {

// Code to execute if condition is true

}

**else Statement**: The else statement follows an if statement and executes its block of code if the condition in the if statement is false.

if (condition) {

// Code to execute if condition is true

} else {

// Code to execute if condition is false

}

This code also uses the **do while loop** for entering the customers phone number and their loyalty tier

A do-while loop is a control flow statement that allows code to be executed repeatedly based on a condition, with the key difference from a regular while loop being that the condition is checked **after** the loop body is executed. This guarantees that the loop runs **at least once**, regardless of whether the condition is true or false at the beginning.

do {  
*// code block to be executed*}  
while (*condition*);

**IDEATE**

Discount Logic:

* Initializes a discount variable to 0.
* Checks the loyalty tier and total purchase amount to determine the applicable discount.
* For VIP members:
  + 20% discount for purchases of $5000 or more.
  + 10% discount for purchases between $2500 and $4999.
* For Gold members:
  + 10% discount for purchases of $5000 or more.
  + 5% discount for purchases between $2500 and $4999.
* Regular members receive no discount.

**Calculating and Displaying Discounts**:

* If a discount applies, calculates the discount amount and the final price after applying the discount.
* Displays the discount percentage, total discount amount, and the final price.
* If no discount is available, it informs the user.

**BUILD**

#include <stdio.h>

#include <string.h>

int main(){

int loyalty\_tier;

int total\_price;

char phoneNumber[11];

const int VIP = 1;

const int Gold = 2;

const int Regular = 3;

do {

printf("Enter a 10-digit phone number: ");

scanf("%10s", phoneNumber);

if(strlen(phoneNumber) == 10) {

printf("Phone number accepted.\n");

} else {

printf("Invalid phone number. Please enter exactly 10 digits.\n");

}

} while(strlen(phoneNumber) != 10);

do {

printf("\nPlease enter your loyalty tier (1 for VIP, 2 for Gold, 3 for Regular): ");

scanf("%d", &loyalty\_tier); // Capture loyalty tier input

if(loyalty\_tier == 1 || loyalty\_tier == 2 || loyalty\_tier == 3) {

printf("Thank you!\n");

} else {

printf("Invalid loyalty tier. Please enter a valid number (1, 2, or 3).\n");

}

} while(loyalty\_tier != 1 && loyalty\_tier != 2 && loyalty\_tier != 3);

printf("Enter the total amount of your purchase: ");

scanf("%d", &total\_price); // %d to capture integer

if(loyalty\_tier == VIP && total\_price >= 5000){

printf("\nYou have got a 20 percent discount!!");

}else if(loyalty\_tier == VIP && total\_price >= 2500){

printf("\nYou have got a 10 percent discount!!");

}else if(loyalty\_tier == Gold && total\_price >= 5000){

printf("\nYou have got a 10 percent discount!!");

}else if(loyalty\_tier == Gold && total\_price >= 2500){

printf("\nYou have got a 5 percent discount!!");

}else{

printf("\nNo special offers.");

}

printf("\nThank you for shopping with us!");

return 0;

}

**TEST**

After we run our code, we get the following output-

Enter a 10-digit phone number:

After we put a number we get the following output

Phone number accepted.

Please enter your loyalty tier (1 for VIP, 2 for Gold, 3 for Regular):

After entering the correct value

Thank you!

Enter the total amount of your purchase:

8000

So according to the conditions

You have got a 10 percent discount!!

Thank you for shopping with us!

**IMPLEMENT**

This program calculates discounts based on customer loyalty tiers and total purchase amounts. It prompts the user for their phone number, loyalty tier, and total purchase amount, then determines the applicable discount.

Here are some well-known examples of loyalty tier programs:

**1. Starbucks Rewards:**

* **Tiers**: Green (entry-level) and Gold (higher level)
* **Rewards**: Free in-store refills, free birthday drinks, personalized offers, and the ability to redeem points for food and beverages.
* **How it works**: Customers earn "Stars" for every purchase. Once a certain number of Stars are accumulated, they can redeem them for rewards. As they spend more, they move up to the Gold tier, which offers better rewards.
  1. **Amazon Prime:**
* **Tiers**: Prime (though it's a single tier, it's a membership program that works similarly to a tier system)
* **Rewards**: Free shipping, access to streaming services (Prime Video, Prime Music), Prime-exclusive deals, and more.
* **How it works**: Members pay an annual fee to access the benefits. While there's only one membership tier, the exclusivity and array of benefits mimic a high-tier loyalty program.

**Summary**

* The program effectively handles user input for phone numbers and loyalty tiers, validates them, and calculates applicable discounts based on specified rules. It provides clear feedback and outputs relevant information to the user

[**https://github.com/swaradadeshpande/BOOST-SALES-WITH-REBATES**](https://github.com/swaradadeshpande/BOOST-SALES-WITH-REBATES)

**CODE LINK:**

**https://www.programiz.com/online-compiler/5EGGhcOrcktPI**