Writing C++ Functions using complex Conditional Statements to solve real world problems

اَللّٰهُمَّ اُرْزُقْنِى عِلْمًا نَافِعًا وَاسِعًا عَمِيْقًا

اَللّٰهُمَّ اُرْزُقْنِى رِزْقًا وَاسِعًا حَلَالًا طَيِّبًا
مُبَارَكًا مِنْ عِنْدِكَ

Review: Conditions

In the previous Lecture, we saw different categories of Conditional Statements.

Single IF Statement	IF Else Statement	Multiple IF Statements	Nested IF Statements

Review: Conditions

Now, what are the different situations in which each of these are used?

Single IF Statement

When there is only one Condition

IF Else Statement

Multiple IF Statements

Nested IF Statements

Review: Conditions

Now, what are the different situations in which each of these are used?

Single IF Statement

When there is only one Condition

IF Else Statement

When there are two conditions and they are contradicting

Multiple IF Statements

Nested IF Statements

Review: Conditions

Now, what are the different situations in which each of these are used?

Single IF Statement

When there is only one Condition

IF Else Statement

When there are two conditions and they are contradicting

Multiple IF Statements

When there are two or more conditions and they are independent of each other

Nested IF Statements

Review: Conditions

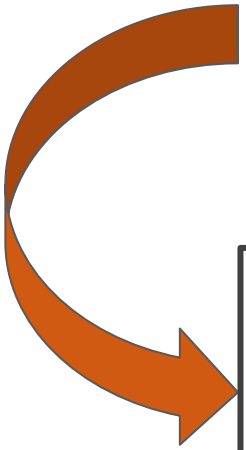
Now, what are the different situations in which each of these are used?

Single IF Statement	IF Else Statement	Multiple IF Statements	Nested IF Statements
When there is only one Condition	When there are two conditions and they are contradicting	When there are two or more conditions and they are independent of each other	When there are two conditions and they are dependent on each other

Review: Conditions

Multiple IF Statements

Nested IF Statements



Multiple IF Statements can be replaced with **OR Logical Operators** if the conditions are different but you have to **perform the same task** under those conditions

Review: Conditions

Multiple IF Statements

Nested IF Statements

Multiple IF Statements can be replaced with **OR Logical Operators** if the conditions are different but you have to **perform the same task** under those conditions

Nested IF Statements can be replaced with **AND Logical Operators** for Simplicity of the Code

Practice makes Us Perfect

Now, let's try to categorize the different problems in into these categories

Problem 01

Kaka is an entrepreneur and he has opened a small shop in Lahore with different prices for the following products:

Product	Lahore
Coffee	400
Sweets	200
Water	50

Write a function that takes product, City, and quantity as input and returns the total payable price.

Problem 01: Test Cases

Input	Output
calculatePrice (“Coffee”, “Lahore”, 2)	Price: 800
calculatePrice (“Water”, “Lahore”, 3)	Price: 150
calculatePrice (“Sweets”, “Lahore”, 1)	Price: 200

Steps: Towards Solution

What is the category of conditional statements, this problem lies?

Steps: Towards Solution

What is the category of conditional statements, this problem lies?

There are 3 conditions and they are all independent.

Steps: Towards Solution

What is the category of conditional statements, this problem lies?

There are 3 conditions and they are all independent.

Multiple IF Statements

When there are two or more conditions and they are independent of each other

Solution

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if(product == "Coffee")
    {
        price = quantity * 400;
    }
    if(product == "Sweets")
    {
        price = quantity * 200;
    }
    if(product == "Water")
    {
        price = quantity * 50;
    }
    return price;
}
```

Problem 02

Kaka lives in Pakistan and due to Inflation (مہنگائی) he has to update his products prices. Now, the updated Prices are:

Product	Lahore
Coffee	400
Sweets	400
Water	100

Write a function that takes product, City, and quantity as input and returns the total payable price.

Problem 02: Test Cases

Input	Output
calculatePrice (“Coffee”, “Lahore”, 2)	Price: 800
calculatePrice (“Water”, “Lahore”, 3)	Price: 300
calculatePrice (“Sweets”, “Lahore”, 1)	Price: 400

Steps: Towards Solution

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if(product == "Coffee")
    {
        price = quantity * 400;
    }
    if(product == "Sweets")
    {
        price = quantity * 200;
    }
    if(product == "Water")
    {
        price = quantity * 50;
    }
    return price;
}
```

What are the changes that we have to do to update the solution?

Steps: Towards Solution

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if(product == "Coffee")
    {
        price = quantity * 400;
    }
    if(product == "Sweets")
    {
        price = quantity * 200;
    }
    if(product == "Water")
    {
        price = quantity * 50;
    }
    return price;
}
```

Steps: Towards Solution

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if(product == "Coffee")
    {
        price = quantity * 400;
    }
    if(product == "Sweets")
    {
        price = quantity * 400;
    }
    if(product == "Water")
    {
        price = quantity * 100;
    }
    return price;
}
```

Can this solution further be improved?

Steps: Towards Solution

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if(product == "Coffee")
    {
        price = quantity * 400;
    }
    if(product == "Sweets")
    {
        price = quantity * 400;
    }
    if(product == "Water")
    {
        price = quantity * 100;
    }
    return price;
}
```

Can this solution further be improved? Which category this problem belongs to?

Steps: Towards Solution

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if(product == "Coffee")
    {
        price = quantity * 400;
    }
    if(product == "Sweets")
    {
        price = quantity * 400;
    }
    if(product == "Water")
    {
        price = quantity * 100;
    }
    return price;
}
```

Multiple IF
Statements can be replaced with **OR Logical Operators** if the conditions are different but you have to **perform the same task** under those conditions

Solution

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if(product == "Coffee" || product == "Sweets")
    {
        price = quantity * 400;
    }
    if(product == "Water")
    {
        price = quantity * 100;
    }
    return price;
}
```

Problem 03

Now, Kaka's business is expanding and he has opened another shop in another city with the following prices:

Product	Lahore	Faisalabad
Coffee	400	350
Sweets	200	250
Water	50	40

Write a function that takes product, City, and quantity as input and returns the total payable price.

Problem 03: Test Cases

Input	Output
calculatePrice (“Coffee”, “Lahore”, 2)	Price: 800
calculatePrice (“Water”, “Lahore”, 3)	Price: 150
calculatePrice (“Sweets”, “Faisalabad”, 1)	Price: 250

Steps: Towards Solution

What is the category of conditional statements, this problem lies?

Steps: Towards Solution

What is the category of conditional statements, this problem lies?

There are conditions that are dependent on each other.

Steps: Towards Solution

What is the category of conditional statements, this problem lies?

There are conditions that are dependent on each other.

For Example: If the city is Faisalabad and the Product is Sweets then…

Steps: Towards Solution

What is the category of conditional statements, this problem lies?

There are conditions that are dependent on each other.

Nested IF Statements

When there are two conditions and they are dependent on each other

Solution

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if(product == "Coffee")
    {
        if(city == "Lahore")
            price = quantity * 400;
        if(city == "Faisalabad")
            price = quantity * 350;
    }
```

Solution

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if(product == "Coffee")
    {
        if(city == "Lahore")
            price = quantity * 400;
        if(city == "Faisalabad")
            price = quantity * 350;
    }
    if(product == "Sweets")
    {
        if(city == "Lahore")
            price = quantity * 200;
        if(city == "Faisalabad")
            price = quantity * 250;
    }
```

Solution

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if(product == "Coffee")
    {
        if(city == "Lahore")
            price = quantity * 400;
        if(city == "Faisalabad")
            price = quantity * 350;
    }
    if(product == "Sweets")
    {
        if(city == "Lahore")
            price = quantity * 200;
        if(city == "Faisalabad")
            price = quantity * 250;
    }
    if(product == "Water")
    {
        if(city == "Lahore")
            price = quantity * 50;
        if(city == "Faisalabad")
            price = quantity * 40;
    }
    return price;
}
```

Solution

Can this solution further be improved?

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if(product == "Coffee")
    {
        if(city == "Lahore")
            price = quantity * 400;
        if(city == "Faisalabad")
            price = quantity * 350;
    }
    if(product == "Sweets")
    {
        if(city == "Lahore")
            price = quantity * 200;
        if(city == "Faisalabad")
            price = quantity * 250;
    }
    if(product == "Water")
    {
        if(city == "Lahore")
            price = quantity * 50;
        if(city == "Faisalabad")
            price = quantity * 40;
    }
    return price;
}
```

Solution

Can this solution further be improved?

Nested IF Statements can be replaced with **AND Logical Operators** for Simplicity of the Code

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if(product == "Coffee")
    {
        if(city == "Lahore")
            price = quantity * 400;
        if(city == "Faisalabad")
            price = quantity * 350;
    }
    if(product == "Sweets")
    {
        if(city == "Lahore")
            price = quantity * 200;
        if(city == "Faisalabad")
            price = quantity * 250;
    }
    if(product == "Water")
    {
        if(city == "Lahore")
            price = quantity * 50;
        if(city == "Faisalabad")
            price = quantity * 40;
    }
    return price;
}
```

Solution

Solution with AND logical Operators

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if (product == "Coffee" && city == "Lahore"){
        price = quantity * 400;
    }
    if (product == "Coffee" && city == "Faisalabad"){
        price = quantity * 350;
    }
    if (product == "Sweets" && city == "Lahore"){
        price = quantity * 200;
    }
    if (product == "Sweets" && city == "Faisalabad"){
        price = quantity * 250;
    }
    if (product == "Water" && city == "Lahore"){
        price = quantity * 50;
    }
    if (product == "Water" && city == "Faisalabad"){
        price = quantity * 40;
    }
    return price;
}
```

Problem 04: Complex Problem

Now, Kaka's business is expanding and he has opened another shop in another city with the following prices:

Product	Lahore	Faisalabad	Karachi
Coffee	400	350	300
Sweets	200	250	300
Water	50	40	60

Write a function that takes product, City, and quantity as input and returns the total payable price.

Problem 04: Test Cases

Input	Output
calculatePrice (“Coffee”, “Karachi”, 2)	Price: 600
calculatePrice (“Water”, “Lahore”, 3)	Price: 150
calculatePrice (“Sweets”, “Faisalabad”, 1)	Price: 250

Solution

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if (product == "Coffee" && city == "Lahore")
        price = quantity * 400;
    if (product == "Coffee" && city == "Faisalabad")
        price = quantity * 350;
    if (product == "Sweets" && city == "Lahore")
        price = quantity * 200;
    if (product == "Sweets" && city == "Faisalabad")
        price = quantity * 250;
    if (product == "Water" && city == "Lahore")
        price = quantity * 50;
    if (product == "Water" && city == "Faisalabad")
        price = quantity * 40;
    if ((product == "Coffee" || product == "Sweets") && city == "Karachi")
        price = quantity * 300;
    if (product == "Water" && city == "Karachi")
        price = quantity * 60;
    return price;
}
```

Solution with Else If

```
int calculatePrice(string product, string city, int quantity)
{
    int price;
    if (product == "Coffee" && city == "Lahore")
        price = quantity * 400;
    else if (product == "Coffee" && city == "Faisalabad")
        price = quantity * 350;
    else if (product == "Sweets" && city == "Lahore")
        price = quantity * 200;
    else if (product == "Sweets" && city == "Faisalabad")
        price = quantity * 250;
    else if (product == "Water" && city == "Lahore")
        price = quantity * 50;
    else if (product == "Water" && city == "Faisalabad")
        price = quantity * 40;
    else if ((product == "Coffee" || product == "Sweets") && city == "Karachi")
        price = quantity * 300;
    else if (product == "Water" && city == "Karachi")
        price = quantity * 60;
    return price;
}
```

Learning Objective

In this lecture, we learnt how to categorize the problems into different conditional statements categories and then solve the problem more efficiently.

Self Assessment

Now, Write a C++ program to give a 10% raise in salary if the experience exceeds 15 years, or the employee's position is AP. The program should take salary, experience, and employee's position as input, and the program should return the updated salary.

Self Assessment

The user entered the employee position as **AP**, and the experience is greater than **15 years**; therefore, the program gave a raise of 10% in the salary

```
C:\C++>c++ example.cpp -o example.exe

C:\C++>example.exe
Enter the Salary: 50000
Enter the Position: AP
Enter The Experience: 19
Increased Salary: 55000

C:\C++>
```

The user entered the employee position as **Lecturer**, and the experience is less than **15 years**; therefore, the program did not give a raise of 10% in the salary.

```
C:\C++>c++ example.cpp -o example.exe

C:\C++>example.exe
Enter the Salary: 30000
Enter the Position: Lecturer
Enter The Experience: 10
No increase in Salary

C:\C++>
```

Self Assessment

Write a C++ program to give a 10% raise in the salary if the employee's position is not AP. The program should take the salary, the experience and the employee's position as input. The program should return the updated salary if the employee position is not AP; otherwise, there will not be any raise in the salary.

Self Assessment

the user entered the employee position as AP; therefore, the program did not give a 10% raise in salary.

```
C:\C++>c++ example.cpp -o example.exe

C:\C++>example.exe
Enter the Salary: 60000
Enter the Position: AP
Enter the Experience: 15
No increase in Salary

C:\C++>
```

The user entered the employee position as **Lecturer**, and the experience is less than **15 years**; therefore, the program did not give a raise of 10% in the salary.

```
C:\C++>c++ example.cpp -o example.exe

C:\C++>example.exe
Enter the Salary: 60000
Enter the Position: LE
Enter the Experience: 15
Increased Salary: 66000

C:\C++>
```