

---

# PROJECT TEMPERATURE CONVERTER

PREPARED BY

**Sana Fatima**

---

sana.faati@gmail.com

PREPARED TO

**A&D Tech**

---

ad.techinnov25@gmail.com



---

# TABLE OF CONTENTS

Problem Statement



Solution Overview



Challenges and Resolution



Instructions



---

## Problem Statement

### Implementing the Program

- Implement two functions: Fahrenheit to Celsius(`fahrenheit`) which converts temperature from Fahrenheit to Celsius. And Celsius to Fahrenheit(`celsius`) which converts temperature from Celsius to Fahrenheit.
- Each function should take a temperature value as an input, perform the conversion, and return the converted value.
- Asks users whether they want to convert from Fahrenheit to Celsius or vice versa.

- Takes the user's temperature input. Calls the appropriate conversion function and displays the result.
- Include an option to convert another temperature or exit the program.
- Ensure that your functions return accurate conversions. Test edge cases like freezing point, boiling point, and negative temperatures.

---

## SOLUTION OVERVIEW

The solution overview for this program is:

The provided code is a simple C++ program that converts temperatures between Fahrenheit and Celsius. The program includes two main functions:

1. `FahrenheitToCelsius(int fahrenheit)`: Converts Fahrenheit to Celsius and prints the result.
2. `CelsiusToFahrenheit(int celsius)`: Converts Celsius to Fahrenheit and prints the result.

The `main` function serves as the user interface, allowing users to choose between the two conversion types or exit the program. It also handles invalid input to ensure the program runs smoothly.

---

## CHALLENGES AND RESOLUTIONS

### ❑ Challenge: Handling Invalid Input

- **Problem:** Users might enter invalid input (e.g., non-numeric values) for temperature, causing the program to crash or behave unexpectedly.
- **Resolution:** The program uses a `while(true)` loop with `cin` validation to ensure that only valid numeric input is accepted. If the input is invalid, it clears the error state and ignores the invalid input, prompting the user to reenter the temperature.

- **Challenge: Accurate Conversion**

- **Problem:** Ensuring the conversion formulae are correct and return accurate results.
- **Resolution:** The code uses the correct formulas for temperature conversion:

- **Challenge: User-Friendly Interface**

- **Problem:** Designing a console interface that is easy to use and understand.
- **Resolution:** The program provides clear instructions and prompts. It uses a `do-while` loop to repeatedly ask the user for their choice and temperature input until they choose to exit.

---

# INSTRUCTIONS

- The `main` function displays a menu with three options:
  - Convert Fahrenheit to Celsius.
  - Convert Celsius to Fahrenheit.
  - Exit the program.
- The user selects an option and enters the temperature to convert. The program then calls the appropriate conversion function and displays the result.
- The program allows users to perform multiple conversions by looping back to the menu until they choose to exit.
- **Testing Your Functions:**
- Test the functions with various inputs to ensure accuracy. Consider edge cases like:
  - Freezing point of water: 32°F (0°C)
  - Boiling point of water: 212°F (100°C)
  - Negative temperatures: -40°F (same in °C)

