Level 2: Intermediate

Task 4: Build a temperature converter program.

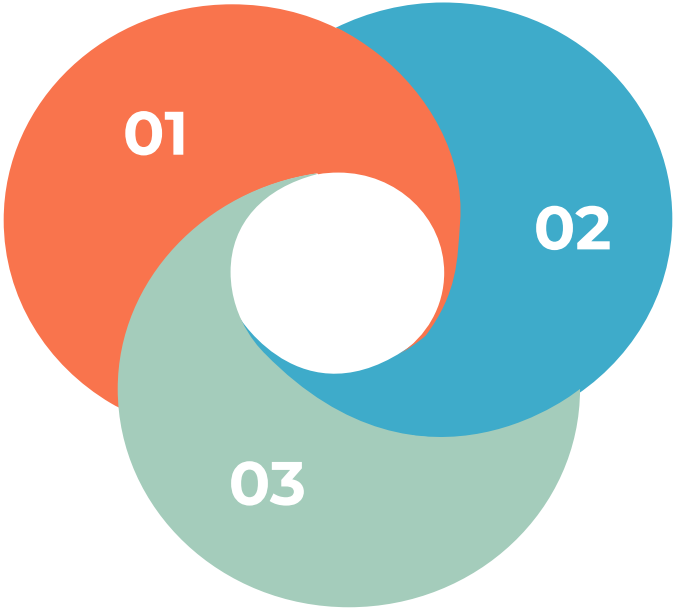
By Shradha Pujari

Building a Temperature Converter Program

This presentation will guide you through the steps to build a temperature converter program, which allows users to input temperatures and choose the conversion direction between Fahrenheit and Celsius.

Step 1: Design the Program

Create a user interface to accept temperature input.



Consider the layout and design of the program.

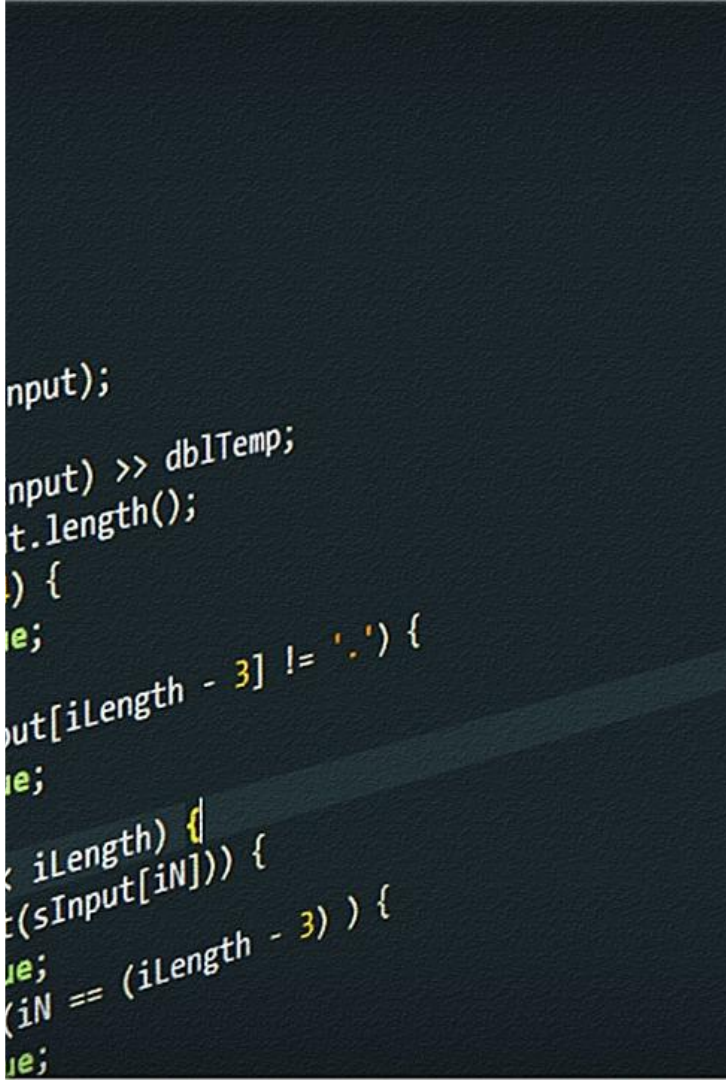
Ensure the input validation for temperature values.

Step 2: Implement Temperature Conversion Logic

01 Consider the formula for conversion: T(°C) = (T(°F) - 32) × 5/9 and T(°F) = T(°C) × 9/5 + 32.

02 Write the code to convert temperature values.

03 Handle the conversion between Fahrenheit and Celsius.

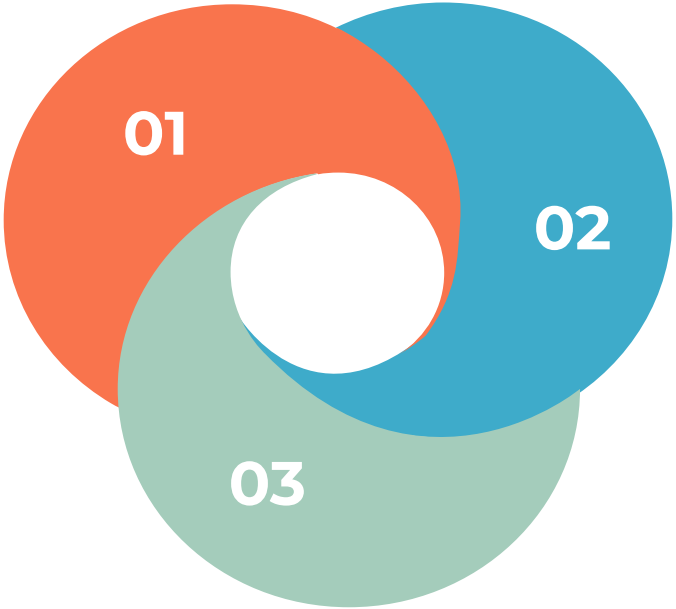


Step 3: Allow Conversion Direction Choice

- Add an option for users to choose the conversion direction.
- Implement a userfriendly interface for selecting the conversion direction.
- Ensure proper functionality and error handling for the chosen direction.

Step 4: Test the Program

Test the program with different input values.



Check for accuracy in conversions.

Ensure the program handles edge cases and invalid input gracefully.

Code

```
def get_temperature():
 """Prompts user for temperature input and validates it as a number."""
 while True:
  try:
   temperature = float(input("Enter temperature: "))
   return temperature
  except ValueError:
   print("Invalid input. Please enter a number.")
```

Code

```
def get_conversion_choice():
 """Prompts user for conversion direction (Celsius to Fahrenheit or vice versa)"""
 while True:
  choice = input("Enter conversion direction (C to F or F to C): ").upper()
  if choice in ('C', 'F'):
   return choice
  else:
   print("Invalid choice. Please enter 'C' or 'F'.")
```

Code

```
def convert_temperature(temperature, from_unit, to_unit):
 """Converts temperature based on user input."""
 if from_unit == 'C' and to_unit == 'F':
  return (temperature * 9/5) + 32
 elif from_unit == 'F' and to_unit == 'C':
  return (temperature - 32) * 5/9
 else:
  print("Error: Invalid conversion direction.")
  return None  # Indicate an error
```

Code

```
def main():
    """Main function to handle user interaction and temperature conversion."""
    temperature = get_temperature()
    conversion_choice = get_conversion_choice()
    from_unit = 'C' if conversion_choice == 'C' else 'F'
    to_unit = 'F' if conversion_choice == 'C' else 'C'
    converted_temperature = convert_temperature(temperature, from_unit, to_unit)
```

Code

```
if converted_temperature is not None:

   print(f"{temperature:.2f} degrees {from_unit} is equal to {converted_temperature:.2f} degrees {to_unit}.")



if __name__ == "__main__":

  main()
```

Code Explanation:

1. **get_temperature():** This function prompts the user for temperature input and validates it using a try-except block to ensure a numerical value.
2. **get_conversion_choice():** This function prompts the user for the conversion direction (Celsius to Fahrenheit or vice versa) and validates the input to ensure 'C' or 'F'.
3. **convert_temperature():** This function performs the temperature conversion based on the user's input. It checks the conversion direction and applies the appropriate formula. It also handles potential errors.
4. **main():** This function is the program's entry point. It calls the other functions to get user input, perform conversion, and display the result.

Output

1. **Enter temperature:**

```
Enter temperature: 86
```

2. **Choose conversion direction (F to C):**

```
Enter conversion direction (C to F or F to C): F
```

Output:

```
86.00 degrees F is equal to 30.00 degrees C.
```

Thank you for your time and attention 🙂