```
def fullname(first_name, last_name):
    return first_name + " " + last_name

def string_alternative(full_name):
    return full_name[::2]

def main():
    # input from the user
    first_name = input('Enter first name ')
    last_name = input('Enter last name ')


    full_name = fullname(first_name, last_name)
    print('Full name', full_name)


    alt_str = string_alternative(full_name)
    print('String with every other character', alt_str)

if __name__ == "__main__":
    main()
```

```
Enter first name jeevan
Enter last name nadi
Full name jeevan nadi
String with every other character jea ai
```

In the above code first I have created two functions such as fullname and string alternative function which takes the input from the user as first name and last name.herew the full name fuctions the returns the full name of the user by string concatination and the second function returns the every alternative character fron the fullname.

```
def word_count_in_line(line):
    words = line.split()
    word_count = {}
    for word in words:
        word = word.strip().strip('.,!?').capitalize()  # Capitalize
        word_count[word] = word_count.get(word, 0) + 1
    return word_count

def main():
        # Open the input file
        with open("input.txt", "r") as infile:
            lines = infile.readlines()

        # Create output file
        with open("output.txt", "w") as outfile:
            for line in lines:
                outfile.write(line)

            outfile.write("\nWord_Count:\n")

            # Count words for each line and write to the output file
            total_word_count = {}
            for line in lines:
                line_word_count = word_count_in_line(line)
                for word, count in line_word_count.items():
                    total_word_count[word] = total_word_count.get(word, 0) + count

            # total word count to the output file
            for word, count in total_word_count.items():
                outfile.write(f"{word}: {count}\n")

        print("Word count has been written to output.txt")


if __name__ == "__main__":
    main()
```

```
Word count has been written to output.txt
```

word_count_in_line function separates a line of text into words and counts how many times each word appears. It also standardized the words by removing punctuation and capitalizing the initial letter. main function:

Reads the content of input.txt. Writes each line to output.txt. After writing the lines, it calculates the word count for all lines combined and writes the word count to output.txt.

```
def inches_to_cm(inches):
    return round(inches * 2.54, 2)

def main():
    # input heights from the user and storing them in a list (L1)
    L1 = []
    n = int(input("Enter the number of heights: "))
    for i in range(n):
        height = float(input(f"Enter height in inches L1 {i + 1}: "))
        L1.append(height)
    print("list L1:",L1)
    #Nested Interactive Loop
    L2_nested_loop = []
    for height in L1:
        cm_height = inches_to_cm(height)
        L2_nested_loop.append(cm_height)

    print("Converted heights using L2(nested):", L2_nested_loop)

    # List Comprehensions
    L2_list_comprehension = [inches_to_cm(height) for height in L1]

    print("Converted heights using L2(list comprehensions)", L2_list_comprehension)

if __name__ == "__main__":
    main()
```

```
Enter the number of heights: 2
Enter height in inches L1 1: 52
Enter height in inches L1 2: 52
list L1: [52.0, 52.0]
Converted heights using L2(nested): [132.08, 132.08]
Converted heights using L2(list comprehensions) [132.08, 132.08]
```

inches to cm function converts the heights in inches to cm by multiplying 2.54 and rounding off two digits.

Nested Interactive Loop:

It then iterates through the list L1 and converts each height to centimeters using the inches_to_cm function. The converted heights are stored in L2_nested_loop.

List Comprehensions:

This piece of code performs the same conversion using a list comprehension,the converted heights are stored in L2_list_comprehension.