

**Name: Mohd Hamza Abbasi**

**PRN: 22070122125**

**Batch: SY CSE B**

**Questions:-**

1. WAP to check if given number is palindrome or not ?

```
import java.util.Scanner;

public class PalindromeNumber {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = scanner.nextInt();

        if (isPalindrome(num))
            System.out.println(num + " is a palindrome.");
        else
            System.out.println(num + " is not a palindrome.");
    }

    public static boolean isPalindrome(int num) {
        int originalNum = num;
        int reverseNum = 0;

        while (num != 0) {
            int digit = num % 10;
            reverseNum = reverseNum * 10 + digit;
            num /= 10;
        }

        return originalNum == reverseNum;
    }
}
```

2. WAP to check if given String is palindrome or not

```
import java.util.Scanner;

public class PalindromeString {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String str = scanner.nextLine();

        if (isPalindrome(str))
            System.out.println(str + " is a palindrome.");
        else
            System.out.println(str + " is not a palindrome.");
    }
}
```

```

    }

    public static boolean isPalindrome(String str) {
        String reversedStr = new StringBuilder(str).reverse().toString();
        return str.equals(reversedStr);
    }
}

```

3. WAP to create 2 variables as final and try to change their value to see the error.

```

public class FinalVariables {
    public static void main(String[] args) {
        final int x = 10;
        // Trying to change the value of a final variable will result in a compile-time error
        // x = 20; // This will result in a compilation error

        final String str = "Hello";
        // str = "World"; // This will also result in a compilation error
    }
}

```

4. WAP to perform linear search.

```

public class LinearSearch {
    public static void main(String[] args) {
        int[] arr = {4, 2, 9, 6, 1, 5, 8};
        int target = 5;

        int index = linearSearch(arr, target);

        if (index != -1)
            System.out.println(target + " found at index " + index);
        else
            System.out.println(target + " not found in the array.");
    }

    public static int linearSearch(int[] arr, int target) {
        for (int i = 0; i < arr.length; i++) {
            if (arr[i] == target)
                return i;
        }
        return -1;
    }
}

```

5. WAP to swap 2 numbers using call by value in Java. How swapping using call by reference will be performed in Java? Demonstrate with a program

```

public class SwapByValue {
    public static void main(String[] args) {
        int a = 5;
        int b = 10;

        System.out.println("Before swapping: a = " + a + ", b = " + b);
        swap(a, b);
    }
}

```

```

        System.out.println("After swapping: a = " + a + ", b = " + b);
    }

    public static void swap(int x, int y) {
        int temp = x;
        x = y;
        y = temp;
    }
}

```

6. WAP to demonstrate ragged array in Java –

```

public class RaggedArrayDemo {
    public static void main(String[] args) {
        int[][] raggedArray = {
            {1, 2, 3},
            {4, 5},
            {6, 7, 8, 9}
        };

        // Accessing elements of the ragged array
        for (int i = 0; i < raggedArray.length; i++) {
            for (int j = 0; j < raggedArray[i].length; j++) {
                System.out.print(raggedArray[i][j] + " ");
            }
            System.out.println();
        }
    }
}

```

7. WAP to enter and print ragged array in Java using for loop.

```

import java.util.Scanner;

public class RaggedArrayInput {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Input the size of the ragged array
        System.out.print("Enter the number of rows: ");
        int rows = scanner.nextInt();
        int[][] raggedArray = new int[rows][];

        // Input elements of the ragged array
        for (int i = 0; i < rows; i++) {
            System.out.print("Enter the number of elements in row " + i + ": ");
            int cols = scanner.nextInt();
            raggedArray[i] = new int[cols];

            System.out.println("Enter elements for row " + i + ":");
            for (int j = 0; j < cols; j++) {
                raggedArray[i][j] = scanner.nextInt();
            }
        }
    }
}

```

```
// Print the ragged array
System.out.println("Ragged Array:");
for (int i = 0; i < raggedArray.length; i++) {
    for (int j = 0; j < raggedArray[i].length; j++) {
        System.out.print(raggedArray[i][j] + " ");
    }
    System.out.println();
}
}
```