

DAY 23:

ASSIGNMENT 9:

Task 9: Generic Classes and Methods

Implement a generic method that swaps the positions of two elements in an array, regardless of their type, and demonstrate its usage with different object types.

ANSWER:

To implement a generic method that swaps the positions of two elements in an array regardless of their type, you can use Java's generics. Here's how you can do it:

Generic Swap Method

```
public class ArrayUtils {

    // Generic method to swap two elements in an array
    public static <T> void swap(T[] array, int index1, int index2) {
        if (array == null || index1 < 0 || index2 < 0 || index1 >= array.length || index2 >= array.length) {
            throw new IllegalArgumentException("Invalid index or null array");
        }

        T temp = array[index1];
        array[index1] = array[index2];
        array[index2] = temp;
    }

    // Main method to demonstrate the usage of the swap method
    public static void main(String[] args) {
        // Demonstrate with an array of Integers
        Integer[] intArray = {1, 2, 3, 4, 5};

        System.out.println("Original Integer array: " + java.util.Arrays.toString(intArray));

        swap(intArray, 1, 3);

        System.out.println("Swapped Integer array: " + java.util.Arrays.toString(intArray));
    }
}
```

```

// Demonstrate with an array of Strings
String[] stringArray = {"apple", "banana", "cherry", "date"};

System.out.println("Original String array: " + java.util.Arrays.toString(stringArray));

swap(stringArray, 0, 2);

System.out.println("Swapped String array: " + java.util.Arrays.toString(stringArray));


// Demonstrate with an array of Doubles
Double[] doubleArray = {1.1, 2.2, 3.3, 4.4};

System.out.println("Original Double array: " + java.util.Arrays.toString(doubleArray));

swap(doubleArray, 2, 3);

System.out.println("Swapped Double array: " + java.util.Arrays.toString(doubleArray));
}
}

```

Explanation:

1. Generic Method Definition:

- The method swap is defined as a static generic method with the syntax `<T> void swap(T[] array, int index1, int index2)`.
- The `<T>` before the return type void specifies that this is a generic method and T is the type parameter.

2. Parameter Check:

- The method checks if the array is null or if the indices are out of bounds. If so, it throws an `IllegalArgumentException`.

3. Swapping Elements:

- The elements at index1 and index2 are swapped using a temporary variable temp.

4. Main Method:

- The main method demonstrates the usage of the swap method with arrays of different types: Integer, String, and Double.

- The arrays are printed before and after the swap to show the effect of the method.