



Mobile Application Development Lab

Name:

Qurat-ul-ain Fatima

Enrollment:

01-134222-121

Class:

BS (CS) 6B

Lab 2

Date: 05/03/2025

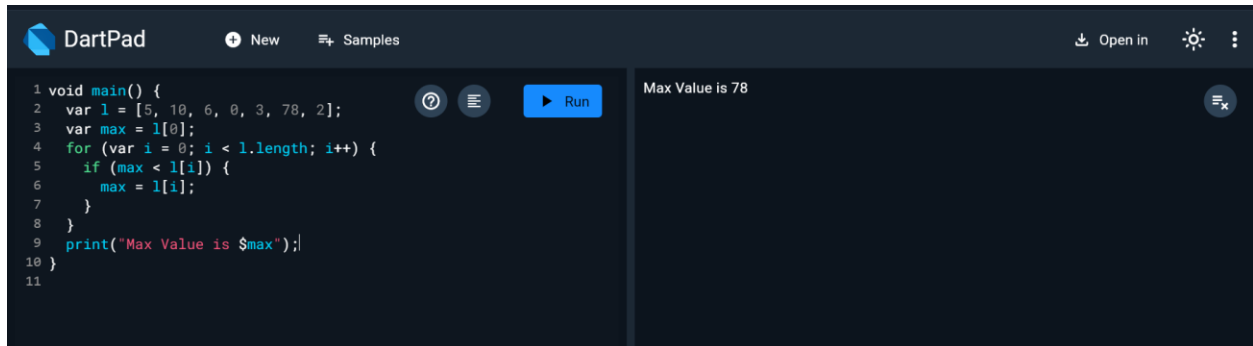
Submitted to:

Sir Mohsin Javed

TASK 1:

Find the largest number in a given list.

Solution:



The screenshot shows the DartPad interface. On the left, the code editor contains a Dart program that finds the maximum value in a list. On the right, the output console shows the result of the program.

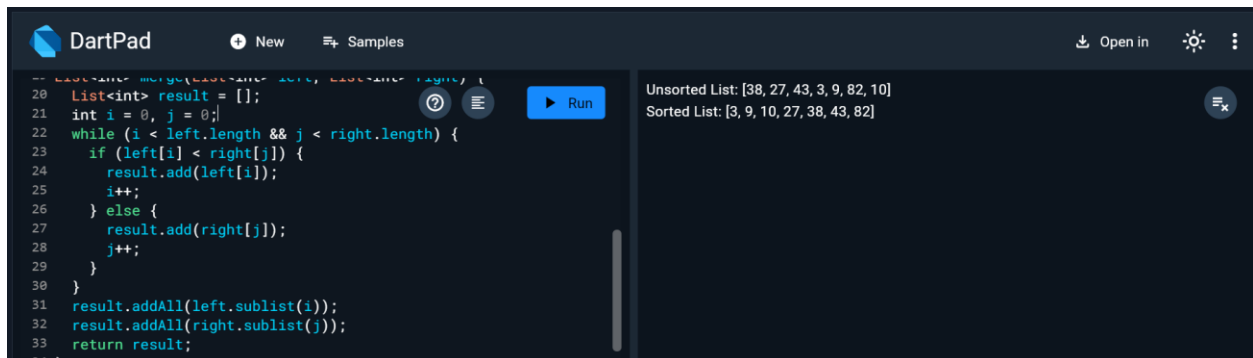
```
1 void main() {  
2   var l = [5, 10, 6, 0, 3, 78, 2];  
3   var max = l[0];  
4   for (var i = 0; i < l.length; i++) {  
5     if (max < l[i]) {  
6       max = l[i];  
7     }  
8   }  
9   print("Max Value is $max");  
10 }  
11
```

Max Value is 78

TASK 2:

Use merge sort to sort a List.

Solution:



The screenshot shows the DartPad interface. On the left, the code editor contains a Dart program that implements the merge sort algorithm. On the right, the output console shows the unsorted and sorted lists.

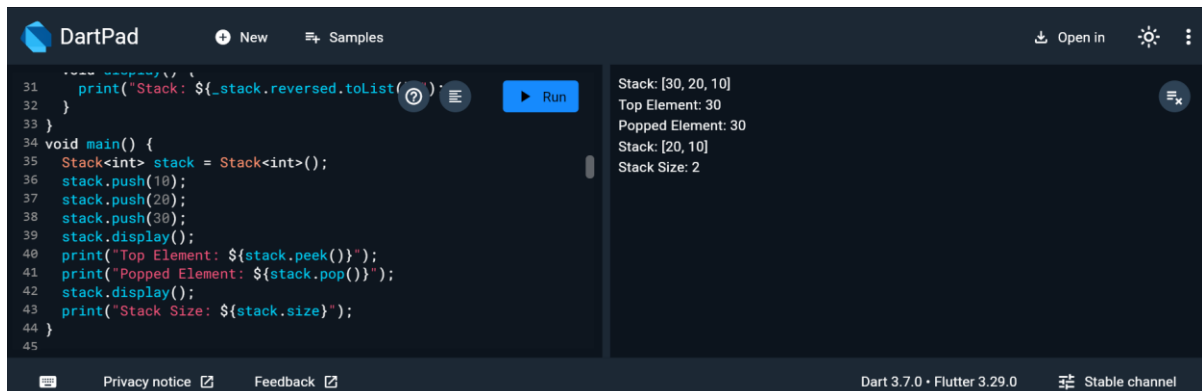
```
20 List<int> result = [];  
21 int i = 0, j = 0;  
22 while (i < left.length && j < right.length) {  
23   if (left[i] < right[j]) {  
24     result.add(left[i]);  
25     i++;  
26   } else {  
27     result.add(right[j]);  
28     j++;  
29   }  
30 }  
31 result.addAll(left.sublist(i));  
32 result.addAll(right.sublist(j));  
33 return result;  
34 }
```

Unsorted List: [38, 27, 43, 3, 9, 82, 10]
Sorted List: [3, 9, 10, 27, 38, 43, 82]

Task 3:

Implement a Stack from Scratch.

Solution:



The screenshot shows the DartPad interface. On the left, the code editor contains a Dart program that implements a stack using a list. On the right, the output console shows the state of the stack after several operations.

```
31 print("Stack: ${_stack.reversed.toList()}");  
32 }  
33 }  
34 void main() {  
35   Stack<int> stack = Stack<int>();  
36   stack.push(10);  
37   stack.push(20);  
38   stack.push(30);  
39   stack.display();  
40   print("Top Element: ${stack.peek()}");  
41   print("Popped Element: ${stack.pop()}");  
42   stack.display();  
43   print("Stack Size: ${stack.size}");  
44 }  
45
```

Stack: [30, 20, 10]
Top Element: 30
Popped Element: 30
Stack: [20, 10]
Stack Size: 2