

FRANSISKA WIDYA KRISANTI

NIM: 2341760101

KELAS / NO. ABSEN: SIB 3B / 16

## **PEMROGRAMAN MOBILE**

### **DART**

Assignment - Tugas

Soal:

- Create a leetcode account and try to finish the problem set using dart

language (choose one):

- Easy: 5 problems
  - Medium: 2 problems
  - Hard: 1 problem
- 
- Buat akun leetcode dan coba selesaikan rangkaian soal menggunakan bahasa dart (pilih salah satu):
- Mudah: 5 soal
  - Sedang: 2 soal
  - Sulit: 1 soal

Jawaban:

Memilih soal yang mudah yaitu 5 soal

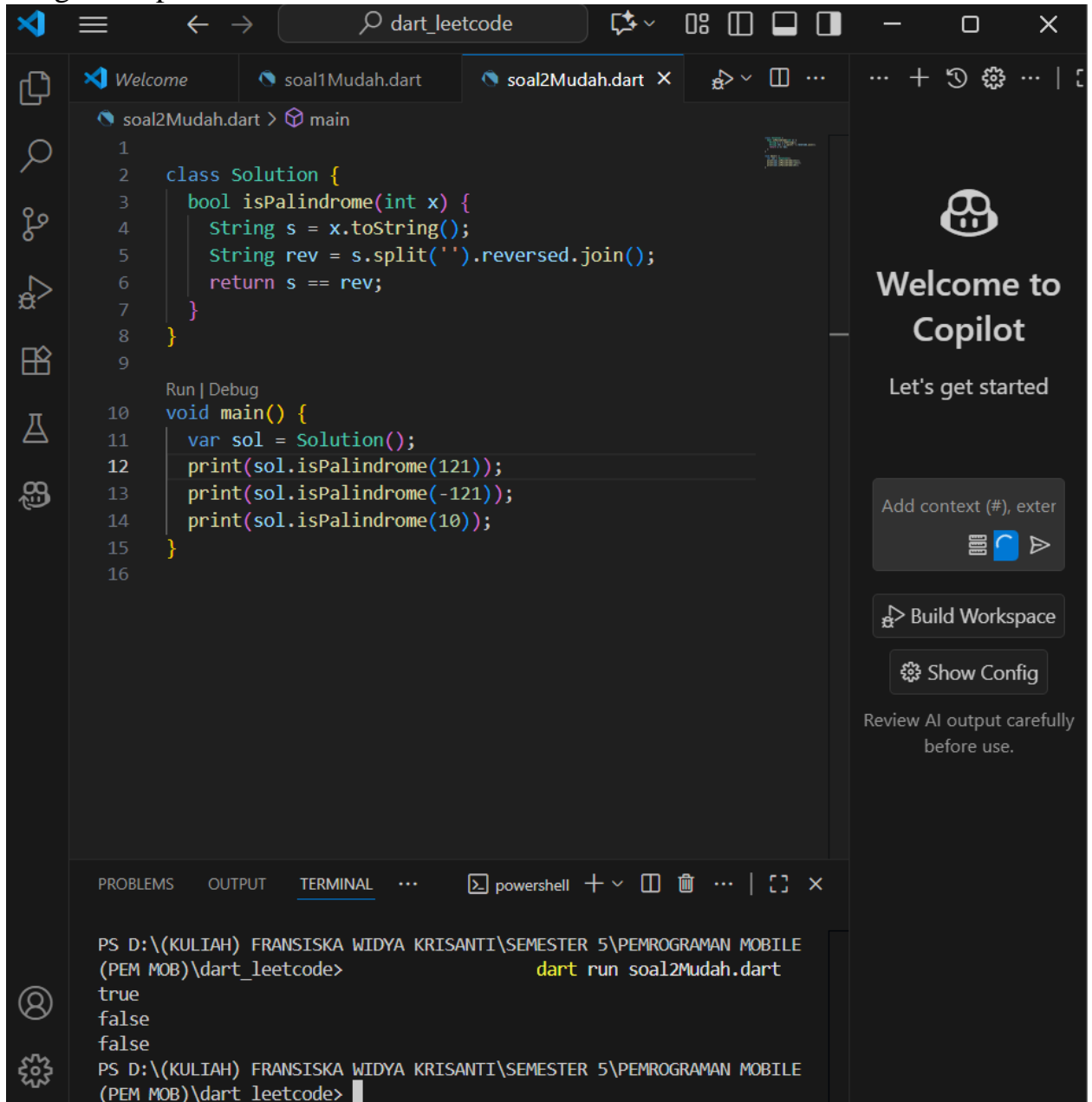
### 1. Penggunaan two sum

The image shows a screenshot of the Visual Studio Code (VS Code) editor interface. The main editor window displays a Dart file named 'soal1Mudah.dart'. The code is as follows:

```
1
2 class Solution {
3     List<int> twoSum(List<int> nums, int target) {
4         Map<int, int> seen = {};
5         for (int i = 0; i < nums.length; i++) {
6             int complement = target - nums[i];
7             if (seen.containsKey(complement)) {
8                 return [seen[complement]!, i];
9             }
10            seen[nums[i]] = i;
11        }
12        return [];
13    }
14 }
15
16 void main() {
17     var sol = Solution();
18     print(sol.twoSum([2,7,11,15], 9));
19 }
20
```

Below the code, there is a 'Run | Debug' section with a 'Run' button. The output of the program is shown in the terminal at the bottom: 'PS D:\(KULIAH) FRANSISKA WIDYA KRISANTI\SEMESTER 5\PEMROGRAMAN MOBILE (PEM MOB)\dart\_leetcode> dart run soal1Mudah.dart [0, 1]'. The right sidebar of VS Code shows the 'Welcome to Copilot' message, along with buttons for 'Add context (#), enter', 'Build Workspace', and 'Show Config'. The terminal tab is active, showing the command prompt and the output of the program.

## 2. Penggunaan palindrome



```
dart_leetcode

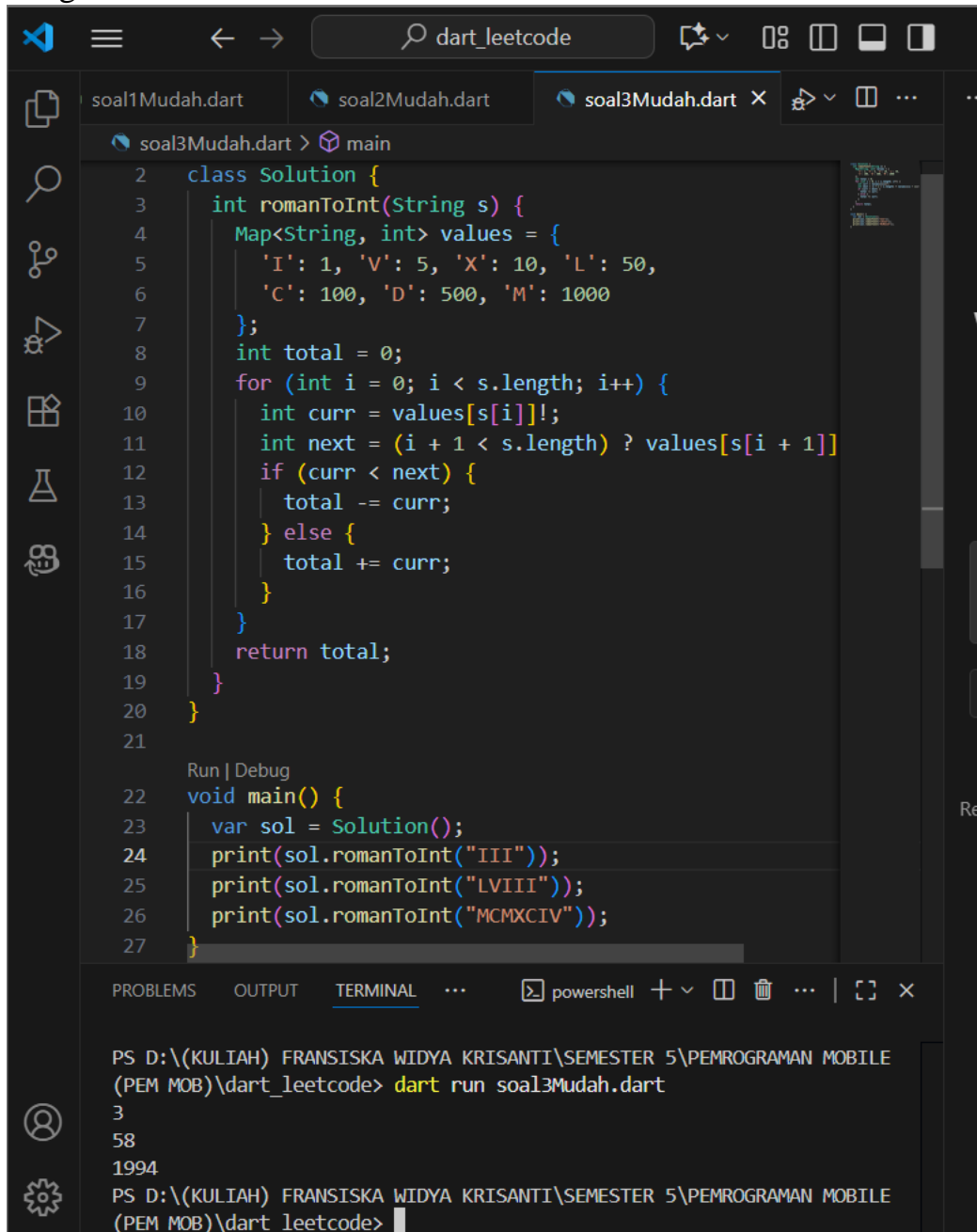
soal2Mudah.dart > main
1
2 class Solution {
3   bool isPalindrome(int x) {
4     String s = x.toString();
5     String rev = s.split('').reversed.join();
6     return s == rev;
7   }
8 }
9
Run | Debug
10 void main() {
11   var sol = Solution();
12   print(sol.isPalindrome(121));
13   print(sol.isPalindrome(-121));
14   print(sol.isPalindrome(10));
15 }
16
```

PROBLEMS OUTPUT TERMINAL ... powershell + - [ ] [ ] x

```
PS D:\(KULIAH) FRANSISKA WIDYA KRISANTI\SEMESTER 5\PEMROGRAMAN MOBILE (PEM MOB)\dart_leetcode> dart run soal2Mudah.dart
true
false
false
PS D:\(KULIAH) FRANSISKA WIDYA KRISANTI\SEMESTER 5\PEMROGRAMAN MOBILE (PEM MOB)\dart_leetcode>
```

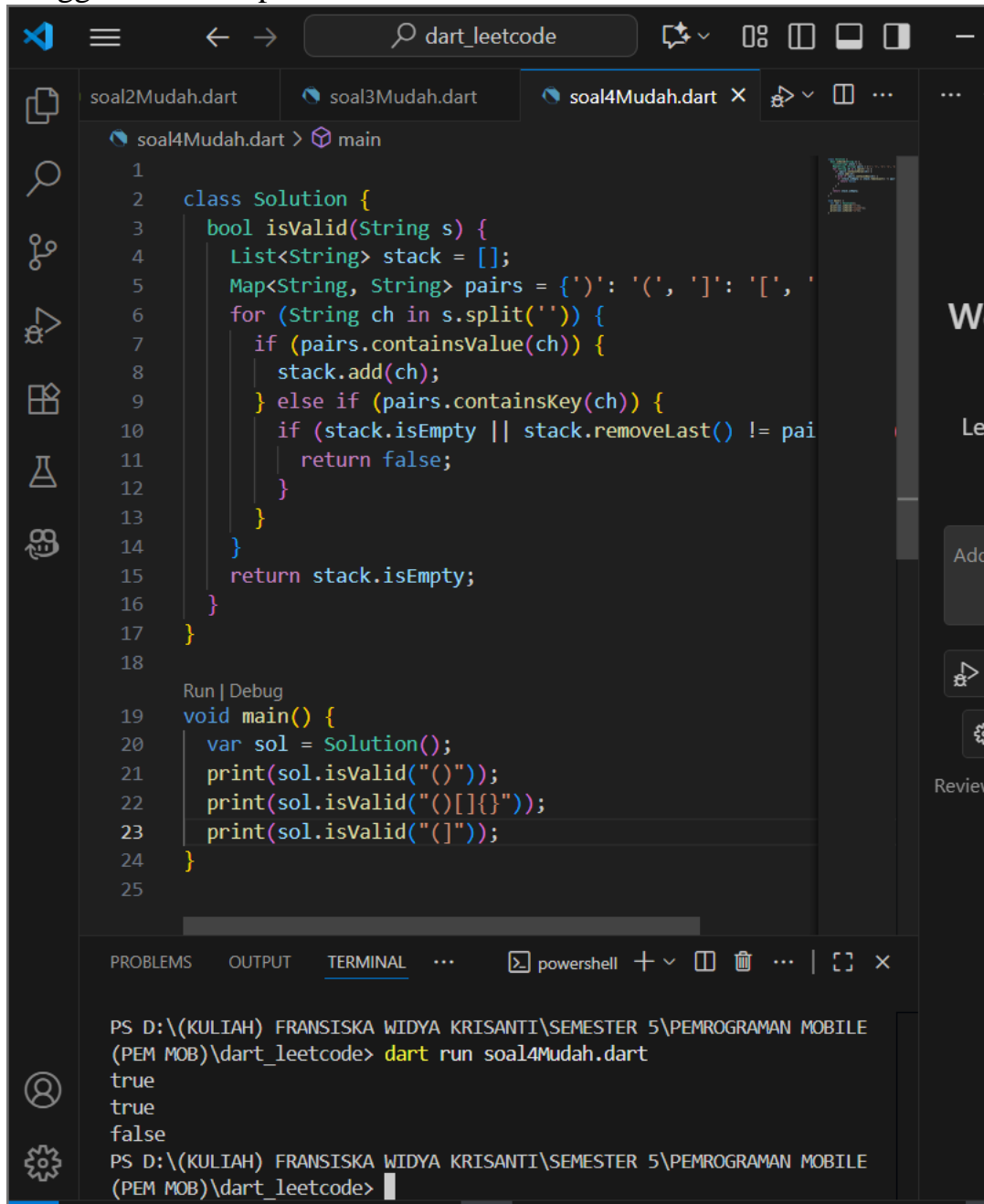
Welcome to Copilot  
Let's get started  
Add context (#), enter  
Build Workspace  
Show Config  
Review AI output carefully before use.

### 3. Penggunaan roman to int



```
soal1Mudah.dart  soal2Mudah.dart  soal3Mudah.dart x
soal3Mudah.dart > main
2  class Solution {
3    int romanToInt(String s) {
4      Map<String, int> values = {
5        'I': 1, 'V': 5, 'X': 10, 'L': 50,
6        'C': 100, 'D': 500, 'M': 1000
7      };
8      int total = 0;
9      for (int i = 0; i < s.length; i++) {
10         int curr = values[s[i]]!;
11         int next = (i + 1 < s.length) ? values[s[i + 1]]
12         if (curr < next) {
13           total -= curr;
14         } else {
15           total += curr;
16         }
17       }
18       return total;
19     }
20   }
21
22   Run | Debug
23   void main() {
24     var sol = Solution();
25     print(sol.romanToInt("III"));
26     print(sol.romanToInt("LVIII"));
27     print(sol.romanToInt("MCMXCIV"));
28   }
29
30   PROBLEMS  OUTPUT  TERMINAL  ...  powershell + -  |  x
31
32   PS D:\(KULIAH) FRANSISKA WIDYA KRISANTI\SEMESTER 5\PEMROGRAMAN MOBILE
33   (PEM MOB)\dart_leetcode> dart run soal3Mudah.dart
34   3
35   58
36   1994
37   PS D:\(KULIAH) FRANSISKA WIDYA KRISANTI\SEMESTER 5\PEMROGRAMAN MOBILE
38   (PEM MOB)\dart_leetcode>
```

#### 4. Penggunaan valid parenthesis



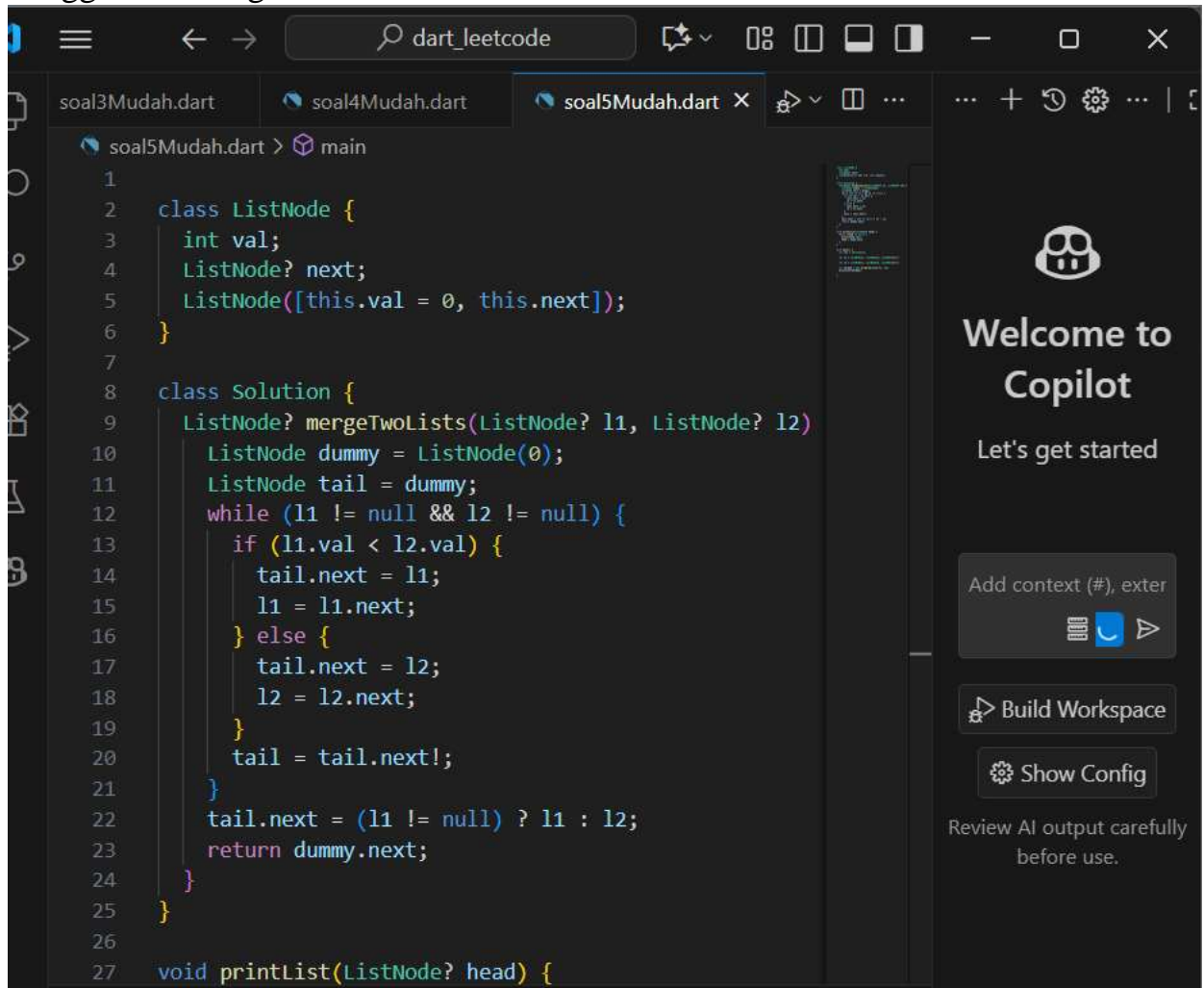
The screenshot shows an IDE with three tabs: `soal2Mudah.dart`, `soal3Mudah.dart`, and `soal4Mudah.dart`. The active tab is `soal4Mudah.dart`, which contains the following Dart code:

```
1
2 class Solution {
3   bool isValid(String s) {
4     List<String> stack = [];
5     Map<String, String> pairs = {'(': '(', ')': ')', '[': '[', ']' : ']' };
6     for (String ch in s.split('')) {
7       if (pairs.containsKey(ch)) {
8         stack.add(ch);
9       } else if (pairs.containsValue(ch)) {
10        if (stack.isEmpty || stack.removeLast() != pairs[ch]) {
11          return false;
12        }
13      }
14    }
15    return stack.isEmpty;
16  }
17 }
18
19 void main() {
20   var sol = Solution();
21   print(sol.isValid("()"));
22   print(sol.isValid("()[]{}"));
23   print(sol.isValid("(]"));
24 }
25
```

Below the code editor, the `Run | Debug` button is visible. The `TERMINAL` tab is active, showing the command `dart run soal4Mudah.dart` and its output:

```
PS D:\(KULIAH) FRANSISKA WIDYA KRISANTI\SEMESTER 5\PEMROGRAMAN MOBILE
(PEM MOB)\dart_leetcode> dart run soal4Mudah.dart
true
true
false
PS D:\(KULIAH) FRANSISKA WIDYA KRISANTI\SEMESTER 5\PEMROGRAMAN MOBILE
(PEM MOB)\dart_leetcode>
```

## 5. Penggunaan merge two lists



```
1
2 class ListNode {
3   int val;
4   ListNode? next;
5   ListNode([this.val = 0, this.next]);
6 }
7
8 class Solution {
9   ListNode? mergeTwoLists(ListNode? l1, ListNode? l2)
10    ListNode dummy = ListNode(0);
11    ListNode tail = dummy;
12    while (l1 != null && l2 != null) {
13      if (l1.val < l2.val) {
14        tail.next = l1;
15        l1 = l1.next;
16      } else {
17        tail.next = l2;
18        l2 = l2.next;
19      }
20      tail = tail.next!;
21    }
22    tail.next = (l1 != null) ? l1 : l2;
23    return dummy.next;
24  }
25 }
26
27 void printList(ListNode? head) {
```

soal3Mudah.dart   soal4Mudah.dart   **soal5Mudah.dart** X

soal5Mudah.dart > main

Welcome to Copilot

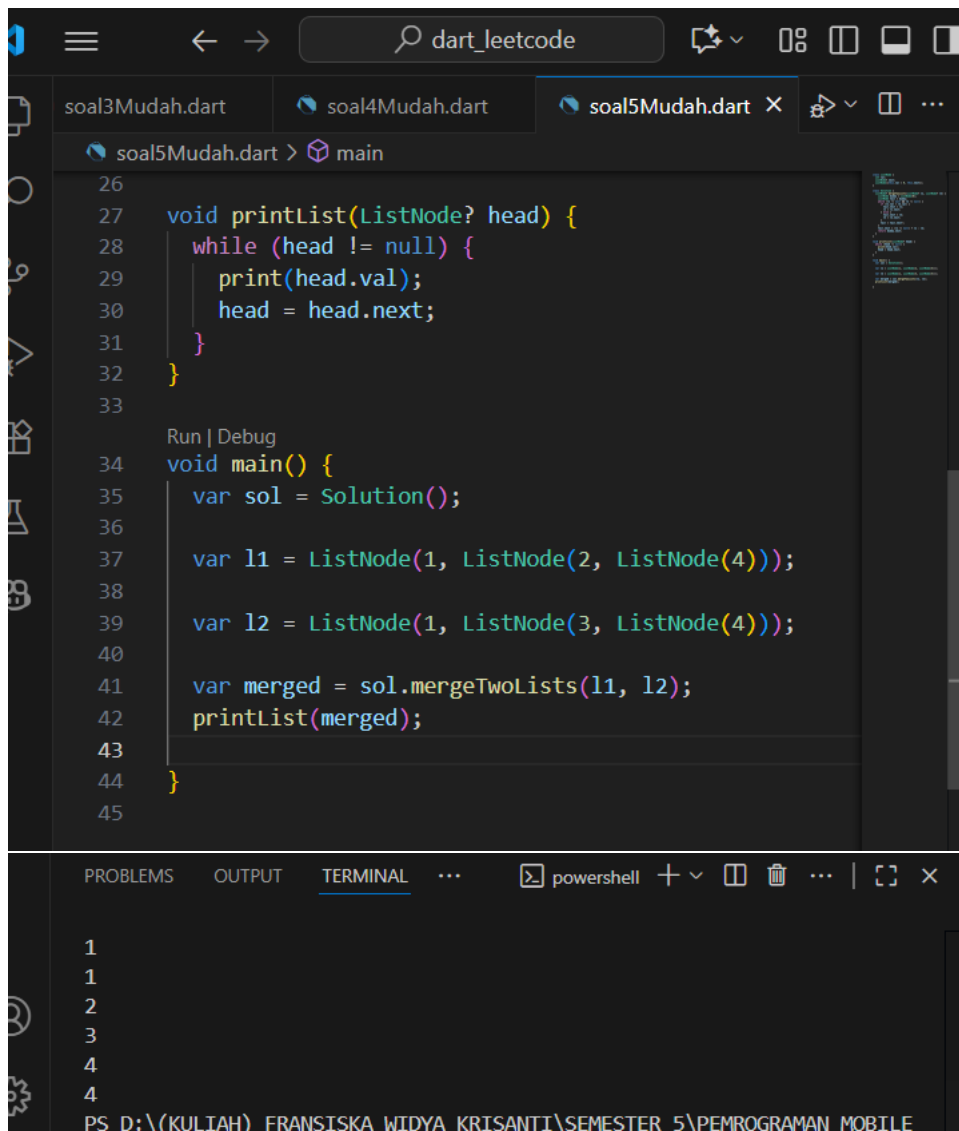
Let's get started

Add context (#), enter

Build Workspace

Show Config

Review AI output carefully before use.



The image shows a screenshot of an IDE with the following components:

- Search Bar:** Contains the text "dart\_leetcode".
- File Explorer:** Shows three files: "soal3Mudah.dart", "soal4Mudah.dart", and "soal5Mudah.dart".
- Code Editor:** Displays the code for "soal5Mudah.dart" with the following content:

```
26
27 void printList(ListNode? head) {
28     while (head != null) {
29         print(head.val);
30         head = head.next;
31     }
32 }
33
34 void main() {
35     var sol = Solution();
36
37     var l1 = ListNode(1, ListNode(2, ListNode(4)));
38
39     var l2 = ListNode(1, ListNode(3, ListNode(4)));
40
41     var merged = sol.mergeTwoLists(l1, l2);
42     printList(merged);
43 }
44
45
```
- Terminal:** Shows the output of the program:

```
1
1
2
3
4
4
```
- Terminal Path:** The path shown is "PS D:\(KULTAH) FRANSISKA WIDYA KRISANTI\SEMESTER 5\PEMROGRAMAN MOBILE".