

# Tugas 3 Dart - Pemrograman perangkat bergerak

☀ Status	Done
📌 Type	Task
📅 Due Date	@October 6, 2024 0:09
🕒 Last edited time	@October 6, 2024 6:37 PM

## Soal 1

```
import 'dart:math';

void main() {
  int a = 3, b = 2;

  // Membuat dan menampilkan matriks
  List<List<int>> matrix = createMatrix(a, b);

  // Menampilkan matriks
  print('Matriks $a x $b:');
  printMatrix(matrix);

  // Menampilkan hasil transpose
  print('Hasil transpose:');
  printMatrix(transpose(matrix));
}

// Fungsi untuk membuat matriks
List<List<int>> createMatrix(int a, int b) =>
  List.generate(a, (_) => List.generate(b, (_) => Random
    ().nextInt(10)));

void printMatrix(List<List<int>> matrix) {
  matrix.forEach((row) => print(row.join(' ')));
}
```

```
}
```

```
List<List<int>> transpose(List<List<int>> matrix) => List.generate(  
    matrix[0].length, (i) => List.generate(matrix.length,  
    (j) => matrix[j][i]));
```

## Output

```
Matriks 3 x 2:
```

```
7 4
```

```
4 7
```

```
7 7
```

```
Hasil transpose:
```

```
7 4 7
```

```
4 7 7
```

```
PS C:\Windows\System32\WindowsPowerShell\v1.0>
```

## Soal 2

```
import 'dart:io';
```

```
void main() {
```

```
    // Membuat list 2 dimensi
```

```
    List<List<int>> list2D = createList2D();
```

```
    // Menampilkan isi list
```

```
    displayList(list2D);
```

```
    // Meminta input dari pengguna untuk mencari bilangan
```

```
    int target = getUserInput();
```

```
    // Mencari nilai target dalam list 2 dimensi dan menampilkan hasil
```

```
    List<String> positions = findInList(list2D, target);
```

```
    displayResults(positions, target);
```

```

}

// Fungsi untuk membuat list 2 dimensi
List<List<int>> createList2D() {
    return [
        // Baris 1: 3 bilangan kelipatan 5, mulai dari 5
        [for (int i = 1; i <= 3; i++) i * 5],

        // Baris 2: 4 bilangan genap, mulai dari 2
        [for (int i = 1; i <= 4; i++) i * 2],

        // Baris 3: 5 bilangan kuadrat dari bilangan asli, mula
        i dari 1
        [for (int i = 1; i <= 5; i++) i * i],

        // Baris 4: 6 bilangan asli berurutan, mulai dari 3
        [for (int i = 3; i < 3 + 6; i++) i]
    ];
}

// Fungsi untuk menampilkan isi list
void displayList(List<List<int>> list2D) {
    print("Isi List:");
    for (List<int> row in list2D) {
        print(row.join(" "));
    }
}

// Fungsi untuk meminta input dari pengguna
int getUserInput() {
    stdout.write("\nBilangan yang dicari: ");
    return int.parse(stdin.readLineSync());
}

// Fungsi untuk mencari nilai dalam list 2 dimensi
List<String> findInList(List<List<int>> list2D, int target)
{
    List<String> result = [];

```

```

    for (int i = 0; i < list2D.length; i++) {
        for (int j = 0; j < list2D[i].length; j++) {
            if (list2D[i][j] == target) {
                // Menyimpan posisi baris dan kolom (baris dan kolom
                // dimulai dari 1)
                result.add("baris ${i + 1} kolom ${j + 1}");
            }
        }
    }

    return result;
}

// Fungsi untuk menampilkan hasil pencarian
void displayResults(List<String> positions, int target) {
    if (positions.isEmpty) {
        print("\nBilangan $target tidak ditemukan dalam list.");
    } else {
        print("\n$target berada di:");
        for (String pos in positions) {
            print(pos);
        }
    }
}

```

## Output

5] Minggu ke 5 (Soal 2.dart)

Isi List:

5 10 15

2 4 6 8

1 4 9 16 25

3 4 5 6 7 8

Bilangan yang dicari: 16

16 berada di:

baris 3 kolom 4

PS C:\Windows\System32\WindowsPowerShell\v1.0>

### Soal 3

```
import 'dart:io';

void main() {
  // Meminta input dari pengguna
  stdout.write("Bilangan 1: ");
  int num1 = int.parse(stdin.readLineSync()!);

  stdout.write("Bilangan 2: ");
  int num2 = int.parse(stdin.readLineSync()!);

  // Menghitung KPK
  int kpk = calculateLCM(num1, num2);

  // Menampilkan hasil
  print("KPK $num1 dan $num2 = $kpk");
}

// Fungsi untuk menghitung KPK
int calculateLCM(int a, int b) {
  return (a * b) ~/ calculateGCD(a, b);
}
```

```
// Fungsi untuk menghitung FPB menggunakan algoritma Euclidean
int calculateGCD(int a, int b) {
    while (b != 0) {
        int temp = b;
        b = a % b;
        a = temp;
    }
    return a;
}
```

### Output

```
3] Minggu ke-3\soal-3.dart
Bilangan 1: 33
Bilangan 2: 90
KPK 33 dan 90 = 990
PS C:\Windows\System32\WindowsPowerShell\v1.0>
```

<https://github.com/DeoFaradyS/Pemrograman-Perangkat-Bergerak.git>