

Nama : Bintang Harida Ramadhan

NIM : L0122034

Kelas : A

TUGAS PRAKTIKUM WEEK 1

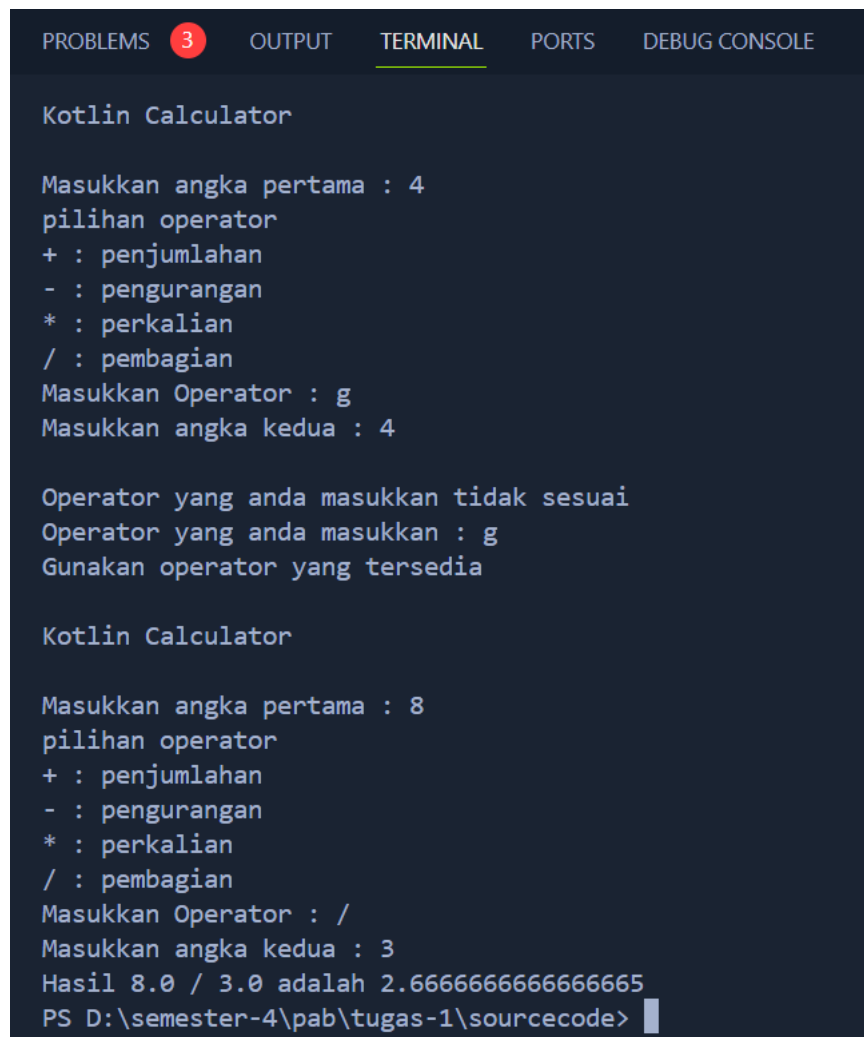
A. Screenshot Sourcecode

```
main.kt 3 x
tugas-1 > sourcecode > main.kt > calculator
1 // Nama : Bintang Harida Ramadhan
2 // NIM : L0122034
3
4 fun calculator()
5 {
6     var firstNumber : Double? = null
7     var secondNumber : Double? = null
8     var result : Double? = null
9     var operator : Char? = null
10
11     do
12     {
13         var invalidOperator : Boolean = false
14         println("Kotlin Calculator\n")
15         print("Masukkan angka pertama : ")
16         firstNumber = readLine()?.toDoubleOrNull()
17
18         println("pilihan operator")
19         println("+ : penjumlahan")
20         println("- : pengurangan")
21         println("* : perkalian")
22         println("/ : pembagian")
23
24         print("Masukkan Operator : ")
25         operator = readLine()?.firstOrNull()
26         print("Masukkan angka kedua : ")
27         secondNumber = readLine()?.toDoubleOrNull()
28
29         if (operator == '+') result = (firstNumber ?: 0.0) + (secondNumber ?: 0.0)
30         else if (operator == '-') result = (firstNumber ?: 0.0) - (secondNumber ?: 0.0)
31         else if (operator == '*') result = (firstNumber ?: 0.0) * (secondNumber ?: 0.0)
32         else if (operator == '/') result = (firstNumber ?: 0.0) / (secondNumber ?: 0.0)
33         else {
34             println("\nOperator yang anda masukkan tidak sesuai")
35             println("Operator yang anda masukkan : $operator")
36             println("Gunakan operator yang tersedia\n")
37             invalidOperator = true
38         }
39     } while (invalidOperator)
40     val textResult : String = "Hasil $firstNumber $operator $secondNumber adalah $result"
41     println(textResult)
42 }
43
44 fun main() {
45     calculator()
46 }
```

B. Implementasi Materi

1. Function “calculator” digunakan untuk menjalankan logika operasi kalkulator
2. Data types & variables untuk menyimpan operasi kalkulator
3. Nullable types agar variable dapat menyimpan inputan null dari user
4. Safe calls agar memungkinkan menerima inputan null dari user (pada pemanggilan method `readLine()`)
5. If expression untuk percabangan kondisi operator pada kalkulator
6. Elvis operators untuk menangani inputan null dari user, jika inputan null maka nilai variable akan diganti dengan nol (0)
7. String template untuk memudahkan dalam menyisipkan nilai dari variable pada string

C. Screenshot Terminal



```
PROBLEMS 3 OUTPUT TERMINAL PORTS DEBUG CONSOLE

Kotlin Calculator

Masukkan angka pertama : 4
pilihan operator
+ : penjumlahan
- : pengurangan
* : perkalian
/ : pembagian
Masukkan Operator : g
Masukkan angka kedua : 4

Operator yang anda masukkan tidak sesuai
Operator yang anda masukkan : g
Gunakan operator yang tersedia

Kotlin Calculator

Masukkan angka pertama : 8
pilihan operator
+ : penjumlahan
- : pengurangan
* : perkalian
/ : pembagian
Masukkan Operator : /
Masukkan angka kedua : 3
Hasil 8.0 / 3.0 adalah 2.6666666666666665
PS D:\semester-4\pab\tugas-1\sourcecode>
```

```

Kotlin Calculator

Masukkan angka pertama : 6
pilihan operator
+ : penjumlahan
- : pengurangan
* : perkalian
/ : pembagian
Masukkan Operator : -
Masukkan angka kedua :
Hasil 6.0 - 0.0 adalah 6.0
lanjut menggunakan kalkulator? y/n :
y
Kotlin Calculator

Masukkan angka pertama : 5
pilihan operator
+ : penjumlahan
- : pengurangan
* : perkalian
/ : pembagian
Masukkan Operator : /
Masukkan angka kedua : 0
pembagian 0 tidak terdefinisi
Kotlin Calculator

Masukkan angka pertama : 4
pilihan operator
+ : penjumlahan
- : pengurangan
* : perkalian
/ : pembagian
Masukkan Operator : +
Masukkan angka kedua : 63
Hasil 4.0 + 63.0 adalah 67.0
lanjut menggunakan kalkulator? y/n :
n
PS D:\semester-4\pab\tugas-1\sourcecode>

```

D. Penjelasan Program

Setelah dieksekusi program akan meminta inputan angka pertama dilanjutkan dengan meminta inputan operator dan angka kedua. Pada percobaan diatas telah dicoba dengan beberapa kondisi sebagai berikut :

- 1) Salah dalam menginputkan operator, maka kalkulator akan kembali mengeksekusi program dari awal
- 2) Inputan null, maka program akan menjalankan elvis dan memberi nilai 0 pada variable yang nilainya null
- 3) Pembagian dengan nol, maka program akan memberi tahu jika pembagian dengan nol hasilnya adalah tidak terdefinisi

E. Kesimpulan

Penggunaan nullable type, safe calls, dan elvis operators akan sangat berguna jika program memungkinkan memiliki variable yang bernilai null. Dengan ini, program yang memiliki variable bernilai null akan tetap dapat berjalan dan terhindar dari NPE (NullPointerException). Pada materi String template, teknik ini memudahkan untuk membuat variable string mmaupun menampilkan output menggunakan println(). Dengan string template programmer dapat dengan mudah

mengkombinasikan antara string regular yang diapit dengan tanda petik (") serta nilai dari variable.