

# **LAPORAN PRAKTIKUM**

## **Aplikasi Mobile Framework KSI 6A**

### **“P4 – Fungsi dan Eksepsi”**

---



**Oleh :**

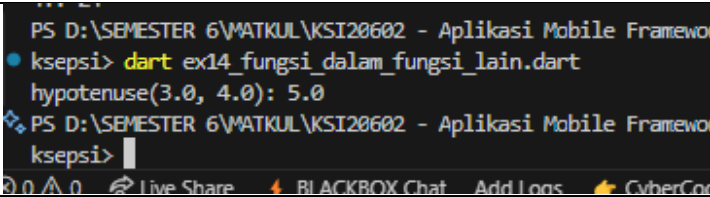
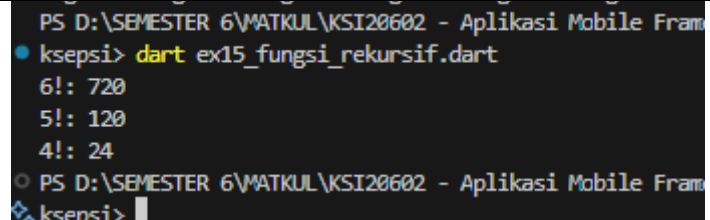
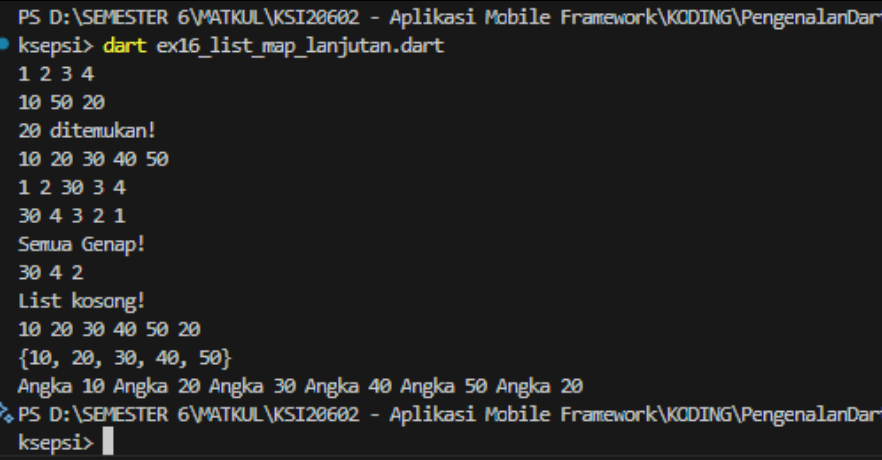
**Fazri Nugraha**  
**(6404221108)**

**Dosen Pengampu:**

**Muhammad Ikhsan Wibowo, M.Kom**

**PROGRAM STUDI D4-KEAMANAN SISTEM INFORMASI**  
**JURUSAN TEKNIK INFORMATIKA**  
**POLITEKNIK NEGERI BENGKALIS**  
**2025**

No	P4 – Fungsi dan Eksepsi
1.	<p><b>ex11_fungsi_lambda.dart</b></p> <p>Berikut ini contoh penggunaan fungsi anonim (anonymous function) dalam Dart. Saya mendeklarasikan variabel fungsiLambda yang menyimpan fungsi tanpa nama, dan saat dipanggil akan mencetak teks "Demo fungsi tanpa nama (anonymous function)". Fungsi ini langsung dijalankan dengan fungsiLambda();. Selanjutnya, saya membuat fungsi anonim lain dalam variabel add, yang menerima dua parameter bertipe double dan mengembalikan hasil penjumlahannya. Fungsi ini dipanggil dengan argumen 34 dan 18, lalu hasilnya disimpan dalam variabel result dan ditampilkan ke layar sebagai "Nilai variabel result: 52.0".</p> 
2.	<p><b>ex12_fungsi_sebagai_parameter.dart</b></p> <p>Ini adalah contoh penerapan konsep fungsi sebagai parameter di Dart. Fungsi calc menerima dua nilai bertipe double dan satu parameter fungsi operation untuk melakukan operasi matematika. Di dalam fungsi main, calc dipanggil empat kali dengan fungsi anonim yang langsung didefinisikan. Operasi pertama adalah penjumlahan (hasilnya 15.0 disimpan di add), lalu perkalian (36.0 ke multiply), pembagian (4.0 ke div), dan modulus atau sisa bagi (0.0 ke mod). Semua hasil operasi kemudian dicetak ke layar dengan format yang menjelaskan masing-masing.</p> 
3.	<p><b>ex13_fungsi_callback.dart</b></p> <p>Contoh ini menunjukkan penggunaan fungsi sebagai parameter dengan pendekatan yang lebih terstruktur. Fungsi calc menerima dua nilai double dan satu parameter fungsi untuk menentukan operasi matematika. Tiga fungsi terpisah didefinisikan secara ringkas dengan ekspresi lambda: add untuk penjumlahan, multiply untuk perkalian, dan div untuk pembagian. Di dalam main, ketiganya dipanggil lewat calc dengan argumen 12.0 dan 3.0, menghasilkan masing-masing: 15.0, 36.0, dan 4.0. Hasilnya kemudian dicetak dengan format yang menunjukkan jenis operasi yang dilakukan.</p> 
4.	<p><b>ex14_fungsi_dalam_fungsi_lain.dart</b></p> <p>Kode ini menghitung panjang sisi miring segitiga siku-siku menggunakan fungsi hypotenuse, yang menerima dua parameter double sebagai panjang sisi siku-siku. Di dalamnya ada fungsi lokal square untuk menghitung kuadrat suatu nilai. Hasil kuadrat dari kedua sisi dijumlahkan,</p>

	<p>lalu diakarkan dengan sqrt dari pustaka dart:math. Di fungsi main, hypotenuse(3.0, 4.0) dipanggil dan menghasilkan 5.0, karena <math>\sqrt{(3^2 + 4^2)} = \sqrt{25} = 5.0</math>. Hasil ini kemudian dicetak ke layar.</p>
	
	<b>ex15_fungsi_rekursif.dart</b>
5.	<p>Kode ini menggunakan fungsi rekursif untuk menghitung faktorial dari bilangan bulat n. Jika n sama dengan 0, maka hasilnya 1 (karena <math>0! = 1</math>). Jika lebih dari 0, fungsi akan memanggil dirinya sendiri dengan n - 1, lalu dikalikan dengan n. Di dalam main, fungsi factorial dipanggil dengan nilai 6, 5, dan 4, yang masing-masing menghasilkan 720, 120, dan 24. Hasilnya kemudian ditampilkan ke layar.</p>
	
	<b>ex16_list_map_lanjutan.dart</b>
6.	<p>Kode ini menunjukkan berbagai manipulasi pada struktur data List di Dart. Dimulai dari list awal [1, 2, 3, 4], lalu ditambahkan elemen baru dengan add, addAll, serta disisipkan dengan insert dan insertAll.</p> <p>Beberapa elemen dihapus menggunakan remove, removeAt, removeLast, removeRange, dan removeWhere berdasarkan kondisi tertentu. Program juga mengecek keberadaan elemen dengan contains.</p> <p>Selanjutnya, list dimodifikasi dengan sublist, diurutkan menurun dengan sort, dan elemen ganjil dihapus. Kemudian dicek apakah semua elemen genap menggunakan every, dikosongkan dengan clear, dan diperiksa lagi dengan isEmpty.</p> <p>Terakhir, list dikonversi ke Set untuk menghapus duplikat, lalu diubah menjadi List&lt;String&gt; dengan map, dengan menambahkan teks ke setiap elemen.</p>
	
	<b>ex17_throw.dart</b>
7.	<p>Kode di atas merupakan program Dart sederhana yang melakukan pembagian bilangan bulat menggunakan operator ~/ . Program meminta input dua bilangan dari pengguna, lalu</p>

	<p>memeriksa apakah nilai pembaginya (b) adalah nol. Jika <math>b = 0</math>, maka akan muncul error <code>IntegerDivisionByZeroException</code> karena pembagian dengan nol tidak diperbolehkan. Jika nilainya valid, program akan menampilkan hasil pembagian bulat dari <math>a \sim b</math>. Misalnya, saat saya masukkan <math>a = 10</math> dan <math>b = 0</math>, program langsung menampilkan error. Program ini juga memperlihatkan bagaimana cara menangani kesalahan (exception) secara eksplisit dalam operasi aritmatika di Dart.</p>
	
8.	<p><b>ex18_throw_exception.dart</b></p> <p>Gambar tersebut menunjukkan program Dart yang melakukan pembagian bilangan bulat menggunakan operator <code>~/</code>. Program meminta dua input dari pengguna (a dan b), lalu mengubahnya menjadi integer. Sebelum menghitung, program mengecek apakah nilai b adalah nol. Jika <math>b = 0</math>, maka akan muncul pesan error berupa exception dengan tulisan “SALAH: terdapat pembagian dengan nilai nol.” dan program berhenti. Jika nilainya valid, hasil pembagian <math>a \sim b</math> akan ditampilkan ke layar. Ketika saya masukkan nilai <math>a = 10</math> dan <math>b = 0</math>, program langsung menampilkan pesan error: Exception: SALAH: terdapat pembagian dengan nilai nol.</p> 
9.	<p><b>ex19_try_on.dart</b></p> <p>Program Dart ini melakukan pembagian bilangan bulat dengan penanganan error memakai blok try-on. Pengguna diminta masukkan dua angka, misalnya <math>a = 10</math> dan <math>b = 0</math>. Karena pembagian dengan nol tidak boleh, program akan menangkap error <code>IntegerDivisionByZeroException</code>. Blok try berisi operasi <math>a \sim b</math>, dan jika ada kesalahan, bagian on akan menampilkan pesan bahwa terjadi pembagian dengan nol, lalu mencetak jenis kesalahan serta stack trace.</p>

	<pre> PS D:\SEMESTER 6\MATKUL\KSI20602 - Aplikasi Mobile Framework\KODING\PengenalanDart\P4_FungsiEksepsi&gt; dart ex19_try_on.dart Masukkan nilai a: 10 Masukkan nilai b: 0 SALAH: terjadi pembagian dengan nilai nol. Jenis eksepsi: IntegerDivisionByZeroException Stacktrace: #0      int~/ (dart:core-patch/integers.dart:27:7) #1      main (file:///D:/SEMESTER%206/MATKUL/KSI20602%20-%20Aplikasi%20Mobile%20Framework/KODING/PengenalanDart/P4_FungsiEksepsi/ex19_try_on.dart:9:11) #2      _delayEntrypointInvocation.&lt;anonymous closure&gt; (dart:isolate-patch/isolate_patch.dart:313:33) #3      _RawReceivePort._handleMessage (dart:isolate-patch/isolate_patch.dart:194:12)  PS D:\SEMESTER 6\MATKUL\KSI20602 - Aplikasi Mobile Framework\KODING\PengenalanDart\P4_FungsiEksepsi&gt; ksepsi&gt; </pre>
10.	<p><b>ex20_try_catch.dart</b></p> <p>Program ini dibuat untuk menampilkan nama hari berdasarkan angka yang dimasukkan pengguna. List days berisi nama-nama hari dari Minggu sampai Sabtu. Setelah pengguna memasukkan angka, program mengambil data dari list dengan indeks index - 1. Kalau angka yang dimasukkan lebih dari 7, seperti contoh hari ke-9, maka akan muncul error karena datanya tidak ada di list. Untuk mencegah program berhenti tiba-tiba, saya gunakan blok try-catch, jadi kalau ada error seperti RangeError, akan ditampilkan pesan bahwa hari tersebut tidak tersedia, lengkap dengan jenis error dan stackTrace-nya.</p> <pre> PS D:\SEMESTER 6\MATKUL\KSI20602 - Aplikasi Mobile Framework\KODING\PengenalanDart\P4_FungsiEksepsi&gt; dart ex20_try_catch.dart Masukkan nomor hari: 9 SALAH: tidak ada hari ke-9. Jenis eksepsi: RangeError (length): Invalid value: Not in inclusive range 0..6: 8 Stacktrace: #0      List.[] (dart:core-patch/growable_array.dart) #1      main (file:///D:/SEMESTER%206/MATKUL/KSI20602%20-%20Aplikasi%20Mobile%20Framework/KODING/PengenalanDart/P4_FungsiEksepsi/ex20_try_catch.dart:16:40) #2      _delayEntrypointInvocation.&lt;anonymous closure&gt; (dart:isolate-patch/isolate_patch.dart:313:33) #3      _RawReceivePort._handleMessage (dart:isolate-patch/isolate_patch.dart:194:12)  PS D:\SEMESTER 6\MATKUL\KSI20602 - Aplikasi Mobile Framework\KODING\PengenalanDart\P4_FungsiEksepsi&gt; ksepsi&gt; </pre>
11.	<p><b>ex21_try_catch_finally.dart</b></p> <p>Program Dart ini melakukan pembagian dua bilangan bulat yang dimasukkan oleh pengguna. Operasi pembagian dilakukan dengan operator ~/ , yang menghasilkan hasil pembagian tanpa desimal. Penanganan error dilakukan dengan try-catch-finally. Jika nilai pembagi (b) adalah 0, maka akan muncul error IntegerDivisionByZeroException yang langsung ditangkap dan ditampilkan, lengkap dengan jenis error dan stackTrace. Apapun hasilnya—berhasil atau error—blok finally tetap dijalankan untuk menunjukkan bahwa proses sudah selesai. Contohnya, saat a = 12 dan b = 4, hasilnya adalah 12 ~/ 4 = 3, lalu muncul pesan “Bagian finalisasi”.</p> <p>Kalau b = 0, akan muncul error IntegerDivisionByZeroException dan ditangani di blok catch, lalu ditampilkan pesan error dan stackTrace. Terlepas dari error atau tidak, bagian finally tetap dijalankan dan akan menampilkan pesan “Bagian finalisasi”. Program ini menunjukkan cara menangani error dengan aman supaya tidak langsung crash saat runtime.</p>

		<pre> PS D:\SEMESTER 6\MATKUL\KSI20602 - Aplikasi Mobile ksepsi&gt; dart ex21_try_catch_finally.dart Masukkan nilai a: 12 Masukkan nilai b: 4 12 ~/ 4 = 3 Bagian finalisasi PS D:\SEMESTER 6\MATKUL\KSI20602 - Aplikasi Mobile ksepsi&gt; </pre>
12.	<p><b>ex22_multiple_exception.dart</b></p> <p>Program Dart ini menangani dua jenis kesalahan (multiple exception) saat pengguna memasukkan dua nilai untuk dibagi. Jika input bukan angka, program akan menangkap dan menampilkan pesan bahwa input bukan bilangan. Kalau pembaginya nol, program akan menangkap dan menampilkan pesan bahwa tidak bisa membagi dengan nol. Setelah itu, program selalu menampilkan “Selesai...” sebagai tanda bahwa proses sudah selesai dijalankan, baik ada error maupun tidak.</p> <p>Contoh:</p> <ul style="list-style-type: none"> <li>Jika input b = a12, muncul pesan: SALAH: nilai yang dimasukkan bukan bilangan.</li> <li>Jika input b = 0, muncul pesan: SALAH: terjadi pembagian dengan nilai nol.</li> </ul>	<pre> PS D:\SEMESTER 6\MATKUL\KSI20602 - Aplikasi Mobile Framework\KOD ksepsi&gt; dart ex22_multiple_exception.dart Masukkan nilai a: 10 Masukkan nilai b: a12 SALAH: nilai yang dimasukkan bukan bilangan. Selesai... PS D:\SEMESTER 6\MATKUL\KSI20602 - Aplikasi Mobile Framework\KOD ksepsi&gt; dart ex22_multiple_exception.dart Masukkan nilai a: 12 Masukkan nilai b: 0 SALAH: terjadi pembagian dengan nilai nol. Selesai... PS D:\SEMESTER 6\MATKUL\KSI20602 - Aplikasi Mobile Framework\KOD ksepsi&gt; </pre>
13.	<p><b>ex23_user_defined_exception.dart</b></p> <p>Program ini dibuat untuk meminta input bilangan non-negatif dari pengguna. Kalau pengguna masukan angka negatif, maka program bakal lempar error khusus (NegativeError) yang udah dibuat sendiri. Selain itu, kalau input yang dimasukkan bukan angka (contoh: “12f”), akan tertangkap sebagai FormatException.</p> <p>Contoh hasil program di gambar:</p> <ul style="list-style-type: none"> <li>Input -4 akan tampil pesan: SALAH: anda memasukkan nilai negatif.</li> <li>Input 12f akan tampil pesan: SALAH: nilai yang dimasukkan bukan bilangan.</li> </ul>	<pre> PS D:\SEMESTER 6\MATKUL\KSI20602 - Aplikasi Mobil ksepsi&gt; dart ex23_user_defined_exception.dart Masukkan bilangan non-negatif: -4 SALAH: anda memasukkan nilai negatif. PS D:\SEMESTER 6\MATKUL\KSI20602 - Aplikasi Mobil ksepsi&gt; </pre> <pre> PS D:\SEMESTER 6\MATKUL\KSI20602 - Aplikasi Mol ksepsi&gt; dart ex23_user_defined_exception.dart Masukkan bilangan non-negatif: 12f SALAH: nilai yang dimasukkan bukan bilangan. PS D:\SEMESTER 6\MATKUL\KSI20602 - Aplikasi Mol ksepsi&gt; </pre>

Link Repositori github : <https://github.com/Fazrinugraha/Pengenalan-Dart>