|  |  |  |
| --- | --- | --- |
| **Nama** | : | Abdullah Azzam |
| **Nim** | : | 2141720073 |
| **Kelas** | : | 3E\_D4\_TI |

Tulislah Jawaban Pada Kolom Yang tersedia di bawah ini;

**LEMBAR JAWABAN JOBSHEET-4**

|  |  |
| --- | --- |
| **Soal No** | **Jawaban** |
| 1 | Silakan selesaikan Praktikum 1 sampai 5, lalu dokumentasikan berupa screenshot hasil pekerjaan Anda beserta penjelasannya!   * Praktikum 1        * Praktikum 2        * Praktikum 3        * Praktikum 4        * Praktikum 5 |
| 2 | Jelaskan yang dimaksud Functions dalam bahasa Dart!  **Jawab**:  Dalam bahasa pemrograman Dart, "functions" (fungsi) adalah blok kode yang dapat digunakan untuk melakukan tugas-tugas tertentu atau operasi tertentu. Fungsi adalah salah satu konsep dasar dalam pemrograman dan memungkinkan Anda untuk mengelompokkan serangkaian pernyataan yang terkait ke dalam satu unit yang dapat digunakan ulang. |
| 3 | Jelaskan jenis-jenis parameter di Functions beserta contoh sintaksnya!  **Jawab**:   1. Positional Parameters (Parameter Posisi):   Parameter-posisi adalah jenis parameter yang nilainya dikirimkan ke fungsi berdasarkan urutan posisi. Mereka adalah jenis parameter yang paling umum digunakan. Untuk mendefinisikan parameter-posisi, Anda hanya perlu menyebutkannya dalam tanda kurung.     1. Named Parameters (Parameter Bernama):   Parameter bernama adalah parameter yang dikirimkan ke fungsi dengan menyebutkan namanya. Mereka memungkinkan Anda untuk mengirim parameter dalam urutan yang bebas, karena Anda menyebutkan nama parameter saat memanggil fungsi. Untuk mendefinisikan parameter bernama, Anda menggunakan kurung kurawal {} dalam definisi fungsi     1. Default Parameter Values (Nilai Parameter Default):   Anda dapat memberikan nilai default kepada parameter, sehingga jika parameter tidak diberikan saat memanggil fungsi, nilai default akan digunakan. Ini berguna untuk memberikan nilai default yang masuk akal kepada parameter yang opsional.     1. Required Parameters (Parameter Wajib):   Parameter wajib adalah parameter yang harus diberikan saat memanggil fungsi. Mereka diperkenalkan dalam Dart 2.13 ke atas dengan kata kunci required.     1. Rest Parameters (Parameter Sisa):   Rest parameter mengizinkan Anda untuk mengirim sejumlah parameter yang tidak terbatas ke dalam fungsi. Mereka diwakili oleh tanda titik tiga (...) diikuti dengan nama parameter. |
| 4 | Jelaskan maksud Functions sebagai first-class objects beserta contoh sintaknya!  **Jawab**:   1. Menyimpan Fungsi dalam Variabel:      1. Mengirimkan Fungsi sebagai Argumen:      1. Mengembalikan Fungsi dari Fungsi Lain: |
| 5 | Apa itu Anonymous Functions? Jelaskan dan berikan contohnya!  **Jawab**:  Anonymous functions, juga dikenal sebagai lambda functions atau fungsi tanpa nama, adalah jenis fungsi yang tidak memiliki nama yang terikat pada mereka. Mereka biasanya digunakan untuk menjalankan tugas-tugas sederhana secara langsung atau sebagai argumen fungsi.  Contoh:   1. Anonymous Function sebagai Argumen Fungsi:   Anda dapat menggunakan anonymous function sebagai argumen fungsi     1. Anonymous Function Sebagai Fungsi yang Dikembalikan:   Anda juga dapat mengembalikan anonymous function dari sebuah fungsi.     1. Anonymous Function dalam Operasi List:   Anonymous function sering digunakan dalam operasi-operasi list seperti map, where, dan reduce. |
| 6 | Jelaskan perbedaan Lexical scope dan Lexical closures! Berikan contohnya!  **Jawab:**   * Lexical Scope (Lingkup Lexical):   Lingkup Lexical mengacu pada aturan yang menentukan bagaimana variabel diidentifikasi dan diakses oleh fungsi dalam kode berdasarkan lokasi fisik dalam kode sumber (lexically). Ini berarti bahwa ketika Anda membuat variabel di dalam sebuah fungsi, variabel tersebut hanya dapat diakses oleh fungsi tersebut dan fungsi-fungsi yang bersarang di dalamnya (nested functions). Variabel di luar fungsi tidak dapat diakses oleh fungsi yang ada dalam fungsi itu, kecuali jika variabel tersebut di-pass sebagai parameter atau dideklarasikan dalam lingkup yang lebih tinggi (scope yang lebih tinggi).  Contoh Lingkup Lexical:     * Lexical Closures (Penutupan Lexical):   Penutupan Lexical adalah konsep di mana sebuah fungsi yang dibuat di dalam fungsi lain (nested function) dapat "menutup" (mengakses dan menyimpan referensi ke) variabel-variabel yang ada dalam lingkup fungsi yang lebih tinggi, bahkan setelah fungsi yang lebih tinggi telah selesai dieksekusi. Ini berarti fungsi yang bersarang dapat memiliki akses ke variabel-variabel dari lingkup yang lebih tinggi meskipun lingkup tersebut sebenarnya sudah tidak aktif.  Contoh Lexical Closures: |
| 7 | Jelaskan dengan contoh cara membuat return multiple value di Functions!   * Menggunakan Map:   Anda dapat menggunakan map untuk mengelompokkan nilai-nilai yang ingin Anda kembalikan dalam sebuah fungsi. Kemudian, Anda dapat mengembalikan map tersebut.  Contoh:    Dalam contoh ini, kita mengembalikan multiple values dalam bentuk map yang berisi hasil penjumlahan, pengurangan, dan perkalian dari dua bilangan.   * Menggunakan List atau Tuple:   Anda juga dapat menggunakan list atau tuple untuk mengembalikan multiple values.  Contoh menggunakan list:    Contoh menggunakan tuple:    Dalam kedua contoh ini, kita mengembalikan multiple values dalam bentuk list atau tuple, dan kita mengakses nilai-nilai tersebut dengan indeks (list) atau properti item (tuple).   * Menggunakan Class atau Object:   Anda dapat membuat class khusus yang memiliki properti untuk multiple values yang ingin Anda kembalikan, kemudian mengembalikan objek dari class tersebut.  Contoh:    Dalam contoh ini, kita mengembalikan objek dari class HasilKalkulasi yang memiliki properti untuk multiple values. |