LAPORAN PRAKTIKUM ALGORITMA PEMOGRAMAN 6



Oleh :

RAFKI AHMAD PAGAMANDA

NIM 2311533016

MATA KULIAH

ALGORITMA PEMOGRAMAN

DOSEN PENGAMPU : Dr.WAHYUDI ST, MT.

FAKULTAS TEKNOLOGI INFORMASI

DEPARTEMEN INFORMATIKA

UNIVERSITAS ANDALAS

1. **Pendahuluan**

Pada praktikum ini, kami akan menjelajahi konsep-konsep kunci dalam pemrograman, yakni Ternary, Switch Case, dan Continue, yang memberikan kemampuan tambahan untuk mengendalikan alur eksekusi program. Salah satu aspek penting dalam bahasa pemrograman adalah penggunaan Ternary Operator, sebuah alat yang memungkinkan penulisan kode yang lebih ringkas dan ekspresif untuk kondisional sederhana. Kami juga akan membahas penggunaan Switch Case, sebuah struktur kontrol yang berguna untuk menangani multiple choice atau pilihan dalam program, menggantikan serangkaian pernyataan if-else yang mungkin menjadi kurang efisien. Continue Statement merupakan elemen lain yang akan kami eksplorasi dalam praktikum ini. Ini memungkinkan kita untuk melompati iterasi tertentu dalam loop, meningkatkan efisiensi dan fleksibilitas dalam pengolahan data. Dengan memahami konsep-konsep ini, diharapkan peserta praktikum dapat meningkatkan keterampilan dalam merancang dan mengimplementasikan algoritma yang lebih efisien dan mudah dimengerti.

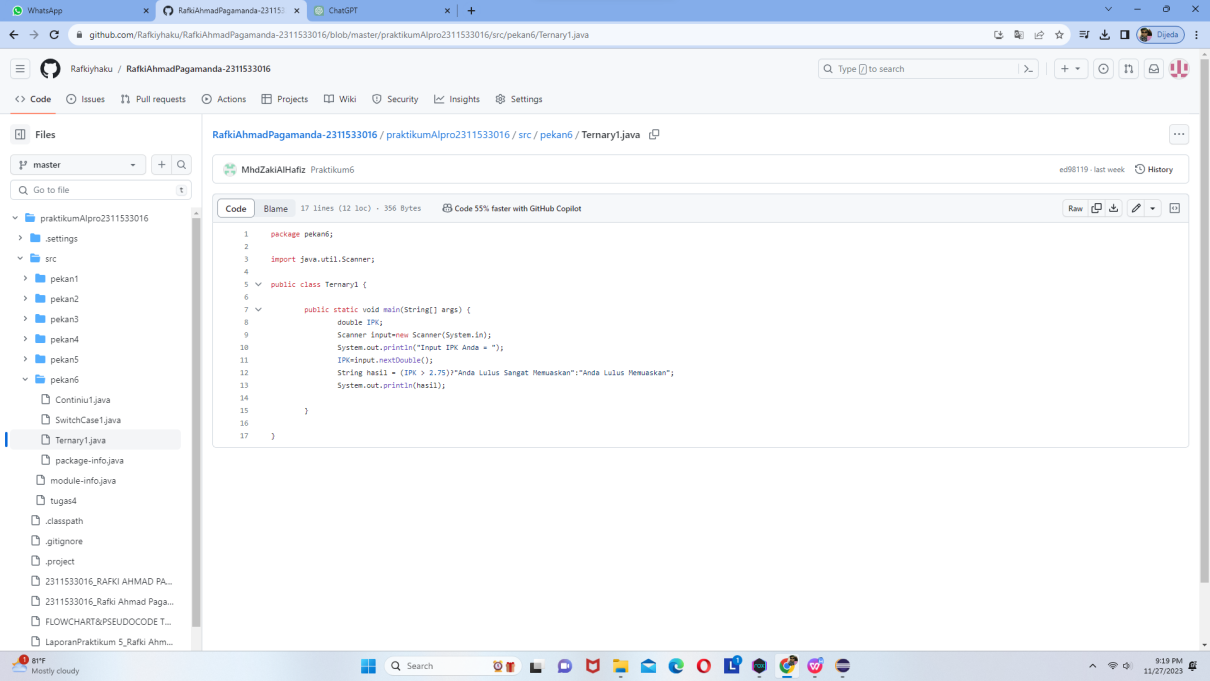
Ternary Operator, Switch Case, dan Continue Statement adalah konsep-konsep kunci dalam pemrograman yang digunakan untuk mengendalikan alur eksekusi program. Dalam laporan praktikum ini, kami akan menjelajahi masing-masing konsep tersebut untuk memahami cara mereka memperkaya desain dan efisiensi kode. Ternary Operator adalah cara singkat untuk mengevaluasi ekspresi kondisional. Penggunaannya mengurangi jumlah baris kode dan membuatnya lebih mudah dipahami. Dalam praktikum ini, kita akan melihat contoh penggunaan Ternary Operator untuk membuat kode lebih ringkas dalam menangani kondisi sederhana. Switch Case adalah struktur kontrol yang memungkinkan pemrogram untuk mengevaluasi ekspresi terhadap serangkaian nilai.

Praktikum ini akan membahas penggunaan Switch Case untuk menggantikan serangkaian pernyataan if-else, meningkatkan kejelasan dan keterbacaan kode ketika ada banyak opsi atau pilihan yang mungkin. Continue Statement digunakan dalam loop untuk melompati iterasi tertentu dan melanjutkan eksekusi dengan iterasi berikutnya. Dengan memahami konsep Continue, peserta praktikum akan dapat meningkatkan kontrol aliran dalam loop, meningkatkan efisiensi, dan mengurangi kompleksitas kode.

1. **Tujuan**
2. Mengetahui pengertian, fungsi, dan penerapan tentang Ternary,
3. Mengetahui pengertian, fungsi, dan penerapan tentang SwitchCase
4. Mengetahui pengertian, fungsi, dan penerapan tentang Continue
5. **Pembahasan**
6. Ternary

Ternary Operator adalah salah satu fitur dalam pemrograman yang digunakan untuk mengevaluasi ekspresi kondisional dengan sintaks yang singkat. Umumnya dituliskan sebagai "condition ? expression\_if\_true : expression\_if\_false". Ternary Operator terdiri dari tiga bagian utama: kondisi (condition), ekspresi yang dievaluasi jika kondisi benar (expression\_if\_true), dan ekspresi yang dievaluasi jika kondisi salah (expression\_if\_false).

Ternary Operator dapat digunakan untuk memberikan nilai awal pada variabel berdasarkan kondisi tertentu. Pemahaman yang baik tentang Ternary Operator membantu pengembang menulis kode yang lebih efisien dan memahami konsep pengambilan keputusan secara kondisional dengan cara yang lebih singkat dan terfokus.

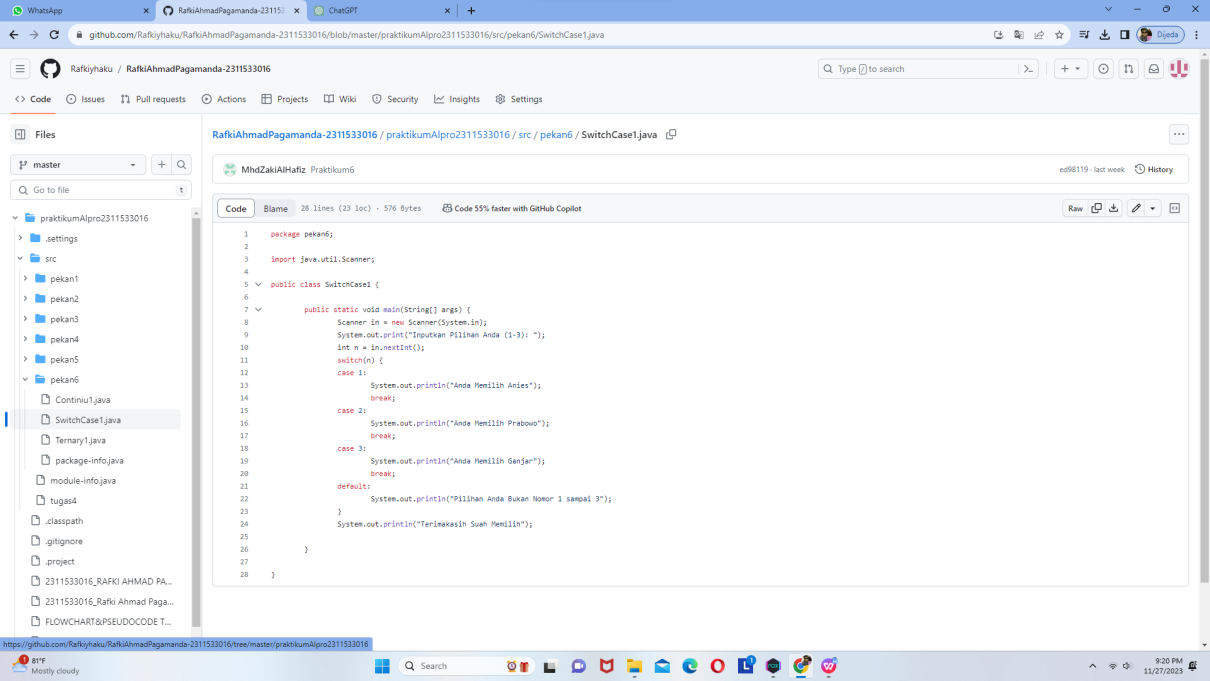


1. Switch-case

Switch-case merupakan struktur kontrol dalam pemrograman yang digunakan untuk mengevaluasi ekspresi terhadap beberapa nilai atau kasus yang mungkin berbeda. Switch-case terdiri dari sebuah blok switch yang mengevaluasi ekspresi, dan serangkaian kasus (case) yang memungkinkan pengembang menentukan tindakan yang sesuai dengan nilai ekspresi.

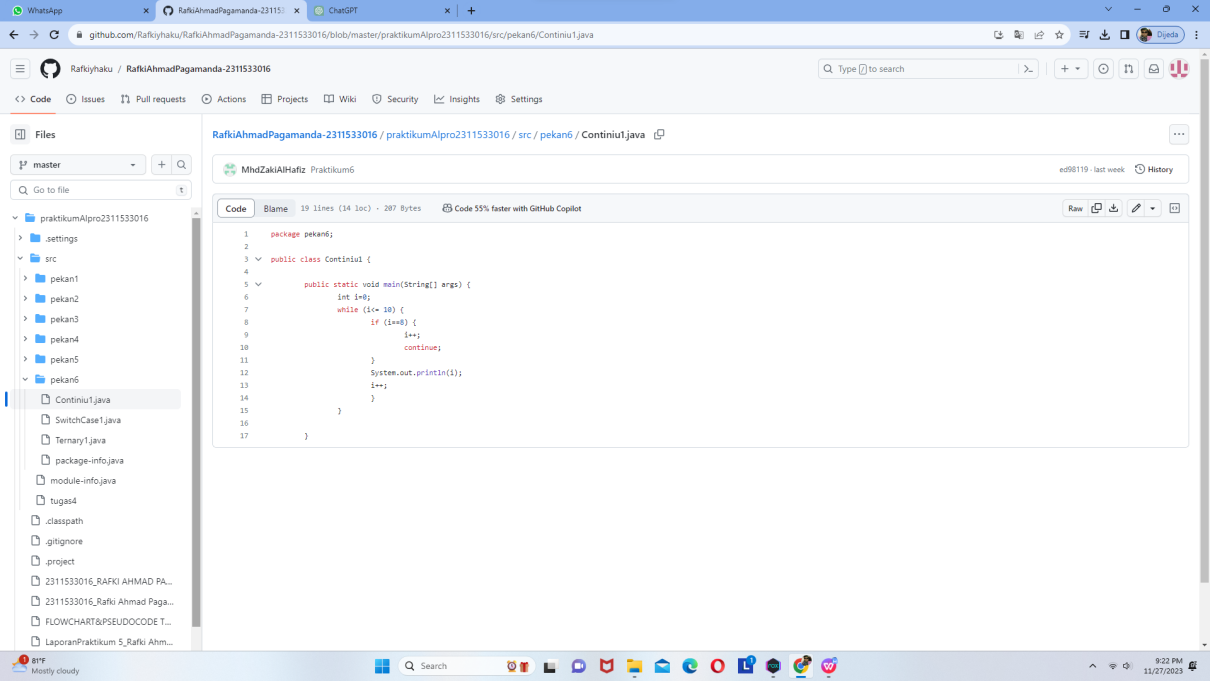
Switch-case dapat membuat kode lebih bersih dan terstruktur daripada penggunaan serangkaian pernyataan if-else yang panjang. Switch-case bekerja dengan indeks kasus, yang memungkinkan eksekusi yang lebih cepat daripada pernyataan if-else di beberapa kasus. Setiap blok kasus diikuti oleh pernyataan break yang menghentikan eksekusi switch-case setelah satu kasus terpenuhi.

Tanpa break, eksekusi akan terus ke kasus berikutnya. Meskipun switch-case memudahkan penanganan beberapa kasus, dalam beberapa situasi, penggunaan if-else mungkin lebih sesuai. Switch-case memberikan struktur yang jelas dan efisien ketika ada banyak pilihan atau kasus yang harus dipertimbangkan dalam suatu program. Pengembang dapat memilih antara switch-case dan if-else berdasarkan kebutuhan dan kejelasan kode.



1. Continue

continue adalah pernyataan dalam pemrograman yang digunakan untuk melanjutkan iterasi berikutnya dalam sebuah loop, mengabaikan sisa kode dalam iterasi saat ini. continue digunakan untuk menghentikan eksekusi kode yang tersisa dalam iterasi saat ini dan melanjutkan ke iterasi berikutnya. Berguna ketika hanya sebagian tertentu dari iterasi yang perlu dilewati, dan eksekusi loop harus dilanjutkan dengan iterasi berikutnya. continue harus digunakan bersamaan dengan suatu kondisi agar efeknya tercapai. Jika tidak ada kondisi, continue akan membuat loop menjadi tak terbatas. continue dapat digunakan baik dalam loop for maupun while. Dalam beberapa kasus, continue dapat digantikan dengan pernyataan if-else, meskipun penggunaannya tergantung pada kejelasan dan gaya kode yang diinginkan.



1. KESIMPULAN
2. Ternary Operator

Ternary Operator adalah alat yang sangat berguna untuk mengevaluasi ekspresi kondisional secara singkat dan ringkas. Dengan sintaks yang sederhana, Ternary Operator dapat menggantikan blok if-else dalam kasus kondisi yang sederhana, meningkatkan kejelasan dan kekompakan kode. Pemahaman yang baik tentang Ternary Operator dapat membantu pengembang menulis kode yang lebih efisien dan mudah dipelajari.

1. Switch Case Statement

Switch Case adalah struktur kontrol yang memungkinkan penanganan efisien dari multiple choice atau opsi dalam program. Penggunaannya mengurangi kebutuhan akan serangkaian pernyataan if-else yang panjang, membuat kode lebih bersih dan mudah dimengerti. Switch Case cocok digunakan ketika terdapat serangkaian kondisi yang bergantung pada nilai ekspresi tertentu. Kelebihan Switch Case terutama muncul dalam situasi di mana beberapa kasus dapat dieksekusi secara bersamaan.

1. Continue Statement

Continue Statement memberikan fleksibilitas tambahan dalam kontrol aliran loop. Dengan menggunakan Continue, pengembang dapat melewati iterasi tertentu dan melanjutkan ke iterasi berikutnya berdasarkan kondisi tertentu. Ini dapat meningkatkan efisiensi dalam pengolahan data dalam loop, memungkinkan pengembang untuk menghindari eksekusi kode yang tidak perlu dan fokus pada langkah-langkah yang relevan.

Secara keseluruhan, pemahaman mendalam tentang Ternary Operator, Switch Case, dan Continue Statement adalah keterampilan yang penting dalam pengembangan perangkat lunak. Dengan menggunakan konsep-konsep ini secara bijak, pengembang dapat meningkatkan kejelasan, keterbacaan, dan efisiensi kode mereka, membantu dalam pengembangan aplikasi yang lebih baik dan dapat dipelihara.