**LAPORAN UJIAN TENGAH SEMESTER**

**KOMPUTER DAN PEMROGRAMAN**



**DISUSUN OLEH :**

Kelompok 15

1. Azzahra Faranisa (G1A023010)
2. Muhammad Hafizh Ario Diffo (G1A023032)
3. Muhammad Sahlan Habibi (G1A023058)

Kelas : B Informatika

NAMA ASISTEN DOSEN

Randi Julian Saputra (G1A019066)

# Dosen Pengampu

Arie Vatresia, S.T. M.TI., P.hD

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS BENGKULU**

**2023**

# **LANDASAN TEORI**

Java merupakan perangkat lunak produksi Sun Microsystem Inc.. Java merupakan perangkat lunak untuk pemrograman beberapa tujuan (multi purpose), multiplatform (dapat berjalan di beberapa sistem operasi), mudah dipelajari dan powerful. Aplikasi-aplikasi yang dapat dibuat dengan Java, meliputi web programming (pemrograman web), Desktop Programming (pemrograman desktop), mobile programming (pemrograman mobile/handphone).Perangkat lunak Java sintaknya (tulisannya) mirip dengan C, karena bahasa Java dibuat memakai bahasa pemrograman C, tetapi bahasa Java menyempurnakan kekurangan C. Pertama rilis, Java disebut JDK (Java Development Kit), hingga JDK Versi 2 atau dikenal dengan Java 2. Dibagi menjadi 3 edisi, yaitu J2SE (Java 2 Standard Edition), J2EE (Java 2 Enterprise Edition) dan J2ME (Java 2 Micro Edition) Java bersifat Write Once, Run Anywhere (program yang ditulis satu kali dan dapat berjalan pada banyak platform). Dengan demikian, tidak mengherankan bila aplikasi yang dibuat dengan Java bisa ditemukan di lingkungan komputer dan smartphone tanpa perbedaan yang berarti. Sama seperti pemrograman pada umumnya, Java merupakan bahasa pemrograman yang mampu bekerja dengan sebuah database. Java merupakan bahasa pemrograman yang dikembangkan dari bahasa pemrograman C++, sehingga hal pemrograman ini seperti bahasa CH. Terdapat 2 Program-program java, yaitu: Application adalah program yang dapat dijalankan di command prompt suatu sistem operasi Application memili dua jenis yaitu window-based application atau console application, Window-based applicaton menggunakan tampilan antarmuka grafikal GUT) sedangkan console application menggunakan karakter karakter tanpa antarmuka (Non GUI). ✓ Applet yaitu program java yang dieksekusi di dalam web page yang mendukung java sepert Microsoft Internet Explorer. Kelebihan utama Java adalah dapat dijalankan di beberapa platform/sistem operasi komputer, sesuai dengan prinsip tulis sekali, jalankan di mana saja. Dengan kelebihan ini, programmer cukup menulis sebuah program Java dan dikompilasi (diubah, dan bahasa yang dimengerti manusia menjadi bahasa mesin/bytecode) sekali, lalu hasilnya dapat dijalankan di atas beberapa platform tanpa perubahan. Setiap edisi Java terdiri atas dua komponen utama yaitu : Java Application Programming Interface (Java API)dan Java Run Time Environment (JRE). JRE merupakan lingkungan yang membuat aplikasi Java dapat dijalankan.

Java bermula dari proyek penelitian perusahaan Sun Microsystems. dengan nama sandi Green pada tahun 1991. Terdapat prediksi bahwa mikroprosesor akan digunakan luas pada peralatan-peralatan elek- tronik. Karena adanya bermacam tipe mikroprosesor, maka dibutuhkan sebuah bahasa pemrograman yang dapat berjalan di semua mikro- prosesor. Terciptalah sebuah bahasa pemrograman baru. Oleh James Gosling. yaitu salah satu orang yang berperan besar dalam proyek tersebut, program ini diberi nama Oak. Sesuai dengan pohon Oak yang tumbuh dan bisa dilihat melalui jendela kerjanya di Sun Microsystems, Selang beberapa waktu kemudian, ditemukan bahwa sudah ada bahasa. pemrograman dengan nama Oak. Akhirnya setelah beberapa pegawai Sun mengunjungi sebuah kedai kopi, nama bahasa pemrograman ini diganti dengan Java. Java merupakan salah satu jenis biji kopi yang ada di kedai tersebut, yaitu biji kopi Java. Sun Microsystems mengumumkan kehadiran bahasa Java secara for- mal di tahun 1995. Bahasa ini mulai disambut hangat masyarakat luas seiring dengan meledaknya era internet.

Tahapan dalam menuliskan program java yaitu dengan cara sebagai berikut:

1. Menuliskan kode program java.

Anda dapat menulis bahasa Java menggunakan aplikasi teks editor biasa, misalnya Notepad. Namun, untuk kemudahan pemrograman, sangat dianjurkan menggunakan aplikasi integrated Development Environment (IDE) Java. IDE Java merupakan aplikasi yang digunakan khusus untuk mengembangkan program Java. Di dalamnya sudah tersedia banyak fitur untuk memudahkan pemrograman, seperti kom- pilasi dan debugging, pengaturan file-file Java, dan deteksi kesalahan sintaks pemrograman Java. IDE Java yang banyak dipakai untuk menuliskan program Java, antara lain NetBeans, Eclipse, JBuilder, JCreator, dan IntelliJ IDEA. Kode program Java biasanya disimpan dengan ekstensi file .Java. Selain menggunakan IDE, penulisan program Java yang masih sederhana dapat dilakukan menggunakan text editor, seperti Notepad++ dan Edit Plus. Kedua aplikasi ini dapat mengenali sintaks Java sehingga memudahkan pengguna menuliskan kode program.

1. Kompilasi program Java.

Setelah kode Java selesai dituliskan, kode tersebut dikompilasi sehingga menjadi file binary. File binary merupakan hasil ter- jemahan kode program yang kita tulis menjadi bentuk yang dapat dibaca computer. File binary hasil kompilasi program Java disebut juga dengan bytecode. File ini disimpan dengan ekstensi file .class. Setiap file berisi kode program Java (Java), akan dikompilasi sendiri- sendiri sehingga setiap file Java mempunyai hasil kompilasinya (.class).

1. Menjalankan program Java.

Hasil kompilasi akan dijalankan oleh Java Virtual Machine dan program Anda siap dipakai. Jadi, Java merupakan bahasa pemrograman intepreted sekaligus compiled. Pertama dikom-pilasi dulu menjadi bytecode. Setelah itu diintepretasikan dari Java bytecode oleh JVM menjadi native code sehingga di- mengerti oleh prosesor yang digunakan.

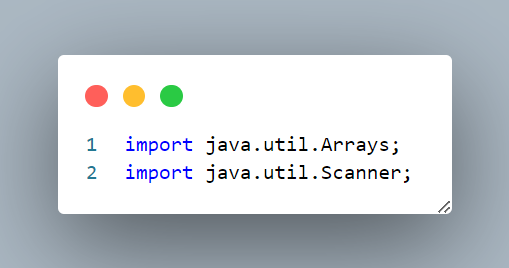
**SOAL DAN PEMBAHASAN**

1.Buatlah Game Tictactoe atau Turtle Maze menggunakan java :

Game TicTacToe

* Import Statements

Printscreen



Kelompok 15

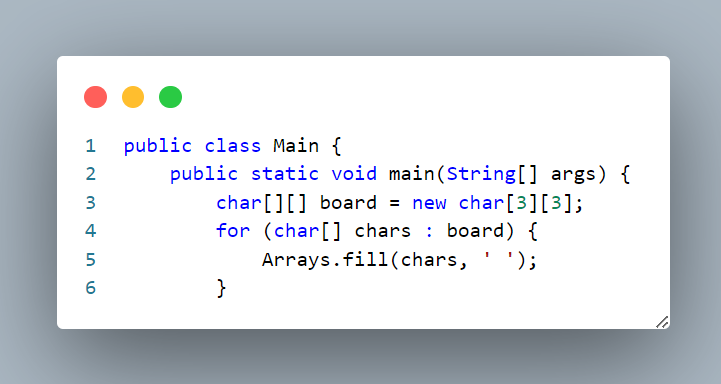
Gambar 1.1 Source Code

Penjelasan Source code:

Baris pertama dan kedua adalah pernyataan impor (import statements). Digunakan untuk mengimpor kelas-kelas yang diperlukan dari pustaka standar Java. Pada code diatas mengimpor dua kelas: `Arrays` dan `Scanner`. Kelas `Arrays` digunakan untuk melakukan operasi terkait array, sedangkan Scanner digunakan untuk membaca input dari pengguna melalui keyboard.

* Membuat Papan Permainan

Printscreen



Kelompok 15

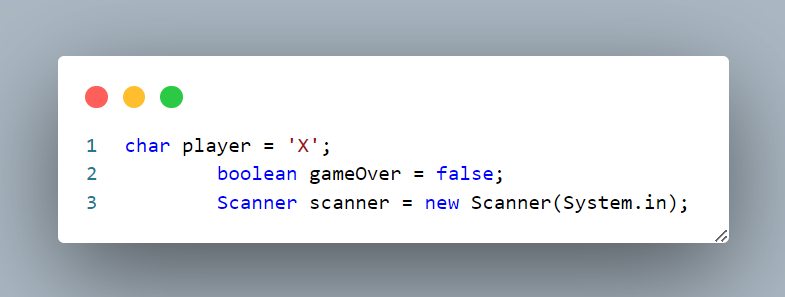
Gambar 1.2 Source code

Penjelasan Source code:

Baris pertama adalah deklarasi kelas utama (main class) dalam program Java. Setiap program Java harus memiliki kelas utama dengan nama yang sama dengan nama file program (dalam hal ini, `Main.java`). Kode program utama akan berada di dalam metode main dalam kelas ini. Kemudian baris kedua adalah deklarasi metode main, yang merupakan titik awal dari eksekusi program. Semua kode yang ingin dijalankan saat program dimulai akan ditempatkan di dalam metode ini. Lalu `char[ ][ ] board = new char[3][3];` Baris ketiga ini mendeklarasikan dan menginisialisasi sebuah array dua dimensi dengan nama board. Array ini memiliki dimensi 3x3, yang berarti terdiri dari 3 baris dan 3 kolom. Setiap elemen dalam array adalah tipe data karakter (char). Pada saat inisialisasi, semua elemen dalam array akan diisi dengan karakter spasi (' '). Baris ke empat dan lima ini menggunakan perulangan for-each untuk menginisialisasi setiap elemen dalam array board dengan karakter spasi (' '). Pada dasarnya, kode ini mengisi seluruh papan permainan dengan karakter spasi agar papan tersebut menjadi kosong dan siap untuk digunakan dalam permainan.

* Menginisialisasi beberapa variable

Printscreen



Kelompok 15

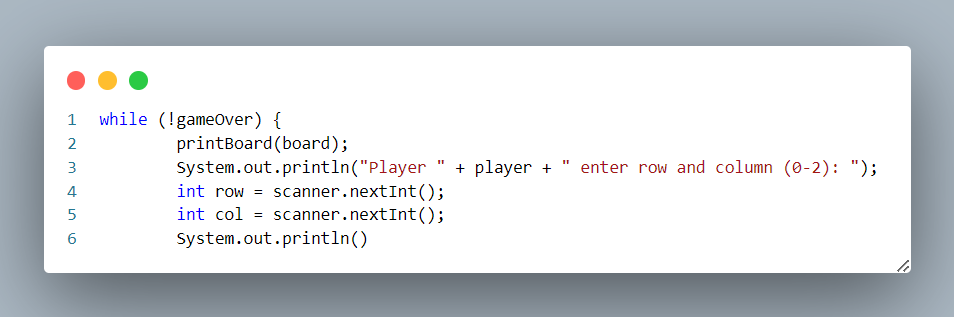
Gambar 1.3 Source code

Penjelasan source code:

Baris pertama mendeklarasikan variabel `player` dengan tipe data char dan menginisialisasinya dengan karakter 'X'. Variabel ini kemungkinan besar digunakan untuk melacak giliran pemain dalam permainan, seperti dalam permainan tic-tac-toe di mana 'X' biasanya merupakan pemain pertama. Baris kedua mendeklarasikan variabel `gameOver` dengan tipe data `Boolean` dan menginisialisasinya dengan nilai `false`. Variabel ini mungkin digunakan untuk mengindikasikan apakah permainan sudah berakhir atau belum. Ketika `gameOver` bernilai `true`, ini bisa menjadi tanda bahwa permainan telah selesai dan tidak akan ada gerakan tambahan yang diperbolehkan. Baris ketiga membuat objek `Scanner` dengan nama `scanner` yang akan digunakan untuk membaca input dari pengguna melalui keyboard. Dengan `System.in` sebagai argumen konstruktor `Scanner`, program akan membaca input dari keyboard pengguna. Kode ini menunjukkan inisialisasi beberapa variabel yang penting dalam permainan atau program Anda. Variabel `player` bisa digunakan untuk mengidentifikasi giliran pemain, `gameOver` untuk mengontrol status permainan, dan `scanner` untuk membaca input dari pengguna.

* Mengatur giliran pemain, mencetak papan permainan, meminta input dari pemain, dan membaca input

Printscreen



Kelompok 15

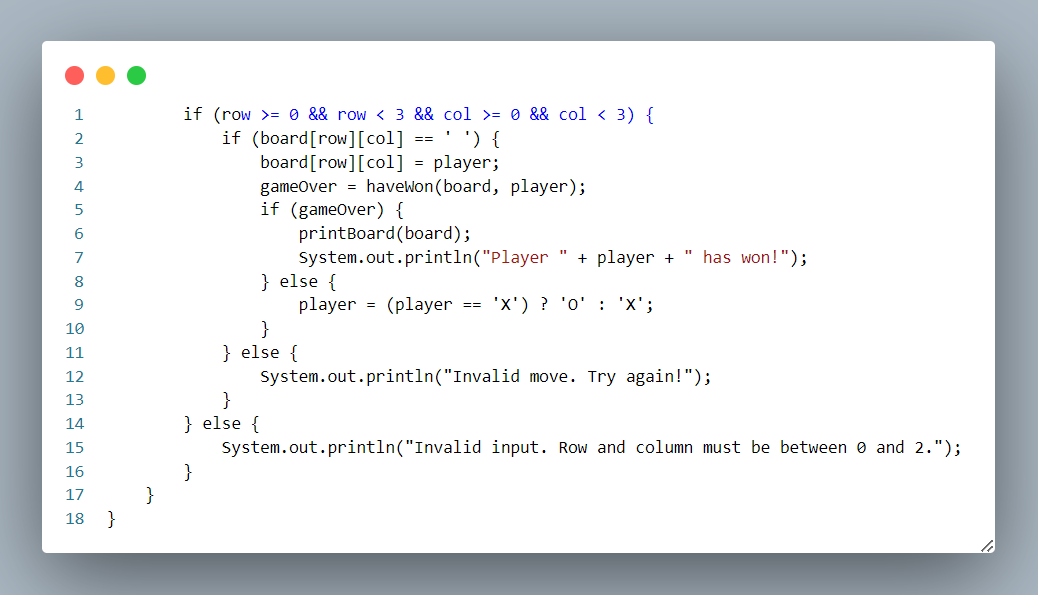
Gambar 1.4 Source code

Penjelasan Source code:

Baris pertama mendefinisikan sebuah perulangan `while` yang akan terus berjalan selama kondisi di dalam tanda kurung (dalam hal ini `!gameOver`) bernilai `true`. Ini berarti perulangan akan terus berjalan selama `gameOver` tidak sama dengan `true`. Dengan kata lain, permainan akan berlanjut selama `gameOver` adalah `false`.Baris kedua memanggil fungsi `printBoard(board)` untuk mencetak papan permainan ke layar. Fungsi ini kemungkinan besar akan menampilkan papan permainan kepada pemain sehingga mereka dapat melihat keadaan permainan saat ini. Baris ketiga mencetak pesan ke layar yang memberi tahu pemain yang sedang giliran (dalam variabel `player`) untuk memasukkan baris dan kolom di mana mereka ingin melakukan gerakan. Ini adalah permintaan input dari pemain. Baris ke empat dan ke lima membaca input dari pemain menggunakan objek `scanner` . Input pertama yang dibaca akan disimpan dalam variabel `row` , dan input kedua akan disimpan dalam variabel `col` ini akan memungkinkan pemain untuk memasukkan koordinat (baris dan kolom) di mana mereka ingin menempatkan tanda (misalnya, 'X' atau 'O') di papan permainan. Baris ke enam hanya mencetak baris kosong untuk memisahkan setiap giliran pemain dan papan permainan yang dicetak di layar.

* Mengendalikan tindakan pemain, memeriksa validitas gerakan , memperbarui papan permainan, memeriksa pemenang.

Printscreen



Kelompok 15

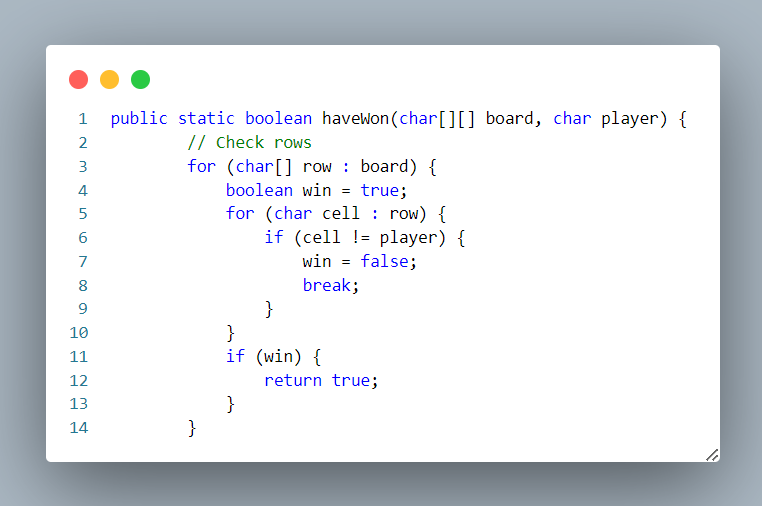
Gambar 1.5 Source code

Penjelasan Source code:

Baris 1, Kode dalam blok ini akan dieksekusi jika kondisi dalam tanda kurung `()` bernilai `true` . Kondisi ini memeriksa apakah nilai `row` dan `col` (baris dan kolom yang dimasukkan oleh pemain) berada dalam rentang 0 hingga 2. Jika baris dan kolom berada dalam rentang yang valid (0, 1, atau 2), maka pernyataan di dalam blok `if` akan dieksekusi. Baris 2 Di dalam blok `if` pertama, kita memeriksa apakah sel yang dipilih oleh pemain di papan permainan (`board`) masih kosong (dengan memeriksa apakah karakter pada posisi tersebut adalah spasi ' '). Jika sel masih kosong, maka pernyataan di dalam blok ini akan dieksekusi. Baris 3 Jika sel yang dipilih masih kosong, maka kode ini akan menempatkan karakter pemain (`player`) di sel tersebut pada papan permainan Baris 4 Setelah pemain melakukan gerakan, kita memanggil fungsi `haveWon(board, player)` untuk memeriksa apakah pemain yang baru saja melakukan gerakan tersebut telah memenangkan permainan. Hasilnya disimpan dalam variabel `gameOver` . Jika pemain telah memenangkan permainan, variabel `gameOver` akan diatur menjadi `true` . Baris 5-10 Jika pemain telah memenangkan permainan (nilai `gameOver` adalah `true`), maka papan permainan akan dicetak ke layar, dan pesan "Player [nama pemain] has won!" akan ditampilkan. Jika pemain belum memenangkan permainan, maka giliran pemain akan berganti. Jika pemain sebelumnya adalah 'X', maka giliran akan beralih ke 'O', dan sebaliknya. Baris 11-13 Jika kondisi dalam blok `if` kedua (ketika sel yang dipilih oleh pemain sudah terisi) bernilai `false` , maka pernyataan dalam blok `else` akan dieksekusi, yang akan mencetak pesan "Invalid move. Try again!" ke layar. Ini memberi tahu pemain bahwa gerakan yang mereka coba lakukan tidak valid. Baris 14-16 jika kondisi dalam blok `if` pertama (ketika baris atau kolom yang dimasukkan oleh pemain berada di luar rentang yang valid) bernilai `false` , maka pernyataan dalam blok `else` ini akan dieksekusi. Ini mencetak pesan "Invalid input. Row and column must be between 0 and 2." ke layar, memberi tahu pemain bahwa input yang mereka masukkan tidak valid.

* Memeriksa kemenangan dalam permainan Tic-Tac-Toe.

Printscreen



Kelompok 15

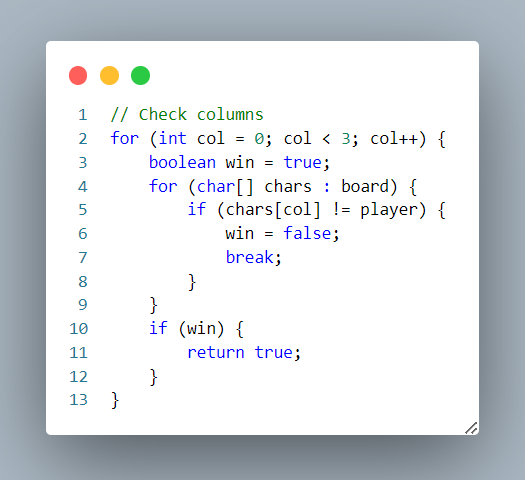
Gambar 1.6 Source code

Penjelasan Source code:

`public static boolean haveWon(char[][] board, char player)` Ini adalah deklarasi dari sebuah metode bernama haveWon. Metode ini menerima dua parameter `board` , yang merupakan papan permainan dalam bentuk array dua dimensi, dan `player` , yang adalah karakter pemain yang akan diperiksa kemenangannya. Kemudian, kode ini memulai pengulangan `for` pertama. Ini berarti kode akan melakukan iterasi melalui setiap baris di papan permainan ( `board` ) menggunakan perulangan for-each. Setiap baris direpresentasikan oleh array karakter ( `char[]` ) yang disebut `row` . Di dalam perulangan for-each baris, kita memiliki perulangan for kedua: Ini adalah perulangan yang digunakan untuk mengiterasi melalui setiap sel (karakter) dalam baris tersebut. Setiap sel direpresentasikan oleh karakter `cell` Di dalam perulangan for kedua, kode melakukan pemeriksaan: Ini berarti, jika sel (cell) dalam baris tidak sama dengan karakter pemain ( `player` ), maka variabel `win` akan diubah menjadi `false` , dan perulangan dalam baris saat ini akan dihentikan menggunakan `break` . Setelah selesai memeriksa seluruh sel dalam baris, kode akan memeriksa nilai dari variabel `win` Jika variabel `win` tetap `true` setelah memeriksa semua sel dalam baris, maka ini berarti bahwa semua sel dalam baris tersebut diisi dengan karakter pemain yang sama, dan pemain tersebut telah memenangkan permainan dalam baris tersebut.

* Menguji kondisi kemenangan dalam kolom kolom papan permainan.

Printscreen



Kelompok 15

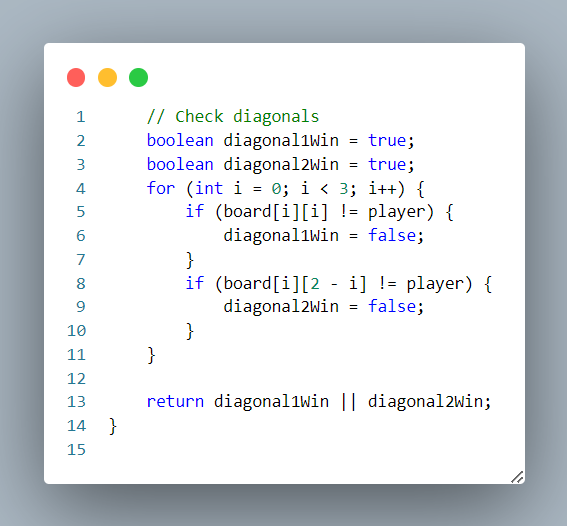
Gambar 1.7 Source code

Penjelasan Source code:

`// Check columns` Ini adalah komentar yang memberikan informasi tentang apa yang akan dilakukan kode berikutnya, yaitu memeriksa kolom-kolom pada papan permainan. Selanjutnya, ada sebuah perulangan `for` yang digunakan untuk mengiterasi melalui setiap kolom pada papan permainan. Ini dilakukan dengan menggunakan variabel `col` yang mulai dari 0 hingga kurang dari 3 . Di dalam perulangan untuk setiap kolom, kode inisialisasi sebuah variabel boolean win menjadi `true` . Variabel ini akan digunakan untuk menentukan apakah pemain telah memenangkan kolom tersebut atau tidak. Kemudian, terdapat perulangan for yang digunakan untuk mengiterasi melalui setiap baris pada papan permainan ( `board` ). Dalam konteks ini, kita ingin memeriksa apakah karakter pemain ( `player` ) sama di setiap kolom yang sama pada setiap baris. Di dalam perulangan untuk setiap baris, kode melakukan pemeriksaan karakter pada kolom tertentu ( `col` ) dan membandingkannya dengan karakter pemain yang sedang diperiksa. Jika karakter dalam kolom yang sedang diperiksa tidak sama dengan karakter pemain , maka variabel `win` akan diubah menjadi `false` , dan perulangan dalam kolom saat ini akan dihentikan menggunakan `break` . Setelah selesai memeriksa semua baris dalam kolom yang sedang diperiksa, kode akan memeriksa nilai dari variabel `win`. Jika variabel `win` tetap `true` setelah memeriksa semua baris dalam kolom tersebut, maka ini berarti bahwa semua sel dalam kolom tersebut diisi dengan karakter pemain yang sama, dan pemain tersebut telah memenangkan permainan dalam kolom tersebut.

* Menguji kondisi kemenangan dalam kolom-kolom papan permainan.

Printscreen



Kelompok 15

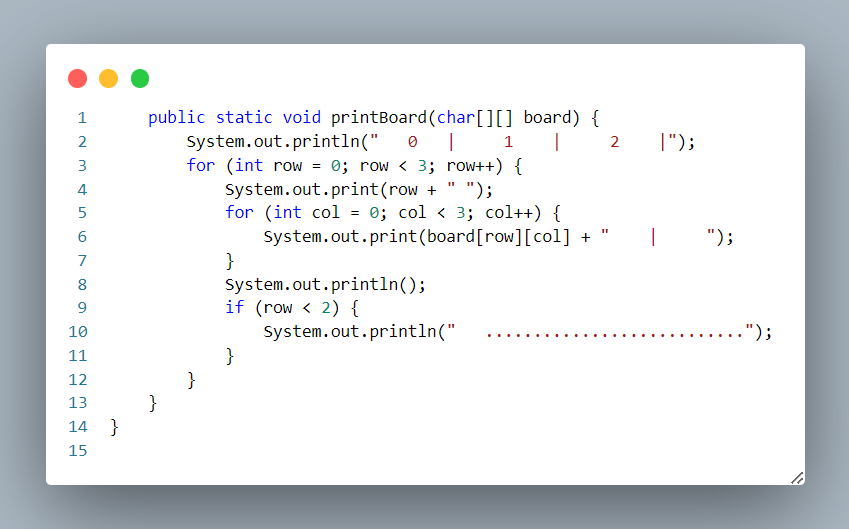
Gambar 1.8 Source code

Penjelasan Source code:

`boolean diagonal1Win = true;` dan `boolean diagonal2Win = true;` adalah dua variabel boolean yang digunakan untuk melacak apakah pemain telah memenangkan permainan melalui dua diagonal yang berbeda. Pada awalnya, kita asumsikan bahwa pemain telah memenangkan kedua diagonal ini. Kemudian ada sebuah loop `for (int i = 0; i < 3; i++)` yang akan berjalan tiga kali. Ini digunakan untuk melakukan iterasi melalui baris dan kolom dalam matriks atau papan permainan. Kode di dalam loop ini memeriksa setiap sel dalam dua diagonal. Di dalam loop, ada dua pernyataan `if` yang digunakan untuk memeriksa apakah tanda (misalnya 'X' atau 'O') pada sel tertentu dalam diagonal pertama atau diagonal kedua tidak sama dengan tanda pemain yang sedang diuji. Jika salah satu dari sel dalam diagonal tidak sesuai dengan tanda pemain, maka variabel `diagonal1Win` atau `diagonal2Win` akan diubah menjadi `false` . Akhirnya, setelah loop selesai, fungsi ini akan mengembalikan hasil `diagonal1Win || diagonal2Win` . Artinya, jika salah satu dari kedua diagonal tersebut sesuai dengan tanda pemain (semua sel di diagonal tersebut memiliki tanda yang sama), maka hasilnya adalah `true` , yang berarti pemain telah memenangkan permainan melalui salah satu dari kedua diagonal tersebut. Jika kedua diagonal tidak sesuai, maka hasilnya adalah `false` , yang berarti pemain belum memenangkan permainan melalui diagonal mana pun.

* Mencetak atau menampilkan papan permainan ke layar dalam format yang terstruktur

Printscreen



Kelompok 15

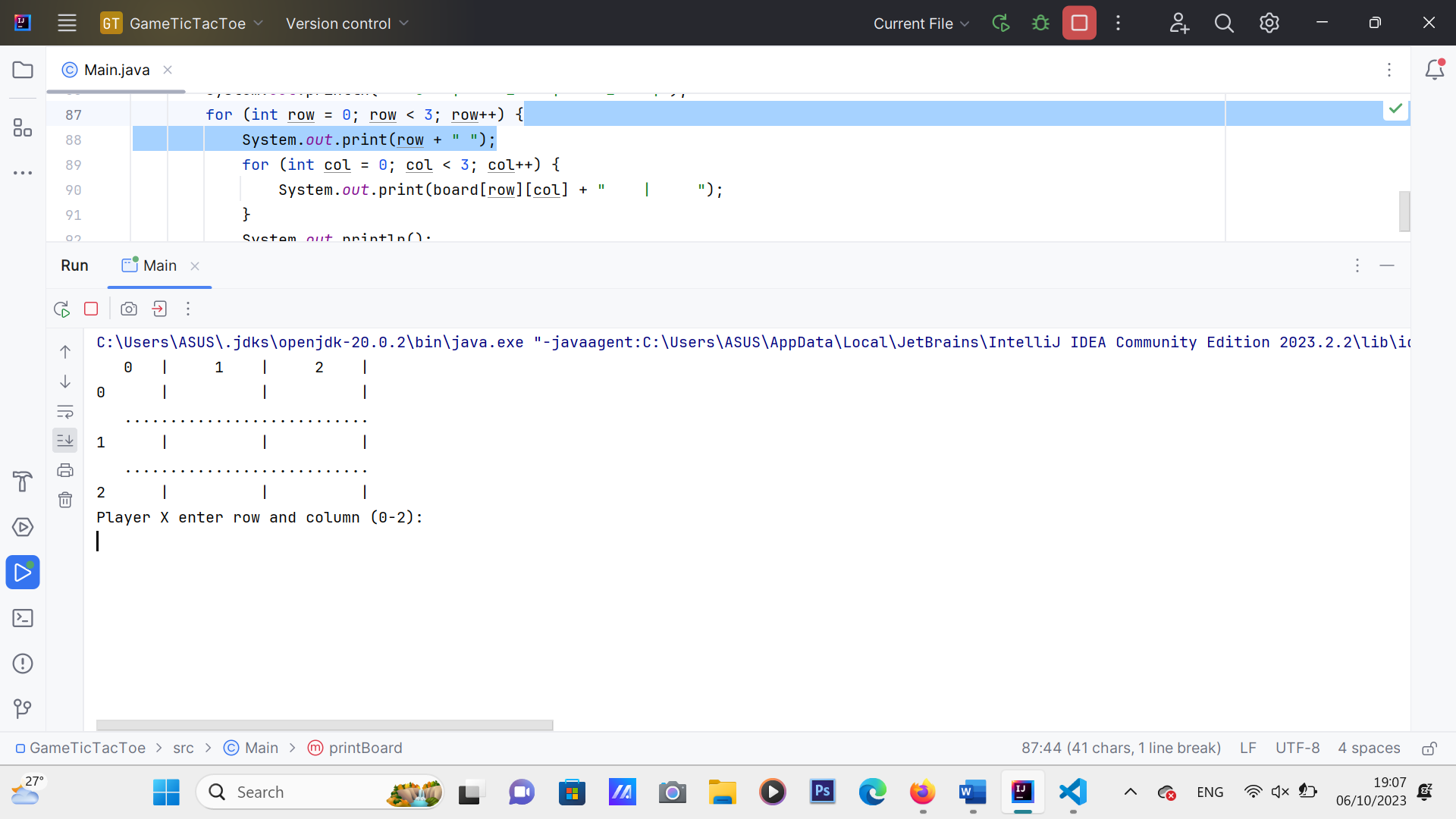
Gambar 1.9 Source code

Penjelasan Source code:

`System.out.println(" 0 | 1 | 2 |");:` Ini adalah baris pertama yang mencetak header untuk kolom papan permainan. Ini mencetak nomor kolom (0, 1, 2) di atas setiap kolom papan permainan untuk memberi pemain referensi terhadap posisi tanda mereka. `for (int row = 0; row < 3; row++) {:` Ini adalah loop luar yang berjalan tiga kali untuk setiap baris pada papan permainan. Di papan 3x3, kita memiliki tiga baris.`System.out.print(row + " ");:` Ini mencetak nomor baris (0, 1, 2) di samping setiap baris papan permainan untuk memberi pemain referensi terhadap posisi tanda mereka di baris tersebut. `for (int col = 0; col < 3; col++) {: ` Ini adalah loop dalam yang berjalan tiga kali untuk setiap kolom pada papan permainan. Di papan 3x3, kita memiliki tiga kolom. `System.out.print(board[row][col] + " | ");: ` Ini mencetak tanda (biasanya 'X', 'O', atau spasi kosong) yang ada di sel (baris, kolom) pada papan permainan. Setelah mencetak tanda, itu juga mencetak garis vertikal "|" yang memisahkan satu sel dari sel berikutnya. `System.out.println();: ` Ini digunakan untuk pindah ke baris berikutnya setelah mencetak semua kolom di baris saat ini. `if (row < 2) { System.out.println(" ..........................."); }: ` Ini digunakan untuk mencetak garis horizontal "..........................." yang memisahkan satu baris dari baris berikutnya, kecuali untuk baris terakhir. Ini memberi tampilan papan permainan yang lebih terstruktur dan mudah dibaca. Setelah selesai, metode `printBoard` ini akan mencetak tampilan papan permainan ke layar sesuai dengan isi papan yang diberikan dalam bentuk array dua dimensi `board` . Tampilan ini membantu pemain dan pengguna untuk melihat dan memahami keadaan papan permainan secara visual.

* Output Papan

Printscreen



Kelompok 15

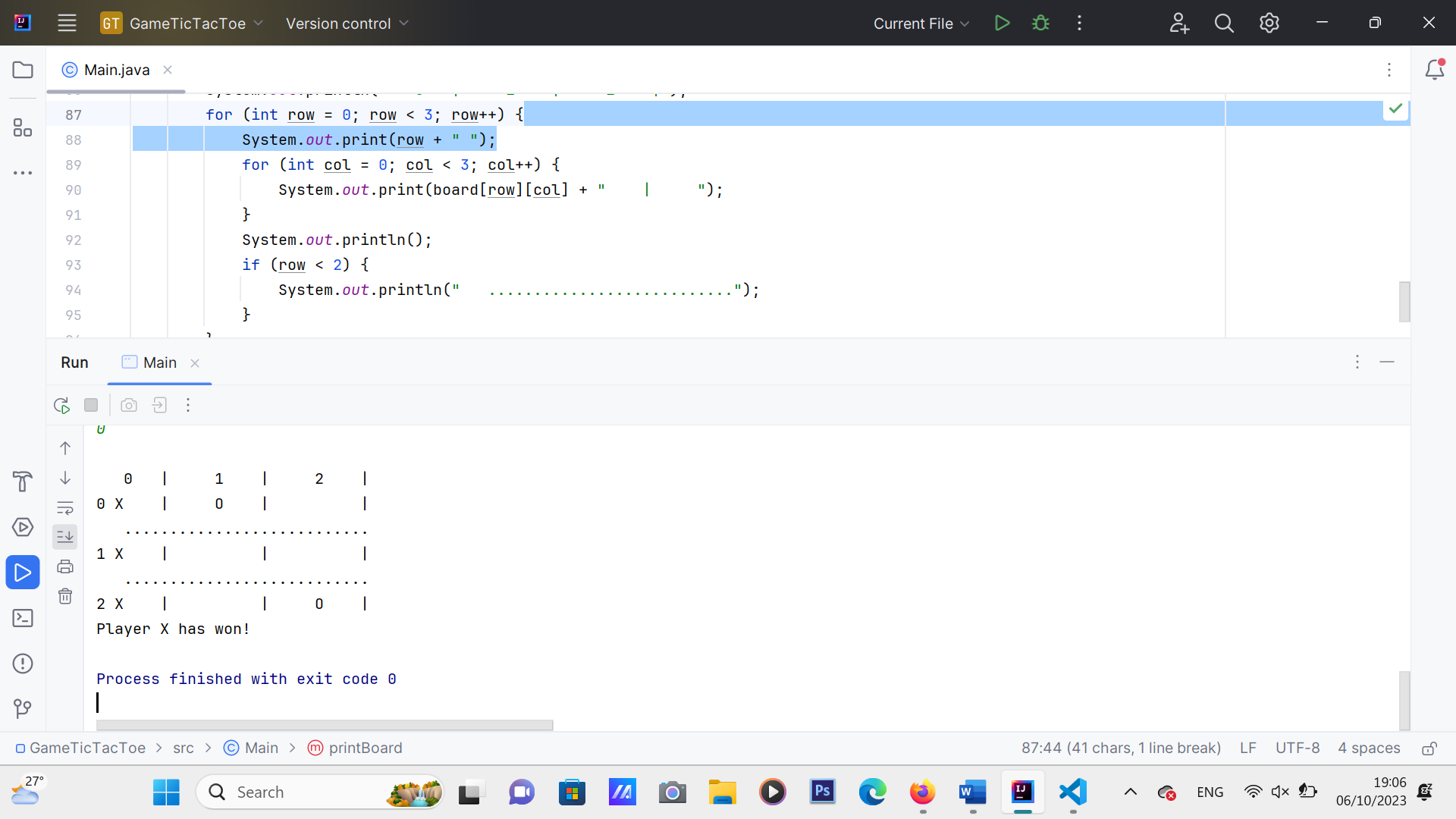
Gambar 1.10 Source code

Penjelasan Source code:

Saat ini menampilkan pemain X memiliki giliran. Papan memiliki 3 baris (0, 1, dan 2) dan 3 kolom (0, 1, dan 2) yang diberi nomor. Pada saat ini, pemain X bisa memulai permainan dengan cara memilih di mana saja dia ingin menempatkan tanda X-nya dengan memasukkan nomor baris dan kolom yang sesuai (misalnya, "1 0" untuk menempatkan X di baris 1, kolom 0)

* Output X win

Printscreen

* 

Kelompok 15

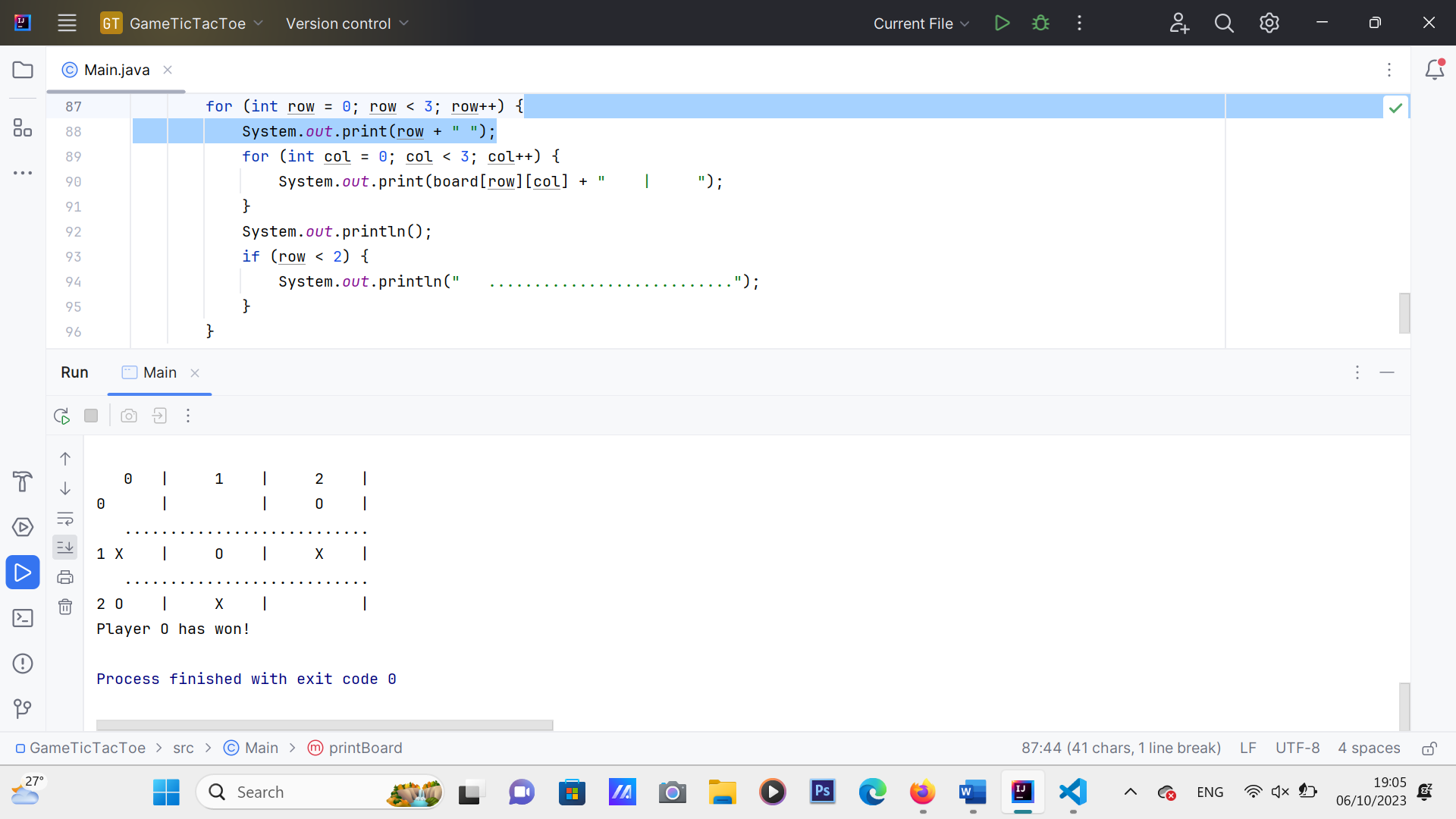
Gambar 1.11 Source code

Penjelasan Output code:

Output tersebut menunjukkan bahwa pemain X telah menang dalam permainan Tic Tac Toe dengan menempatkan tanda X di semua kolom pertama.

* Output 0 Menang

Printscreen



Kelompok 15

Gambar 1.12 Source code

Penjelasan output code:

Output tersebut menunjukkan bahwa pemain O telah menang dalam permainan Tic Tac Toe dengan menempatkan tanda 0 secara diagonal.

**KESIMPULAN DAN SARAN**

1. **Kesimpulan**

Bahasa pemrograman java merupakan bahasa yang berorientasi objek, java dikembangkan oleh Sun Microsystems pada tahun 1991. Sun Micosystem adalah salah satu perusahaan besar di Amerika Serikat. Java memiliki model yang hampir sama dengan bahasa pemrograman Smaltalk dan C++ akan tetapi bahasa java lebih mudah digunakan dan termasuk dalam platform independent. source code java yang sudah dibuat dapat dijalankan di berbagai jenis sistem operasi Bahasa java juga dirancang untuk pemrograman jaringan yang terjamin keamanannya dan bersifat portabel.

Dalam mendiskusikan Java, kiranya penting sekali untuk mem- bedakan antara bahasa pemrograman Java, Java Virtual Machine, dan platform Java. Bahasa pemrograman Java adalah bahasa yang digunakan untuk menghasilkan aplikasi-aplikasi Java. Pada umum- nya, bahasa pemrograman hanya mendefinisikan sintaks dan peri- Jaku bahasa. Pada saat program Java dikompilasi, la akan dikonversi ke bentuk bytecode, yang merupakan bahasa mesin yang portable. Selan- jutnya, bytecode tersebut dijalankan di Java Virtual Machine (atau disebut Java VM atau JVM). Meskipun JVM dapat diimplementasikan langsung di perangkat keras, namun biasanya diimplementasikan dalam bentuk program perangkat lunak yang mengemulasi mesin (komputer) dan digunakan untuk menginterpretasi bytecode. Platform dapat didefinisikan sebagai perangkat lunak pendukung untuk aktivitas-aktivitas tertentu. Platform Java sendiri pada prin- sipnya berbeda dengan bahasa Java atau JVM. Platform Jave adalah himpunan kelas-kelas Java yang sudah didefinisikan sebelumnya. dan eksis sejak instalasi Java. Platform Java juga mengacu pada lingkungan runtime atau API (Application Programming Interface) Java.

1. **Saran**

Dalam membuat project dengan java ini sebaiknya kita harus memahami konsep dasar Java sebelum melompat ke konsep yang lebih kompleks. Mulai dengan memahami tipe data, variabel, dan struktur kontrol dasar sebelum beralih ke pemrograman. Dengan mempelajari dasar-dasar pemrograman java, nantinya dapat memudahkan kita dalam membuat sebuah program menggunakan java. Selain itu kita juga harus banyak berlatih agar kita terbiasa dengan bahasa-bahasa yang ada di dalam pemrograman java, perlu adanya kesabaran dalam prose pembelajaran dan mempunyai sikap pantang menyerah dan terus berusaha.

**DAFTAR PUSTAKA**

[Elex Media Komputindo](https://www.google.co.id/search?hl=id&gbpv=1&dq=KELEBIHAN+JAVA&pg=PA3&printsec=frontcover&q=inpublisher:%22Elex+Media+Komputindo%22&tbm=bks&sa=X&ved=2ahUKEwjO8cSdsuuBAxUK1zgGHbLcBlAQmxMoAHoECBwQAg). (2020 ). *The 40 Best Java Applications*. [Wahana Komputer](https://www.google.co.id/search?hl=id&gbpv=1&dq=KELEBIHAN+JAVA&pg=PA3&printsec=frontcover&q=inauthor:%22Wahana+Komputer%22&tbm=bks&sa=X&ved=2ahUKEwjO8cSdsuuBAxUK1zgGHbLcBlAQmxMoAHoECB4QAg).

Kadir, A. (2020). *Logika Pemrograman Menggunakan Java*. Elex Media Komputindo.

‌Didik Dwi Prasetyo. (2007). *150 Rahasia Pemrograman Java*. PT Elex Media Komputindo.

Rachmad Hakim S. (2009). *Mastering Java + Cd. (2009).* [Elex Media Komputindo](https://www.google.co.id/search?hl=id&gbpv=1&kptab=overview&q=inpublisher:%22Elex+Media+Komputindo%22&tbm=bks&sa=X&ved=2ahUKEwjpzsDJs-uBAxVa3jgGHc_gDsoQmxMoAHoECA0QAg).