

Sudoku

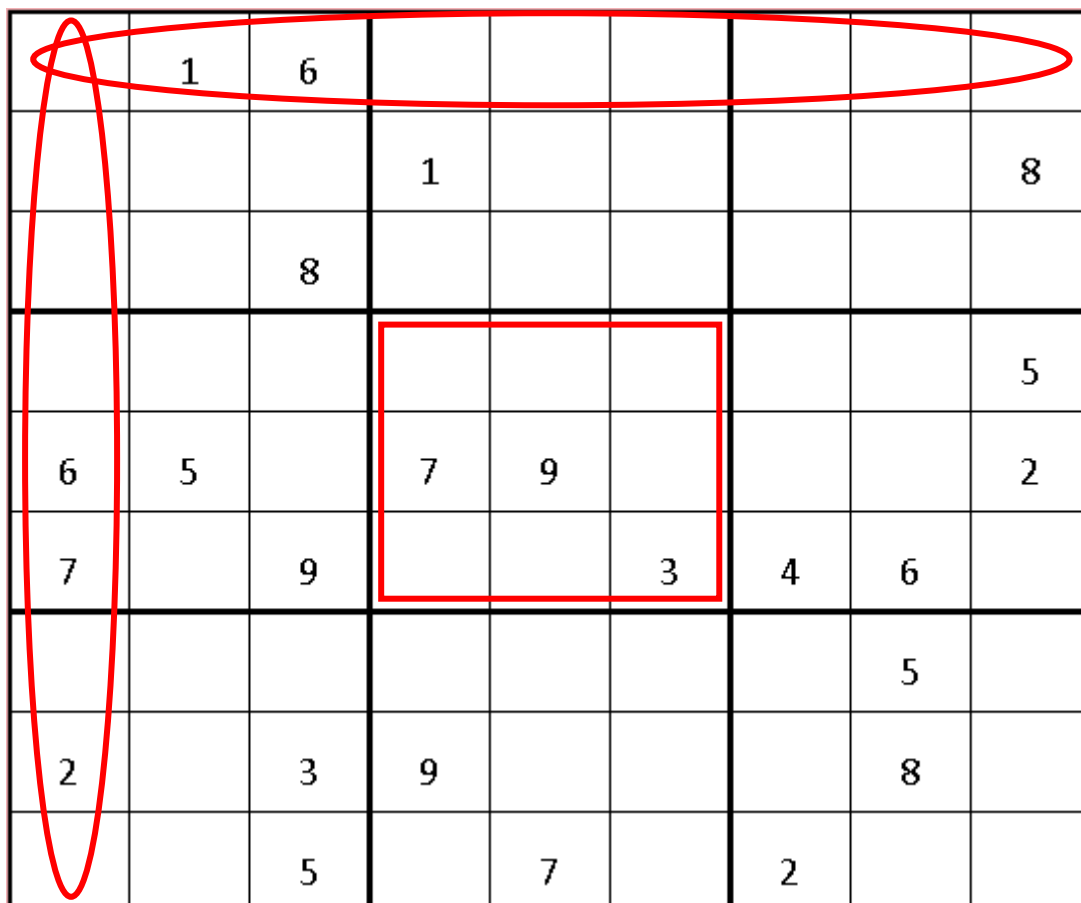
Kelompok: 9

Kelas: LB08

Anggota:

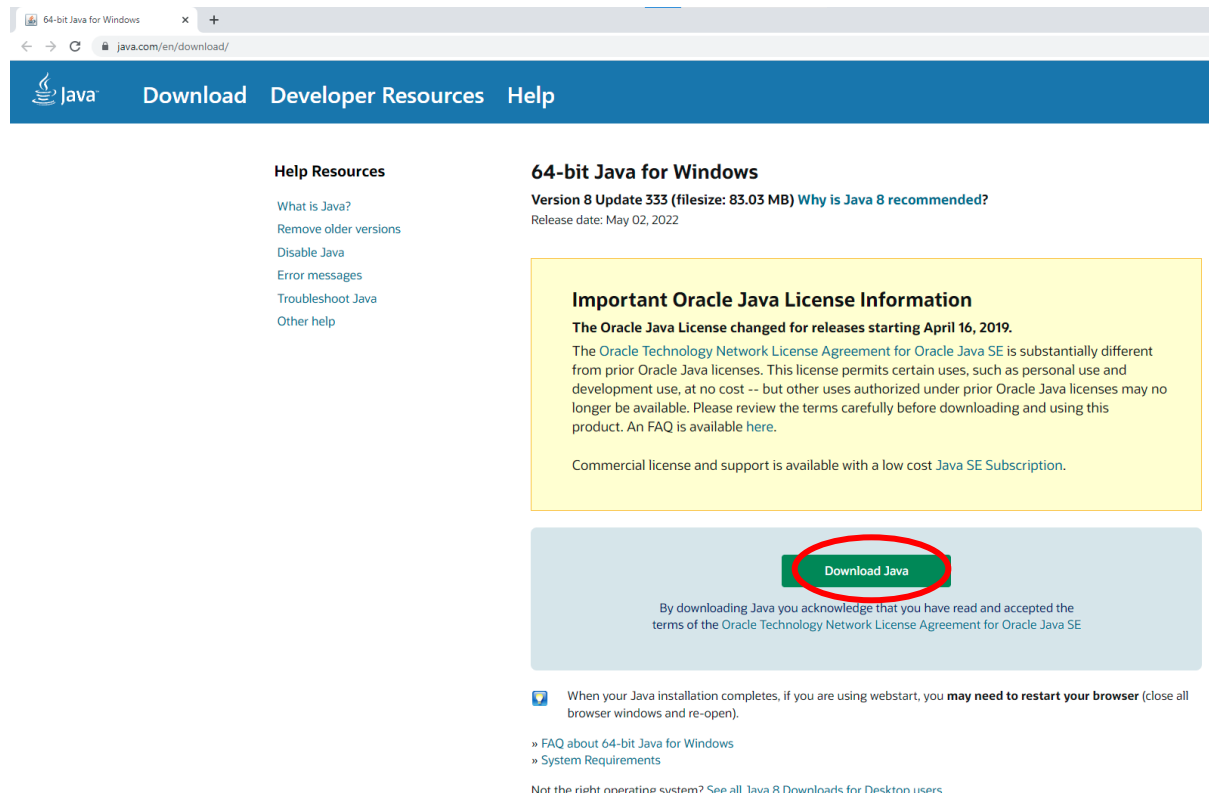
1. Rhenaldy – 2440006974 – Game Logic
2. Ladysa Stella Karenza – 2440007724 – Interface
3. Gilbert Nathaniel – 2401954893 – Code

Project yang dibuat oleh kami Kelompok 9 adalah permainan Sudoku. Sudoku adalah sebuah permainan puzzle atau teka-teki dimana pemain harus memasukkan angka 1 hingga 9 pada sebuah papan dengan cells sebanyak 9 x 9 dengan sebagian angka pada papan ditampilkan diawal permainan. Peraturan dari permainan Sudoku adalah setiap baris, kolom, dan kotak (area yang terdiri dari cells sebanyak 3 x 3 sehingga satu papan memiliki 9 kotak) tidak boleh memiliki angka yang berulang. Berikut contoh baris, kolom, dan kotak tersebut.



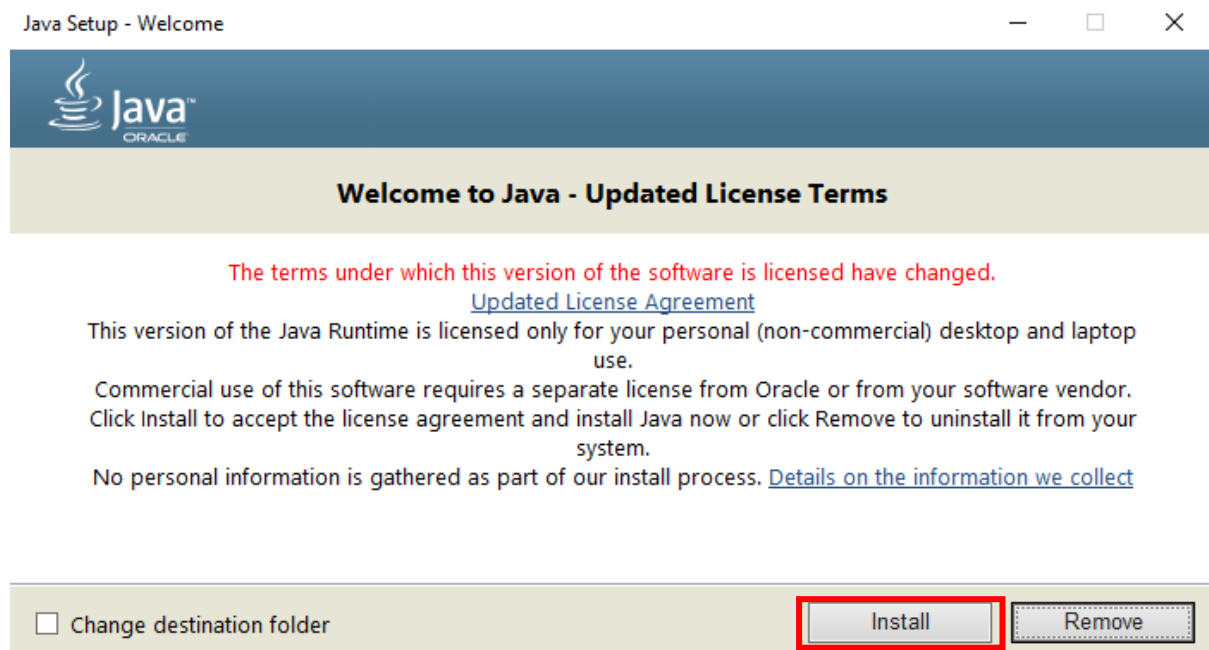
	1	6						
			1					8
		8						
								5
6	5		7	9				2
7		9			3	4	6	
							5	
2		3	9				8	
		5		7		2		

Sebelum memainkan Sudoku, pertama-tama pengguna harus terlebih dahulu mengunduh atau men-download Java 8. Java 8 installer dapat diunduh atau di-download terlebih dahulu melalui website berikut. <https://java.com/en/download/>.



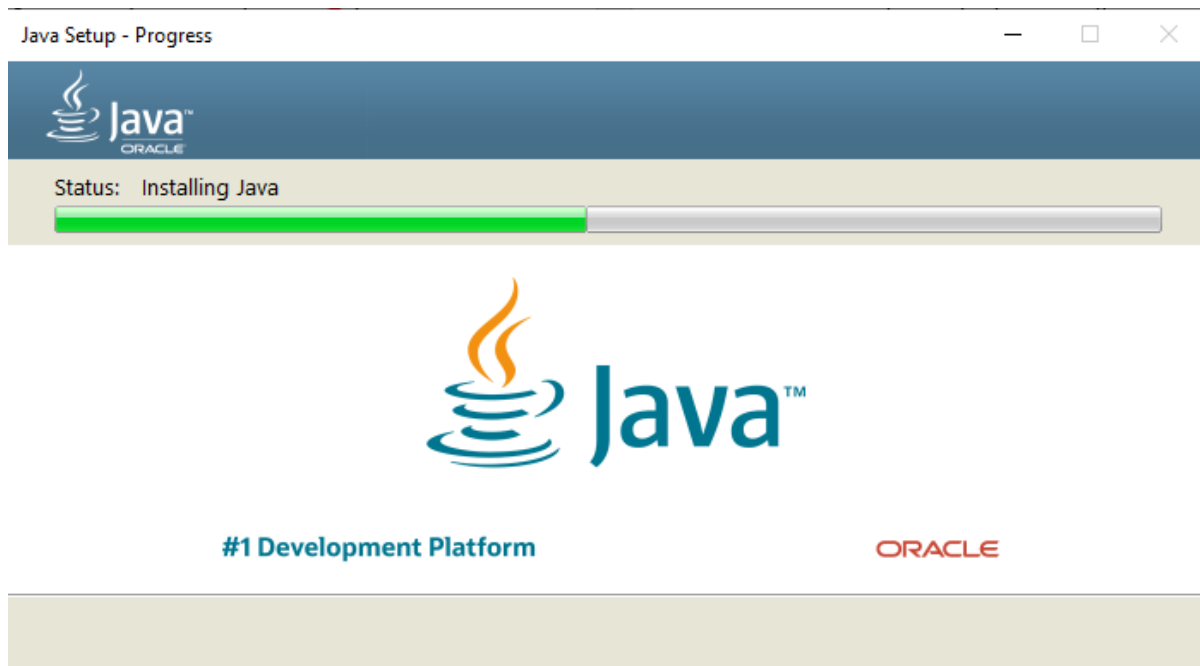
The screenshot shows the Java 8 download page. The browser address bar shows 'java.com/en/download/'. The page has a blue header with the Java logo and navigation links: 'Download', 'Developer Resources', and 'Help'. On the left, there's a 'Help Resources' section with links like 'What is Java?', 'Remove older versions', 'Disable Java', 'Error messages', 'Troubleshoot Java', and 'Other help'. The main content area is titled '64-bit Java for Windows' and 'Version 8 Update 333 (filesize: 83.03 MB) Why is Java 8 recommended?'. It includes a release date of May 02, 2022. A yellow box contains 'Important Oracle Java License Information' stating that the license changed starting April 16, 2019. Below this, a green 'Download Java' button is circled in red. A light blue box contains a disclaimer about the license agreement. At the bottom, there are links for FAQ and System Requirements, and a note about restarting the browser.

Kemudian, jalankan installer Java 8. Berikut adalah tampilan yang keluar ketika menjalankan installer Java 8. Jika ingin mengganti tujuan folder dari Java 8, centanglah kotak dibawah kiri (Optional). Kemudian, tekan Install

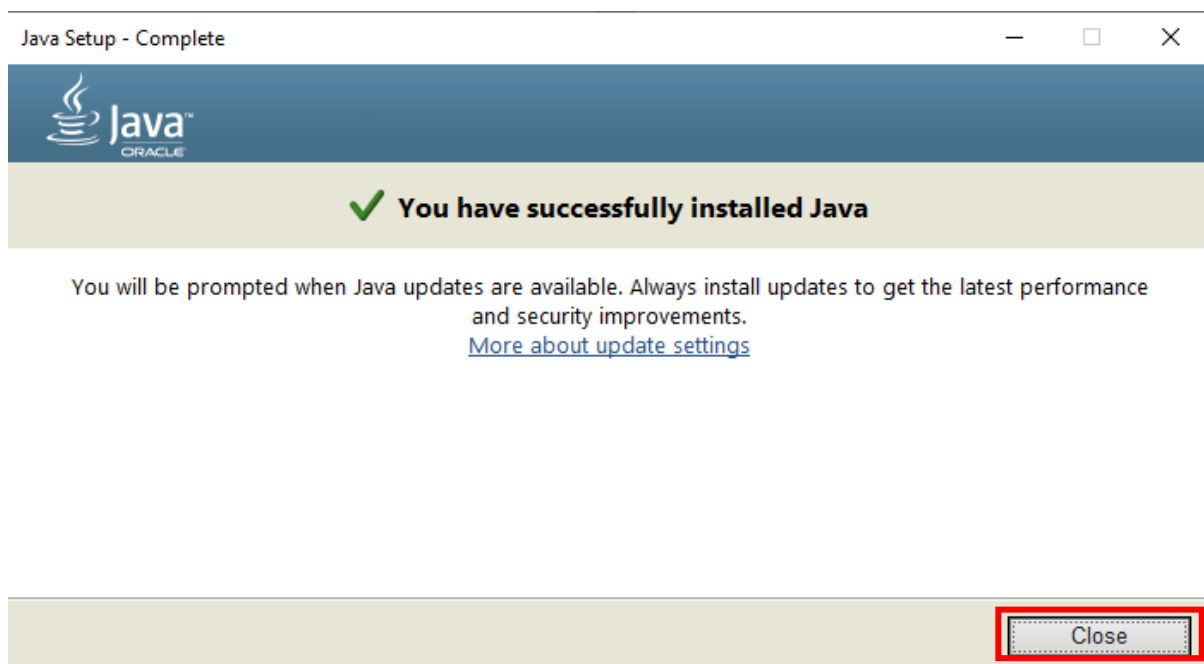


The screenshot shows the 'Java Setup - Welcome' window. The title bar says 'Java Setup - Welcome'. The window has a blue header with the Java logo and 'ORACLE'. The main content area is titled 'Welcome to Java - Updated License Terms'. It contains text about the updated license terms, stating that the terms have changed and that the software is licensed for personal (non-commercial) desktop and laptop use. It also mentions that commercial use requires a separate license. At the bottom, there's a checkbox for 'Change destination folder' and two buttons: 'Install' and 'Remove'. The 'Install' button is highlighted with a red rectangle.

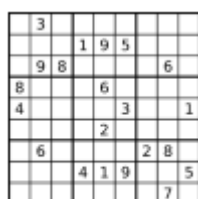
Setelah menekan Install, berikut tampilan yang akan muncul. Tunggulah hingga installer selesai.



Berikut tampilan ketika Java 8 telah selesai di-download. Tekanlah close.



Kemudian, unduhlah Sudoku.exe yang memiliki tampilan *icon* seperti berikut. Sudoku.exe dapat diletakkan dimana saja oleh pengguna, Sudoku.exe dapat dijalankan asalkan Java 8 telah ter-install.



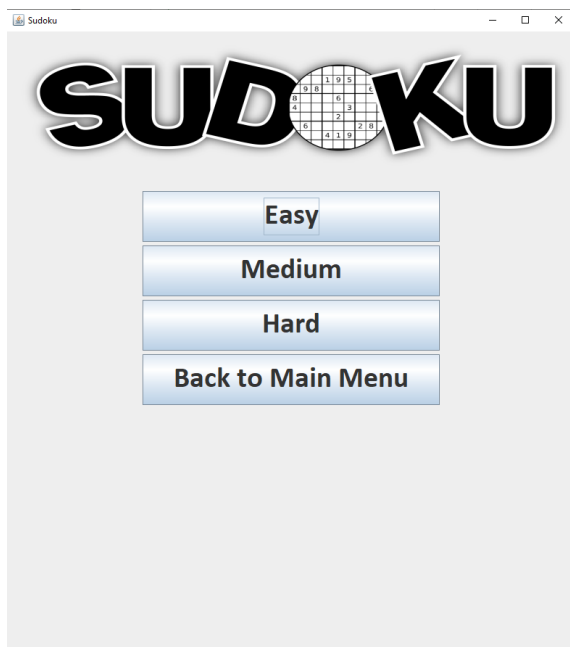
Sudoku.exe

Setelah Sudoku.exe telah selesai diunduh, pengguna dapat menjalankan aplikasi Sudoku.exe dengan tujuan memainkan permainan Sudoku.

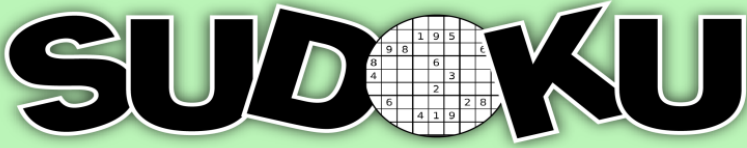
Berikut tampilan Main Menu Sudoku.exe ketika dijalankan.



Jika pengguna ingin memulai permainan, pengguna dapat menekan tombol "New Game", yang akan memberikan pilihan tingkat kesusahan permainan. Dalam Sudoku.exe, terdapat tingkat kesusahan "Easy" (mudah), "Medium" (sedang), "Hard" (susah). Jika pengguna menekan tombol "Quit", maka Sudoku.exe akan tutup. Berikut tampilan Sudoku.exe ketika pengguna menekan tombol "New Game".



Ketika pengguna menekan tombol "Easy", "Medium", atau "Hard", Sudoku.exe akan menampilkan permainan Sudoku menurut tingkat kesusahannya. Jika pengguna menekan tombol "Back to Main Menu", pengguna akan dibawa ke "Main Menu" Sudoku.exe. Berikut tampilan dari Sudoku.exe ketika pengguna menekan atau memilih permainan "Easy", "Medium", dan "Hard".



6			2	3	7		8	9
8	2	7		4	9		6	
4				6		7	1	2
1	5	4	3			6	9	7
3	7		4	9		5	2	
	9		6	7	5	1	3	4
5	4	1			3	9	7	
			9		4	8	5	3
9	8	3	7	5		2	4	1

1

2

3

4

5

6

7

8

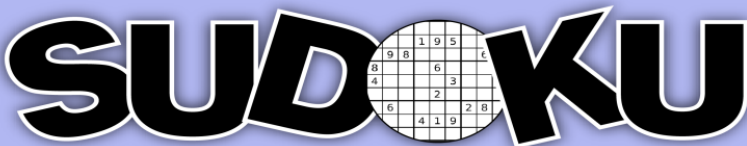
9

X

Back

Restart Game

Check



9					6		7	8
5	2		1	7		3	4	
8	3		4	5		1	6	
						8		7
		2			1		3	
4	9			8		2		
2		5			3			6
3	6	8	9		2		5	
	7	9	8			4	2	

1

2

3

4

5

6

7

8

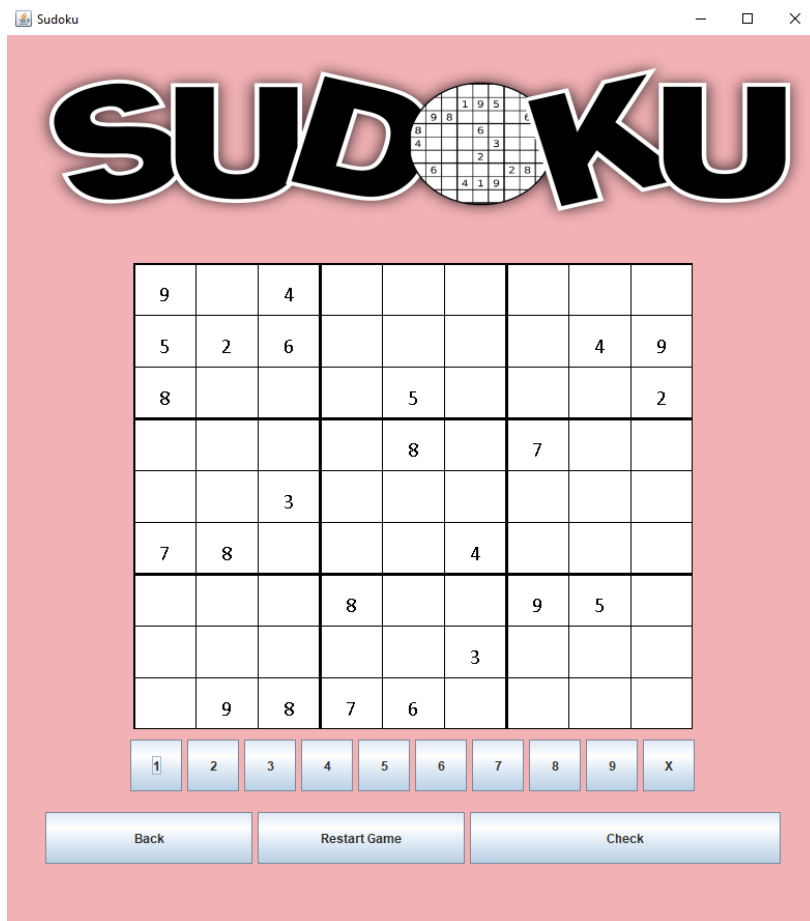
9

X

Back

Restart Game


Check



Jika pengguna memilih tingkat kesusahan “Easy”, maka permainan Sudoku akan ditampilkan dengan *background* warna hijau. Jika pengguna memilih tingkat kesusahan “Medium”, maka permainan Sudoku akan ditampilkan dengan *background* warna biru. Sedangkan, jika pengguna memilih tingkat kesusahan “Hard”, maka permainan Sudoku akan ditampilkan dengan *background* warna merah.

Dibawah papan Sudoku, terdapat tombol 1 hingga 9 dengan tombol X diakhir. Ketika pengguna ingin menulis sebuah angka di papan, pengguna perlu menekan cell yang diinginkan terlebih dahulu. Cell yang dipilih atau ditekan akan berganti warna menjadi biru muda. Kemudian, tekanlah angka yang diinginkan, misalkan angka 1. Jika pengguna ingin menghapus angka yang telah ditulis oleh pengguna, maka pengguna perlu menekan cell yang ingin dihapus, kemudian tekanlah tombol X untuk menghapus angka pada cell tersebut. Angka yang dapat dihapus hanyalah angka-angka yang ditulis oleh pengguna. Angka-angka yang ditampilkan dari awal permainan tidak dapat dihapus oleh pengguna.

Berikut contoh tampilan ketika pengguna ingin menulis angka. Ketika pengguna menekan tombol angka (seperti pada contoh tombol 1), dapat dilihat cell tersebut diisi oleh angka 1, tetapi seluruh angka 1 yang terdapat pada papan Sudoku ikut ter-*highlight*. *Highlight* tersebut dapat ditampilkan juga dengan cara menekan angka mana pun pada papan. *Highlight* tersebut membantu pengguna dalam melihat letak-letak angka tertentu agar tidak ada lebih dari satu angka yang sama dalam kolom, baris, dan atau kotak.


Sudoku
— □ ×

SUDOKU

8	1	6		4	7	3	5	9
			1	5	9			8
			3		8	1	7	2
5	2		4	8	1		9	
	6		5		2	8	3	1
		8	6		3	5	2	4
9	4	1	7	2	5	6		
3	7		8	1	6	9	4	5
6	8	5	9			2	1	

1

2

3

4

5

6

7

8


9

X

Back

Restart Game

Check


Sudoku
— □ ×

SUDOKU

8	1	6		4	7	3	5	9
			1	5	9			8
			3		8	1	7	2
5	2		4	8	1		9	
	6		5		2	8	3	1
1		8	6		3	5	2	4
9	4	1	7	2	5	6		
3	7		8	1	6	9	4	5
6	8	5	9			2	1	

1

2

3

4

5

6

7

8

9

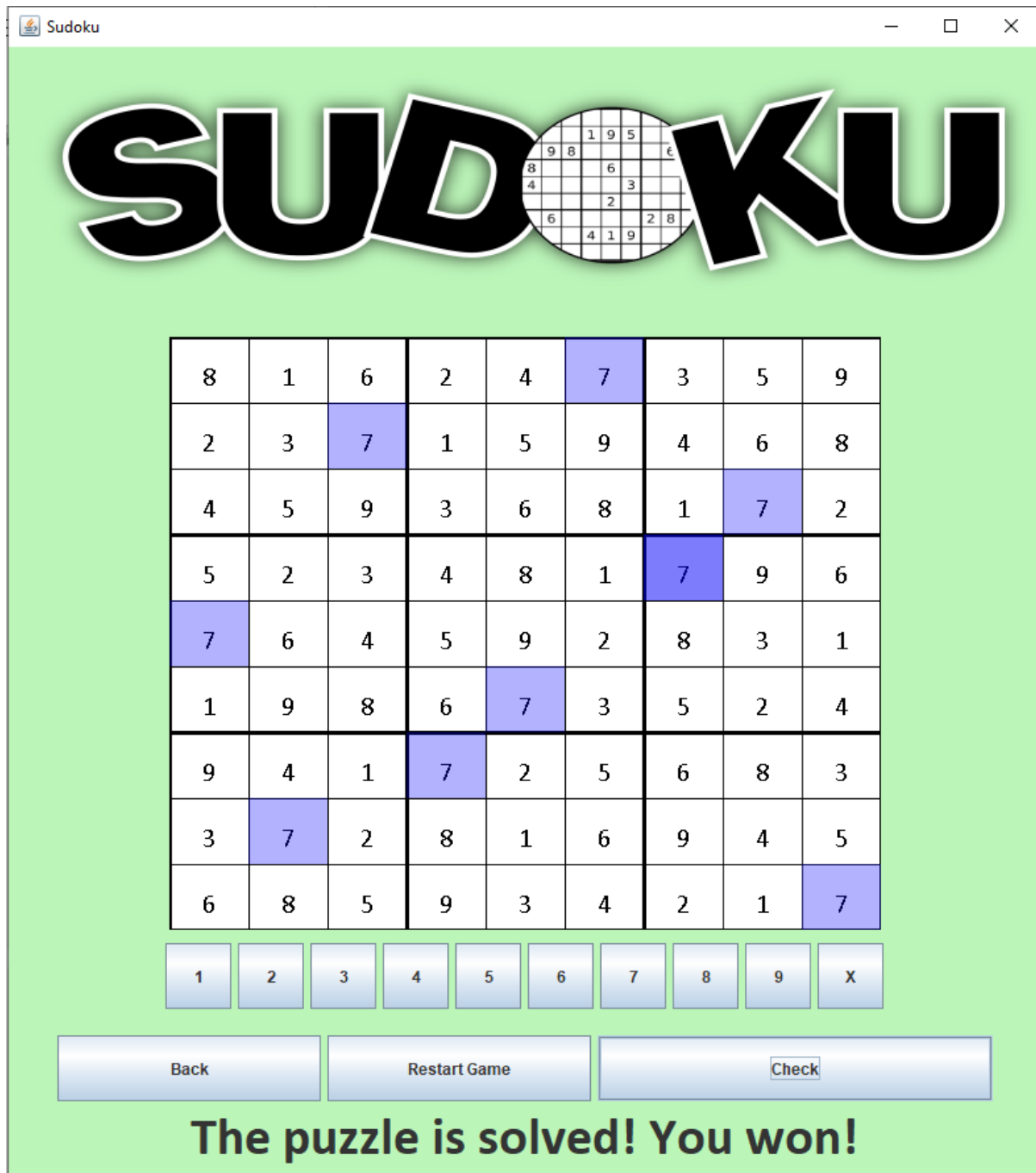
X

Back

Restart Game

Check

Tombol “Back” dibawah kiri tampilan akan membawa pengguna balik ke Tampilan pemilihan tingkat kesusahan permainan. Tombol “Restart Game” disebelah kanan tombol “Back” akan menampilkan ulang papan Sudoku seperti awal. Tombol “Check” disebelah kanan tombol “Restart Game” akan memeriksa jawaban pengguna ketika ditekan. Jika papan Sudoku sudah diselesaikan dengan benar oleh pengguna, maka akan muncul tulisan “The puzzle is solved! You won!” dibawah dan pengguna akan dibawa kembali ke “Main Menu”. Jika papan Sudoku tidak diselesaikan dengan benar, maka akan muncul tulisan “The puzzle is still not solved!” dibawah. Berikut contoh tampilan pengguna ketika permainan sudoku telah diselesaikan dengan benar dan tombol “Check” telah ditekan oleh pengguna.



Aspek-aspek Object Oriented Programming (OOP) pada project Sudoku kami kelompok 9 terdiri dari:

1. Encapsulation

Encapsulation pada project Sudoku kami dapat dilihat pada atribut-atribut class seperti Puzzle dan Mode yang menggunakan private access modifier,

Berikut tampilan code untuk class Puzzle dengan atribut-atribut yang menggunakan private access modifier. Penggunaan private access modifier dilakukan dengan tujuan agar atribut-atribut tersebut tidak dapat diakses semena-mena dari luar classnya. Jika ingin membuat instance atau memanggil nilai atribut tertentu, dapat digunakan constructor atau getters yang ada dalam class Puzzle.

```
public class Puzzle {
    private String[][] board;
    private boolean[][] mutable;
    private final int row; // 9
    private final int column; // 9
    private final int width; // 3
    private final int height; // 3
    private final String[] numbers; // 1-9

    // this constructor creates an empty sudoku puzzle
    public Puzzle() {
        this.row = 9;
        this.column = 9;
        this.width = 3;
        this.height = 3;
        this.numbers = new String[] { "1", "2", "3", "4", "5", "6", "7", "8", "9" };
        this.board = new String[9][9];
        this.mutable = new boolean[9][9];
        initializeBoard();
        initializeMutableSlots();
    }

    public int getRow() {
        return this.row;
    }

    public int getColumn() {
        return this.column;
    }

    public int getWidth() {
        return this.width;
    }

    public int getHeight() {
        return this.height;
    }

    public String[] getNumbers() {
        return this.numbers;
    }
}
```

2. Abstraction

Pada project Sudoku kami, abstraction diterapkan pada class Mode, yang memiliki atribut berupa Puzzle dan dua abstract method yang bertujuan menentukan warna *background* tiap mode / tingkat kesusahan dan menentukan jumlah angka yang ditampilkan diawal. Berikut tampilan code abstract class Mode.

```
import java.awt.Color;

public abstract class Mode {
    private Puzzle puzzle;

    public Mode() {
        super();
        this.puzzle = new SudokuGenerator().generateRandomSudoku(this.getVisibilityPercentage());
    }

    public Puzzle getPuzzle() {
        return puzzle;
    }

    public abstract Color getBackgroundColour(); // this method is created to get the colour for every mode
    public abstract double getVisibilityPercentage();
    // this method is created to determine the total amount of numbers to be displayed to the user
}
```

3. Inheritance

Pada project Sudoku kami, terdapat tiga class yang merupakan child class atau turunan dari abstract class Mode, yaitu Easy, Medium, dan Hard.

```
import java.awt.Color;

// Child class of Mode
public class Easy extends Mode {
    public Easy() {
        super();
    }

    @Override
    public Color getBackgroundColour() {
        return new Color(187, 245, 184); // easy mode -> pastel green
    }

    @Override
    public double getVisibilityPercentage() {
        return 0.7; // about 70% of numbers on the sudoku is shown to
    }
}

import java.awt.Color;

// Child class of Mode
public class Medium extends Mode {
    public Medium() {
        super();
    }

    @Override
    public Color getBackgroundColour() {
        return new Color(177, 183, 242); // medium mode -> pastel blue
    }

    @Override
    public double getVisibilityPercentage() {
        return 0.5; // about 50% of numbers on the sudoku is shown to t
    }
}

import java.awt.Color;

// Child class of Mode
public class Hard extends Mode {
    public Hard() {
        super();
    }

    @Override
    public Color getBackgroundColour() {
        return new Color(242, 177, 181); // hard mode -> pastel red
    }

    @Override
    public double getVisibilityPercentage() {
        return 0.3; // about 30% of numbers on the sudoku is shown to
    }
}
```

4. Polymorphism

Polymorphism pada project Sudoku kami dapat dilihat pada class Mode dan child classesnya. Dari class Mode diturunkan dua abstract method, yaitu getBackgroundColour dan getVisibilityPercentage.

Pada class Easy, method getbackgroundColour mengembalikan warna hijau pastel. Pada class Medium, method tersebut mengembalikan warna biru pastel. Sedangkan pada class Hard, method tersebut mengembalikan warna merah pastel.

Pada class Easy, method getVisibilityPercentage menentukan jumlah angka yang ditampilkan diawal permainan Sudoku sebanyak 70%, dengan sisa 30% diisi oleh pengguna agar permainan selesai. Pada

class Medium, jumlah angka yang ditampilkan pada awal permainan Sudoku adalah sebanyak 50%. Sedangkan pada class Hard, jumlah angka tersebut adalah sebanyak 30%.

References:

<https://commons.wikimedia.org/wiki/File:Sudoku-by-L2G-20050714modif.svg>

<https://github.com/mattnenterprise/Sudoku>