**LAPORAN RESMI**

**MODUL IV  
INTERNAL FRAME & TABBED PANE  
PEMROGRAMAN VISUAL**



**NAMA : DONY EKA OCTAVIAN PUTRA**

**N.R.P : 230441100041**

**DOSEN : FIRMANSYAH ADIPUTRA, ST., M.CS.  
ASISTEN : MUHAMMAD IQBAL FIRMASNYAH**

**TGL PRAKTIKUM : Rabu, 23 Oktober 2024**

**Disetujui : 29 Oktober 2024**

**Asisten**

**MUHAMMAD IQBAL FIRMASNYAH 21.04.411.00084**

**LABORATORIUM BISNIS INTELIJEN SISTEM**

**PRODI SISTEM INFORMASI**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS TRUNOJOYO MADURA**

# PENDAHULUAN

## Latar Belakang

Dalam konteks pengembangan aplikasi Java menggunakan NetBeans, Palette menu adalah sumber daya yang sangat berharga. Palette menyediakan berbagai komponen GUI yang dapat digunakan untuk merancang antarmuka pengguna dengan mudah. Di antara bidang yang ada dalam Palette, Swing Containers adalah salah satu yang sangat penting. Komponen-komponen seperti JInternalFrame dan JTabbedPane dalam bidang Swing Containers membantu dalam merancang tampilan yang terstruktur dan interaktif. JInternalFrame memungkinkan kita untuk membuat jendela internal dalam aplikasi, sementara JTabbedPane memungkinkan kita mengatur konten dalam tab yang dapat diakses dengan cepat.

Pemrograman Swing adalah fondasi pengembangan antarmuka pengguna Java. Ini memberikan toolkit yang kuat untuk membuat aplikasi desktop yang kaya fitur dan interaktif. NetBeans menyederhanakan proses pengembangan antarmuka pengguna Swing dengan menyediakan alat GUI builder yang intuitif. Dengan alat ini, Anda dapat secara visual menarik dan mengatur komponen GUI, mengatur properti, dan menghubungkan event tanpa harus menulis banyak kode secara manual.

Dalam laporan ini, kami akan menjelaskan konsep dasar penggunaan JInternalFrame dan JTabbedPane, dan juga akan menunjukkan cara menggunakan alat GUI builder dalam NetBeans untuk membuat halaman antarmuka sederhana yang memanfaatkan Internal Frame dan Tabbed Pane. Ini akan membantu Anda memahami cara mengatur konten dalam aplikasi Anda dengan baik dan membangun antarmuka pengguna yang mudah dinavigasi serta menarik bagi pengguna.

## Tujuan

* Mahasiswa mampu memahami konsep pemrograman Swing.
* Mahasiswa mampu membuat halaman sederhana menggunakan komponen swing dibantu tool gui builder.
* Mahasiswa mampu menggunakan internal frame dan tabbed pane.

# DASAR TEORI

## JTabbedPane

Pada praktikum kali ini akan dibahas mengenai penggunaan dari tabbed pane dan internal frame.

Kelas JTabbedPane merupakan komponen grafis penting yang digunakan untuk Menyusun dan mengelompokkan berbagai komponen Java lain ke dalam lapisan-lapisan (layers). Ketika obyek kelas JTabbedPane dimunculkan, hanya satu lapisan (one layer) yang ditampilkan. Untuk berpindah dari satu lapisan ke lapisan lain, Anda cukup mengklik pada tab yang diinginkan.

Kelas JTabbedPane mempunyai tiga buah konstruktor. Untuk mengkonstruksi obyek tabbed pane, Anda dapat menggunakan salah satu dari dua konstruktor yang paling sering digunakan berikut ini.

### Konstruktor Kelas JTabbedPane

|  |  |
| --- | --- |
| **Konstruktor** | **Keterangan** |
| JTabbedPane() | Mengkonstruksi obyek tabbed pane dengan menggunakan tab default JTabbedPane.TOP. |
| JTabbedPane(int tapPlacement) | Mengkonstruksi obyek tabbed pane dengan penempatan tap sesuai dengan nilai argumen dari parameter tapPlacement. |

Anda dapat memberi argumen parameter tapPlacement untuk konstruktor kedua kelas JTabbedPane dengan salah satu konstanta bernilai integer yaitu JTabbedPane.TOP, JTabbedPane.BOTTOM, JTabbedPane.LEFT atau JTabbedPane.RIGHT. Kelas JTabbedPane mempunyai banyak metoda untuk mengkreasi tampilan obyek tabbed pane. Berikut ini adalah daftar beberapa metoda kelas JTabbedPane yang sering digunakan.

### Metoda Kelas JtabbedPane

|  |  |
| --- | --- |
| **Metoda** | **Keterangan** |
| setIconAt (int index) | Mengatur ikon di index tap yang ditentukan. |
| getTapCount() | Mengembalikan jumlah tap dari obyek tabbed pane. |
| setTapPlacement(int tapPlacement) | Mengatur penempatan tap-tab dari obyek tabbed pane. |
| setTitleAt(int index, String title) | Mengatur title tap di index tap yang telah ditetapkan. |
| setToolTipTextAt(int index, String toolTipText) | Mengatur tool tip text di index tap yang ditetapkan. |
| setSelectedComponent( Component c) | Mengatur komponen yang dipilih  untuk obyek tabbed pane |
| setSelectedIndex | Mengatur index yang dipilih saat ini untuk obyek tabbed pane. |
| indexOfComponent(Component c) | Mengembalikan index tap untuk  komponen yang ditetapkan. |
| indexOfTab(Icon icon) | Mengembalikan index tap untuk ikon yang ditetapkan. |
| indexOfTab(String title) | Mengembalikan index tap untuk title yang ditetapkan. |
| remove(int index) | Menghilangkan tab beserta komponennya pada index yang telah ditetapkan. |
| removeTabAt(int index) | Menghilangkan tab di index yang ditentukan. |
| setBackgroundAt(int index, Color color) | Mengatur background di index yang ditentukan oleh warna argument color. |
| setForegroundAt(int index, Color color) | Mengatur foreground di index yang  ditentukan oleh warna argumen color. |
| setToolTipTextAt(int index, String toolTiptext) | Mengatur ulang tool tip text di index  yang telah ditentukan. |

### JInternalFrame

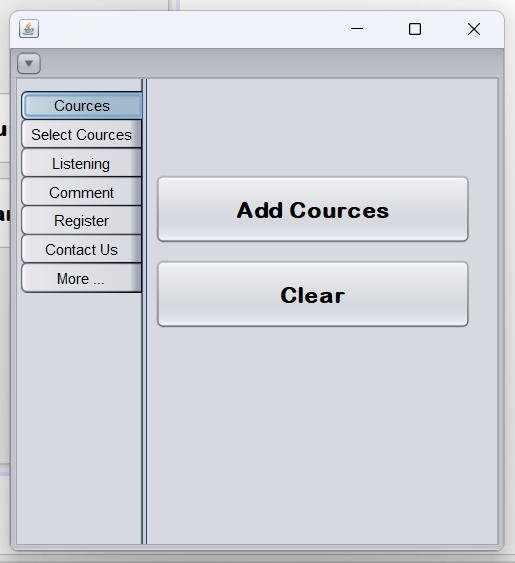
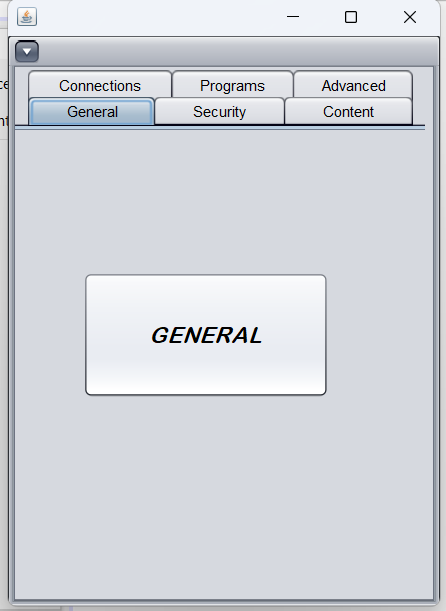
Internal frame digunakan ketika membutuhkan beberapa objek window yang terbuka dalam satu aplikasi. JInternalFrame ditambahkan ke dalam JDesktopPane.

Aturan dalam menggunakan JInternalFrame :

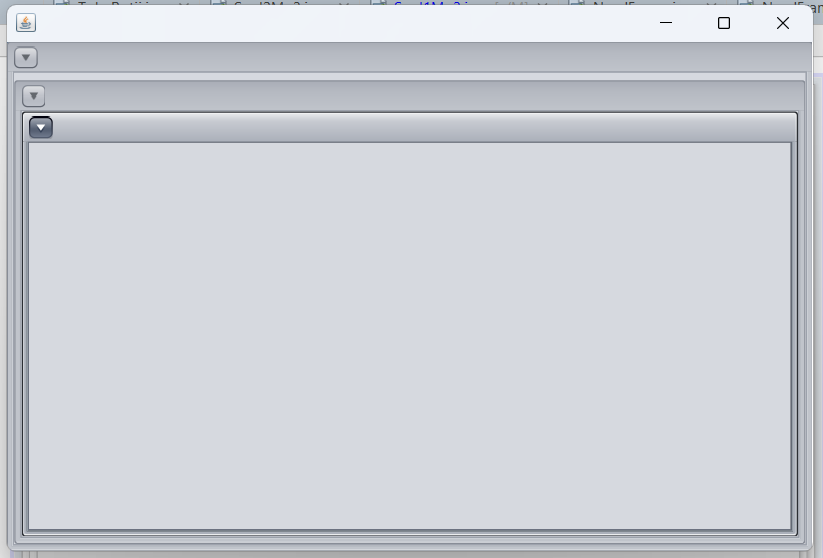
* JInternalFrame harus diberikan size menggunakan metode setSize(), pack() atau setBounds(), jika tidak diberikan size maka JinternalFrame akan berukuran nol dan tidak akan pernah terlihat.
  + - JInternalFrame harus diberikan lokasi menggunakan metode setLocation() atau setBounds(), jika hal ini tidak diberikan maka JInternalFrame akan terletak pada koordinat(0,0).
    - JInternalFrame harus ditambahkan ke dalam Container biasanya JDesktopPane, jika hal ini tidak dilakukan maka JInternalFrame tidak akan terlihat.
    - JInternalFrame harus memanggil metode show() atausetVisible(true) agar bisa terlihat.
    - JInternalFrame hanya bisa menggunakan event-event internal frame bukan window event.
    - JInternalFrame hanya bisa menggunakan dialog berupa JOptionPane dan JInternalFrame bukan JDialog.

## Latihan Praktikum

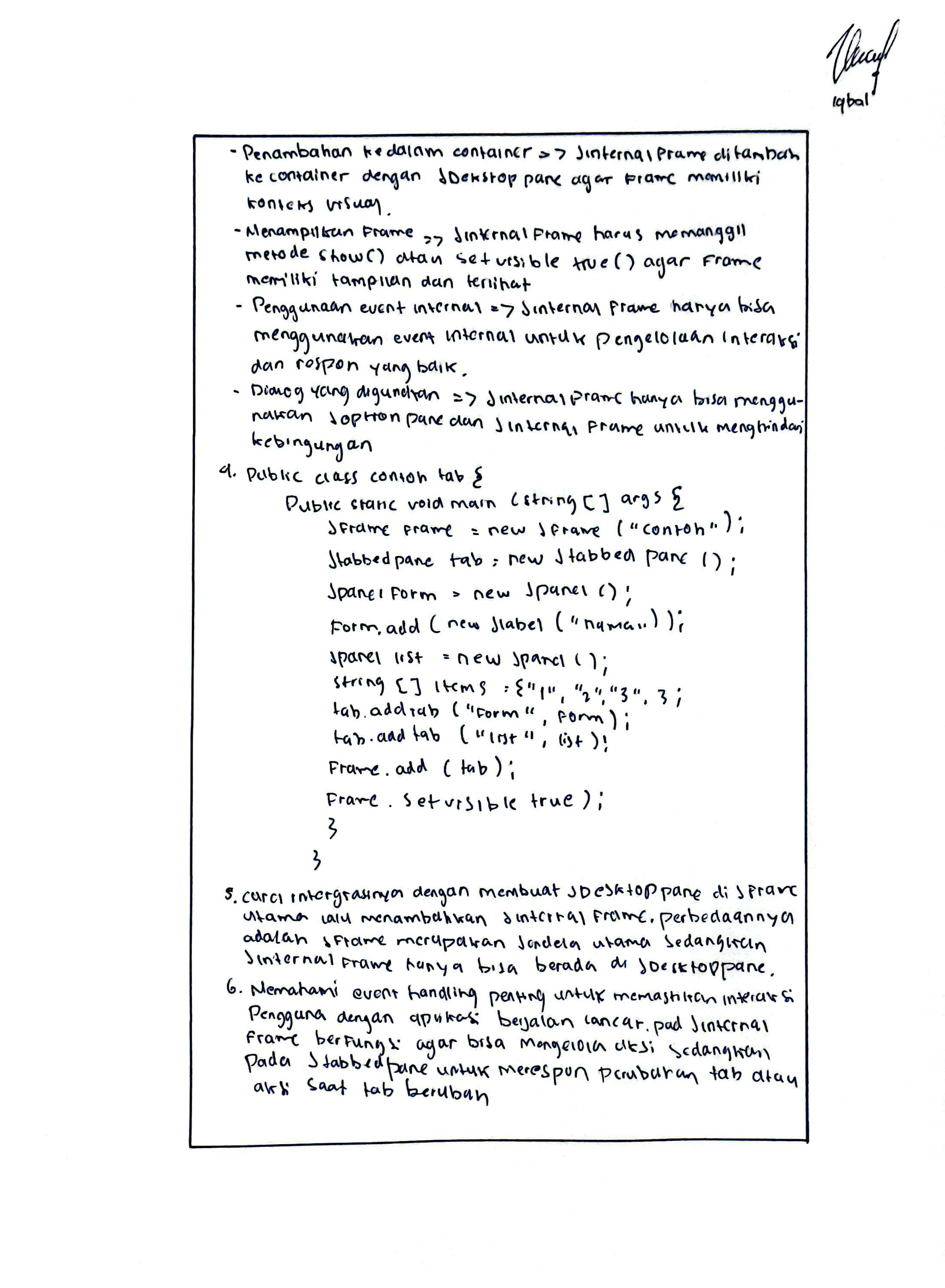
### Penggunaan JTabbedPane



### Penggunaan JInternalFrame



# IMPLEMENTASI

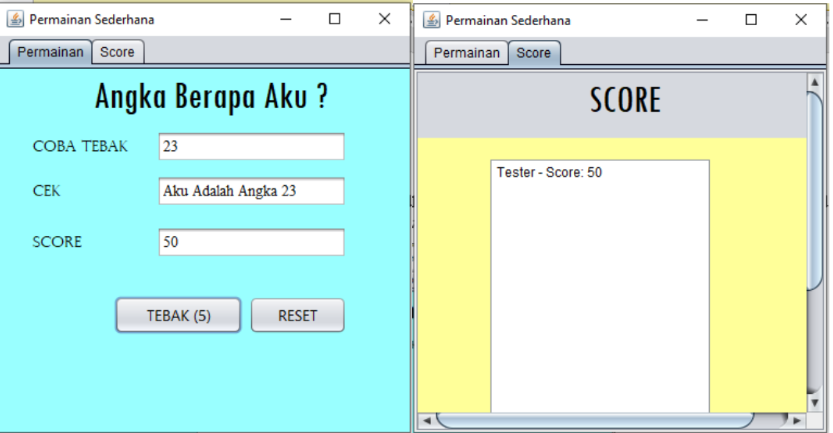


# IMPLEMENTASI

## Tugas Praktikum

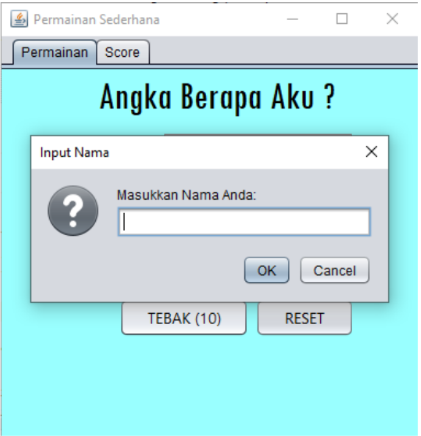
### Soal Nomor 1

Buatlah Program Permainan Tebak Angka menggunakan Tabbed Pane seperti contoh dibawah ini



Alur Program Sebagai Berikut :

1. Target berupa angka acak dari sistem 1 – 100.
2. Ketika user menginputkan angka tebakan dan hasilnya lebih kecil atau lebih besar dari target maka akan muncul angka lebih besar atau lebih kecil.
3. Ketika tebakan user benar atau sama dengan target, maka akan muncul keterangan ”Anda Berhasil” dan menginputkan nama dengan showInputDialog lalu data akan dikirim ke List pada Tab Score.



1. User punya kesempatan menebak 10x, dan jika user menebak lebih dari 10x maka akan muncul keterangan ”Anda Gagal” dan program akan reset secara otomatis.
2. Score didapat dari sisa tebakan user dikali 10.
3. Gunakan List untuk menampilkan riwayat score permainan berisi nama dan jumlah score nya.
4. Terapkan Layout

## Source Code

### Soal Nomor 1

RESET

 tebak.setText("");

        cek.setText("");

        score.setText("0");

        TEBAK.setText("Tebak(10)");

TEBAK

System.out.println(Tebakan);

        try {

            int angka2 = Integer.parseInt(tebak.getText());

            if (Tebakan > angka2) {

            kesempatan -= 1;

            TEBAK.setText("Tebak (" + kesempatan + ")");

            cek.setText("Masukkan Angka Lebih Besar");

            tebak.setText("");

            score.setText("0");

            JOptionPane.showMessageDialog(this, "TEBAKANMU SALAH", "PESAN PEMBERITAHUAN", JOptionPane.INFORMATION\_MESSAGE);

        if (kesempatan == 0) {

            resetGame();

            }

        } else if (Tebakan < angka2) {

            cek.setText("Masukkan Angka Lebih Kecil");

            kesempatan -= 1;

            TEBAK.setText("Tebak (" + kesempatan + ")");

            tebak.setText("");

            score.setText("0");

            JOptionPane.showMessageDialog(this, "TEBAKANMU SALAH", "PESAN PEMBERITAHUAN", JOptionPane.INFORMATION\_MESSAGE);

        if (kesempatan == 0) {

            resetGame(); // Metode untuk mereset permainan

            }

        // Periksa jika tebakan benar

        }else if (Tebakan == angka2) {

            cek.setText(String.valueOf(Tebakan));

            int hasil = kesempatan \* 10; // Hitung skor berdasarkan sisa kesempatan

            score.setText(String.valueOf(hasil));

            JOptionPane.showMessageDialog(this, "YEYYY TEBAKANMU BENAR", "INFORMASI", JOptionPane.INFORMATION\_MESSAGE);

        String nama; // Deklarasi variabel nama

        while (true) { // Loop untuk memeriksa nama

            nama = JOptionPane.showInputDialog(this, "Masukkan Namamu", "Nama player", JOptionPane.INFORMATION\_MESSAGE);

            // Periksa jika nama valid dan tidak kosong

            if (nama != null && !nama.trim().isEmpty()) {

                boolean nameExists = false;

                // Periksa jika nama sudah ada di skor

                for (int i = 0; i < index; i++) {

                    if (data\_score[i][0].equalsIgnoreCase(nama)) {

                        nameExists = true;

                        break;

                    }

                }

                if (!nameExists) {

                    // Simpan skor jika nama unik

                    if (index < data\_score.length) {

                        data\_score[index][0] = nama; // Simpan nama

                        data\_score[index][1] = String.valueOf(hasil); // Simpan skor

                        index++; // Increment index setelah menambahkan data

                    }

                    // Bersihkan input

                    tebak.setText("");

                    cek.setText("");

                    score.setText("0");

                    TEBAK.setText("Tebak(10)");

                    // Bersihkan StringBuilder sebelum menambahkan data baru

                    score\_border.setLength(0);

                    // Tambahkan data baru ke score\_border

                    for (int i = 0; i < index; i++) {

                        score\_border.append("Name: ").append(data\_score[i][0]).append(", Score: ").append(data\_score[i][1]).append("\n");

                    }

                    // Perbarui area teks dengan score\_border yang baru

                    AREA.setText(score\_border.toString());

                    // angka acak baru untuk putaran berikutnya

                    Random random = new Random();

                    Tebakan = random.nextInt(100) + 1; // Rentang contoh: 1 hingga 100

                    kesempatan = 10; // Reset kesempatan

                    TEBAK.setText("Tebak (" + kesempatan + ")");

                    break; // Keluar dari loop jika nama valid

                } else {

                    JOptionPane.showMessageDialog(this, "Nama sudah digunakan! Silakan masukkan nama yang lain.", "Peringatan", JOptionPane.WARNING\_MESSAGE);

                }

                }else {

                JOptionPane.showMessageDialog(this, "Nama tidak boleh kosong!", "Peringatan", JOptionPane.WARNING\_MESSAGE);

              }

            }

          }

         } catch (NumberFormatException e) {

           JOptionPane.showMessageDialog(this, "TEBAKAN HARUS BERUPA ANGKA", "PESAN PERINGATAN", JOptionPane.ERROR\_MESSAGE);

       }

    }

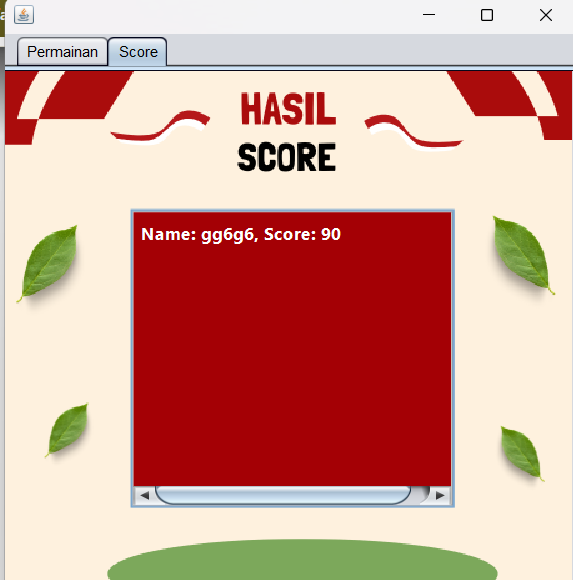
## Hasil

### Soal Nomor 1

* **Tap ke 1**



* **Tap ke 2**



## Penjelasan

### Soal Nomor 1

Program ini merupakan kelanjutan dari aplikasi berupa game Tebak Angka. User akan menebak menebak satu angka dari 1 hingga 100. ketika aplikasi dibuka maka akan muncul pop-up dengan panduan cara bermain. Game ini memiliki dua tab yaitu satu untuk permainan dan satu lagi untuk menampilkan skor.

Pada tab permainan, user akan mencoba menebak angka yang diacak oleh program yang sudah saya buat. Jika user belum mengisi kolom tebakan lalu menekan tombol, akan muncul peringatan untuk mengisi tebakan. Jika tebakan terlalu rendah atau tinggi, pesan muncul untuk membantu pemain mendekati angka yang benar. Pemain memiliki 10 kesempatan menebak, dan angka acak berubah otomatis jika kesempatan habis. Skor dihitung berdasarkan sisa kesempatan dikalikan 10. Setelah berhasil menebak, pop-up meminta nama pemain untuk dicatat pada tab skor.ketika kita menombol reset lalu tekan no makan program akan keluar

Pertama saya membuat random angka ini untuk mengacak angka yang keluar di terminal , kemudian saya membuat variabel tebakan sama dengan variabel angka .NextInt untuk memberikan angka yang lain lalu saya membuat System.out.println(Tebakan); di bagian public frame dan di button reset agas ketika kita run akan menampilkan angkanya dan juga akan tampil ketika kita meresetnya untuk permainan ke 2

# PENUTUP

## Analisa

Dari hasil praktikum, praktikan menganalisis bahwa program GUI menggunakan Tabbed Pane berguna untuk menyusun dan mengelompokkan berbagai komponen Java lain ke dalam lapisan - lapisan (layers). Jadi, ketika obyek kelas JTabbedPane dimunculkan maka hanya satu lapisan (one layer) yang ditampilkan. Untuk berpindah dari satu lapisan ke lapisan lain, cukup mengklik pada tab yang diinginkan.

Seperti pada program diatas, jika kita mengklik tab “Date and Time” maka akan menampilkan sebuah informasi tentang waktu dan tanggal. Selain itu juga dapat mengubah waktu dan tanggal serta dapat mengubah zona waktu. Jika mengklik tab “Additional Clocks” maka akan menampilkan jam tambahan, seperti memilih zona waktu dan di tab ini memiliki opsi jika kita ingin menggunakan zona waktu, maka kita harus mengklik kotak “Show this clock”. Dan jika mengklik tab “Internet Time” maka akan menampilkan sebuah pengaturan

## Kesimpulan

Dari hasil praktikum, praktikan menyimpulkan bahwa Internal Frame adalah komponen yang digunakan untuk membuat jendela internal dalam aplikasi. Ini memungkinkan pengaturan tampilan atau dokumen yang berbeda dalam satu jendela utama, meningkatkan struktur tampilan antarmuka pengguna. Di samping itu, Tabbed Pane digunakan untuk mengatur konten dalam tab, memberikan pengguna akses yang mudah ke berbagai fitur atau informasi dalam aplikasi. Praktikan juga berhasil menguasai aturan penggunaan JInternalFrame dan berhasil membuat contoh GUI dengan JTabbedPane. Hasilnya adalah antarmuka pengguna yang terstruktur dengan baik, memungkinkan pengguna untuk dengan mudah menjelajahi konten dan fitur dalam aplikasi melalui tab yang tersedia.