

第2章 フレームのデザイン

【学習内容とねらい】

Windows 上のアプリケーションソフトウェアでは、プログラムの操作画面上にボタンやメニューが用意されておりユーザはそれらをマウスによって適宜選択することで、必要な処理を実現できるようになっています。それでは、このようなプログラムを作成するにはどうしたらいいのでしょうか？かつては、プログラム作成時に開発者がボタンや入力欄等の設計（形状、大きさ、色等のデザイン）を一から行っており、これはとても大変な作業でした。俗に「ユーザは天国。しかし作る方は地獄の世界！」と言われたものです。

これに対して、前章で概観した通り JBuilder では、プログラム開発上必要となるボタンや入力欄そしてメニューなど（JBuilder ではこれらを「コンポーネント」と呼びます）を予め用意しておき、開発者はそれらをプログラム基盤（フレームと呼びます）の適当な位置に貼り付ける（デザインする）事でプログラム操作画面を設計する事が可能になっています。これにより作業効率が大幅に向上しました。この様なプログラム開発形態の事を「**ビジュアルプログラミング**」と呼びます。

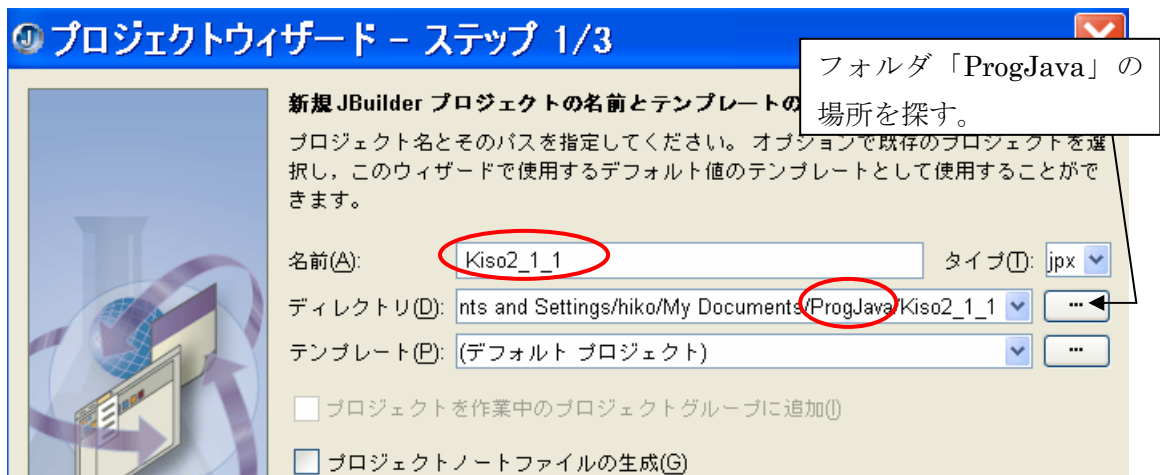
今や、ビジュアルプログラミング環境は、Windows 上のソフトウェア開発形態の主流になりました。本章ではビジュアルプログラミングを行うに当たっての基本動作を学習します。具体的に言うと、代表的なコンポーネントをフレームに貼り付ける操作、つまりフレームのデザインの練習です。とりあえず、この章の学習だけで、“形だけ”は立派なプログラムを作る事ができるようになります。ここで、大事なことは、各々のコンポーネントには大きさや色等の、特性（プロパティと呼びます）があるという事です。これらプロパティの指定の仕方も重要な学習内容です。本章では、「コンポーネント」、「フレーム」そして「プロパティ」がキーワードになります。

<第2章の構成>

- 2-1 ボタンのプロパティ
- 2-2 いろいろなコンポーネント
- 2-3 プロパティに対する操作
- 2-4 コンポーネントの name プロパティ
- 2-5 章末課題

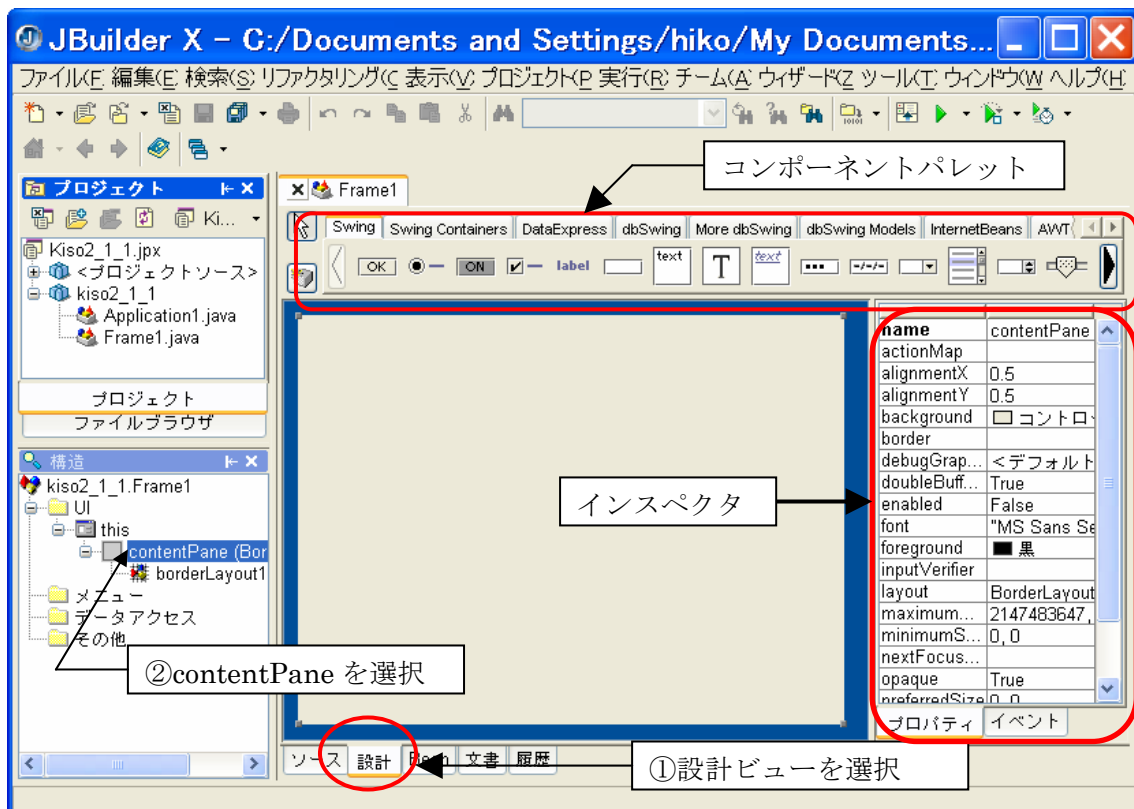
2-1 ボタンのプロパティ

JBuilder を起動し、1-2 節で学習した通り、プロジェクトそしてアプリケーション（のひな形）を新規作成してください。その際、プロジェクト名は「Kiso2_1_1」とし、ディレクトリとしては前章で作成した「ProgJava」を指定してください。



その他はアプリケーションウィザードのフレームタイトルを「ボタンコンポーネント」とする他は、特にデフォルト（既定値）から変更する必要はありません。以下、特に指示がなければ、プロジェクト名としては課題名を指定するようにしてください。

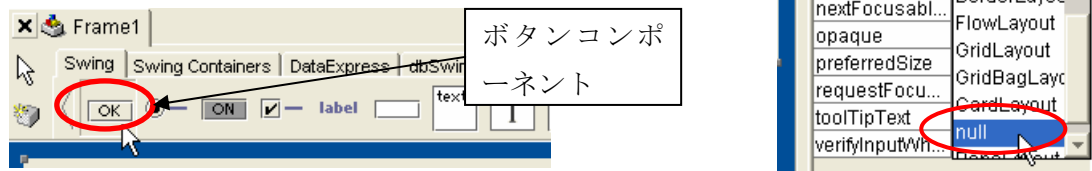
作成したらファイルビュータブから「設計ビュー」を選択してください。



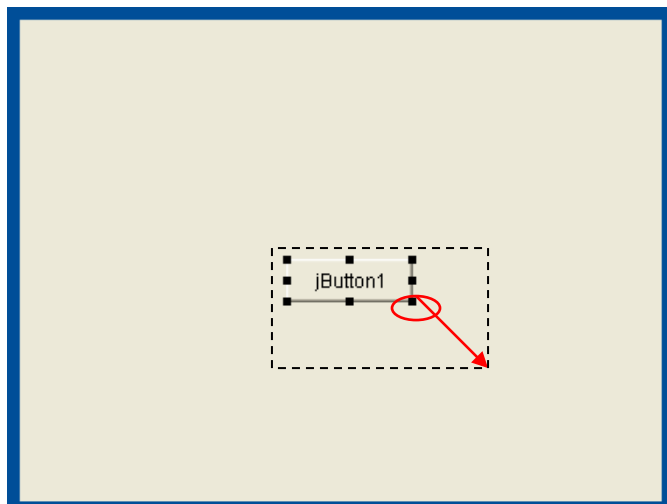
ここで、上の図のように構造ペインから「contentPane」を選択してください。また新しい名前が出てきて混乱しそうですが、ここでは「contentPaneの上にボタンなどのコンポーネントを配置できる」ということだけ頭に入れておいてください。(→次頁のコラム参照)

さて、コンポーネントは(内容ペイン上方にある) **コンポーネントパレット**に用意されています。現在「**Swing**」というタブに登録されているコンポーネントが表示されています。以下の要領に従って、ボタンコンポーネントを配置しましょう。

- ① コンポーネントを配置する準備として、「contentPane」のインスペクタにある「**layout**」欄をクリックし、そこから「**null**」を選択してください。(→次頁のコラム参照)
- ② **Swing** タブの左端にあるボタンコンポーネントを選択(クリック)し、その後 contentPane 上の適当な場所(例えば中央)をクリックしてください。



- ③ すると、次のように contentPane 上にボタンが配置されます。



ボタンコンポーネントの中央をクリックしてドラッグすると、ボタンの位置を変えることができます。また、ボタン周囲の6つの黒点をクリックしてドラッグすると、ボタンを任意の大きさに変えることができます。

このプログラムを実行すると、ボタンが配置されただけのウィンドウが現れます。

【基礎課題 2-1-1】

上の手続きに従って、contentPane 上にボタンを配置し、ボタンの移動と拡大・縮小が自在にできることを確認してください。

コラム 用語の整理:Frame、contentPane そして Swing とは？

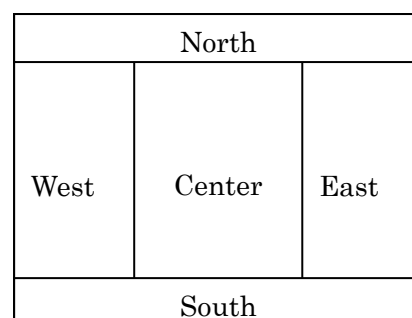
Java 言語では、GUI コンポーネントを **Swing** と命名されたライブラリに集めています。ライブラリとは、ここでは、プログラム作成に用いる” 道具集 ” とでも言うておきましょう。JBuilder では、その Swing に入っている GUI コンポーネントを、コンポーネントパレットの Swing タブに集めています。

次に、通常のWindowアプリケーションで用いるウィンドウ（表示画面）に対応するものが、Swingライブラリにあるフレーム（**JFrame**）というコンポーネントです。ですから、このフレームコンポーネントにボタンなどGUIコンポーネントを貼り付ければ良いのですが、Java言語では、フレームコンポーネントにあるコンテンツペイン（contentPane）成分の上に貼り付ける、という様になっています。少しややこしいですね。このcontentPaneの正体については第 10 章で学習しますが、ここではとりあえず、「フレームコンポーネントの上にコンテンツペインがかぶさっていて、さらにその上にGUIコンポーネントを貼り付ける」ようになっていると、理解しておいてください。

コラム レイアウトについて

contentPaneのプロパティを設定した際に、最初は「**BorderLayout**」となっており、「**null**」以外にも「**GridLayout**」など、様々なレイアウトがあることに気づいたと思います。まず、「**null**」というレイアウトは、特にレイアウト（配置）を指定しない、という意味です。ですから、コンポーネントを任意の場所に配置することができます。

これに対して「**BorderLayout**」の場合は、次のように東西南北に中心という 5 つの区画のいずれかに強制的に配置されます。例えば中心付近をクリックしたら **Center** に、右端部をクリックしたら **East** に配置される、という具合です。

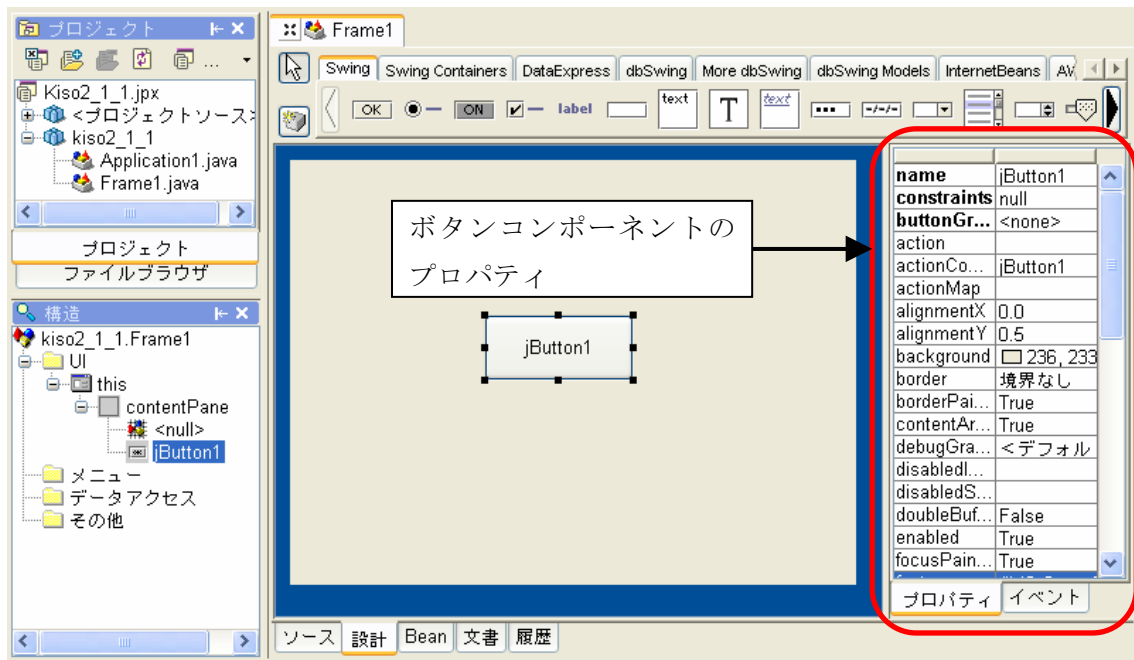


そのほか、「**GridLayout**」の場合は網目の様に格子状にコンポーネントを配置する場合に用いられます。

Java 言語では、「**BorderLayout**」を中心に、これら所定のレイアウトを用いることを想定しているのですが、最初の内は、レイアウトに戸惑うこともあるので、本テキストでは、特に断らない限り「**null**」を用いて、フリーハンドでコンポーネントを配置する事にします。

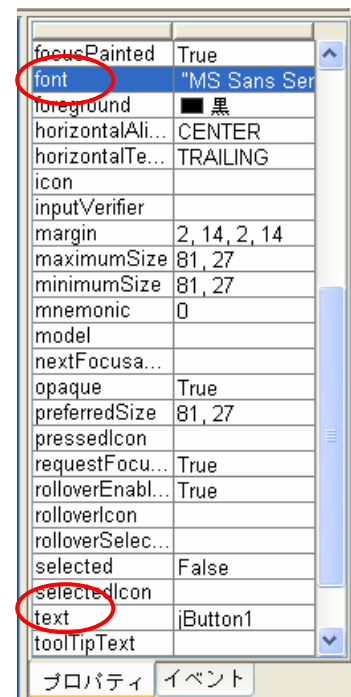
【基礎課題 2-1-2】

【基礎課題 2-1-1】のプログラムで、ボタンコンポーネントを選択した状態にしてください。構造ペイン上で「jButton1」をクリックするか、contentPane 上でボタンコンポーネントをクリックすると選択できます。



インスペクタを利用して、ボタンコンポーネントのプロパティを少しいじってみましょう。

- (1) インспекタの「text」欄には「jButton1」と書かれています。「jButton1」を消して、何か文字を入力して下さい。何が起きるかを担当の補助学生あるいは指導員に説明して下さい。
- (2) インспекタの「font」欄をクリックして、「MS Sans Serif, 0, 13」の右端をクリックしてください。ここで、ボタンに表示されている文字のフォントと大きさを変えることができます。適当に変更してみてください。
- (3) インспекタの上方にある「background」欄をクリックするとボタンの外枠の色を指定することができます。各自好きな色に変更してみてください。

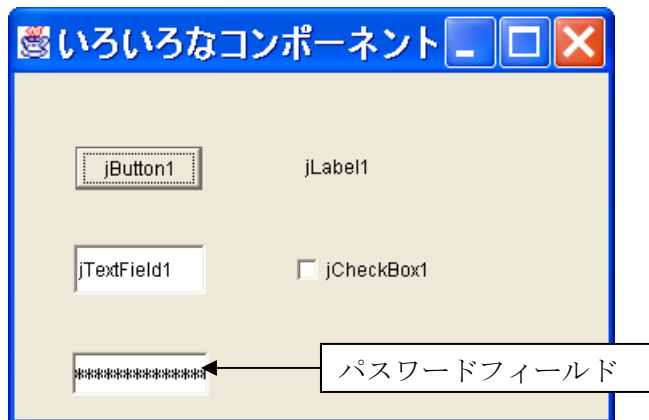


2-2 いろいろなコンポーネント

代表的な Swing コンポーネントを使用してみましょう。

【基礎課題 2-2-1】

次のようなフレームを作ってみましょう。



これは、5つのコンポーネントを配置しただけで、実用的な意味はありません。とにかく、コンポーネントの扱いに慣れることがこの目的です。

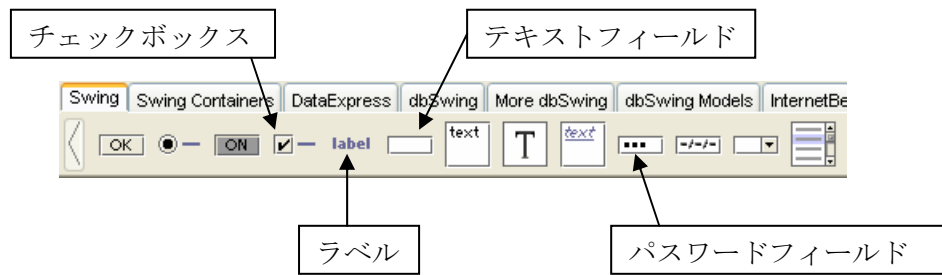
新しいプロジェクトとして作成してください。やり方は、1-2 節で説明した通りですが、念のために以下の点を注意しておきます。不明な点があれば 1-2 節を参照してください。

＜新規アプリケーション作成にあたって＞

- ① 既存のプロジェクトが表示されていれば、それを閉じる。
- ② 新規プロジェクトを [ファイル] → [新規] から「アプリケーション」を選択し、プロジェクトウィザードによって作成する。その際、プロジェクト名は課題名「Kiso2_2_1」とすること。その他は、デフォルト（既定値）から変更の必要はない。
- ③ 続いてアプリケーションウィザードにおいて、フレームタイトルを上の実行例のように「いろいろなコンポーネント」とする。
- ④ フレームのデザインは設計ビューで行う。
- ⑤ その際、前節で行ったように、構造ペインで「contentPane」を選択し、layout プロパティを「null」にしてからコンポーネントを貼り付けること。

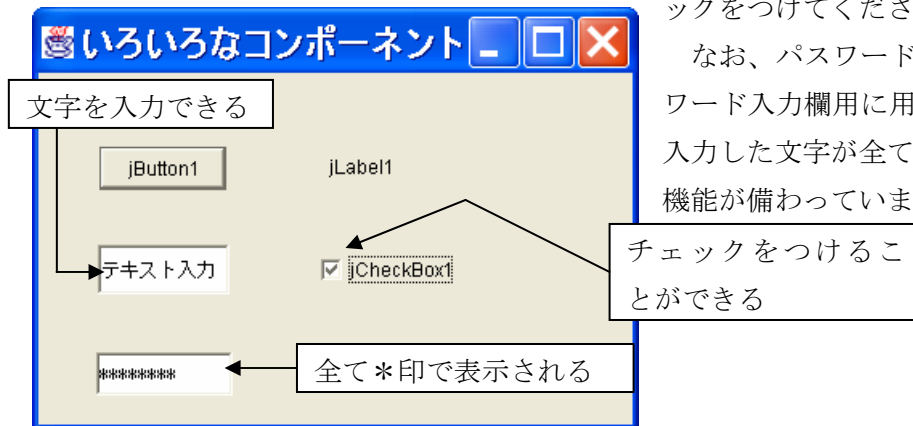
ここでは、(ボタンコンポーネント以外に) 新たに4つのコンポーネントを用いています。それらは、コンポーネントパレットの **Swing** タブに次のように用意されています。前節のボタンと同様に各コンポーネントを配置し、配置できたらプログラムを実行させてください。

<各コンポーネント>



< 解説 >

プログラムの実行画面では、「テキストフィールド」、「パスワードフィールド」には、入力文字の削除や挿入を行うことができます。また、チェックボックス欄にはクリックすることでチェックの有無を切り替える事ができます。次のように、入力文字を変更し、チェックをつけてください。



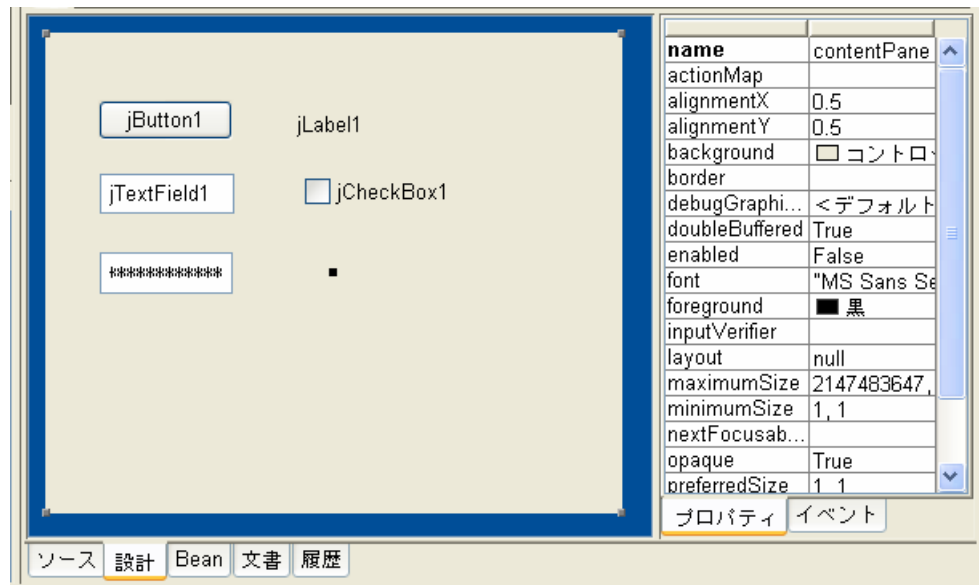
チェックをつけてください。

なお、パスワードフィールドはパスワード入力欄用に用意されているので、入力した文字が全て「*」で表示される機能が備わっています。

コンポーネント	コンポーネントの名前	はたらき
	ボタン	押すことができる。
	ラベル	文字を表示することができる。
	テキストフィールド	文字を1行分入力することができる。
	パスワードフィールド	パスワードのような非開示文字を入力できる。
	チェックボックス	白い四角形をクリックして、チェックをつけたり外したりできる。

2-3 プロパティに対する操作

前節のプログラム（プロジェクト）を開いておいてください。本節では、各コンポーネントの代表的なプロパティをいじってみましょう。そうすることで、どのようなプロパティがあるか、に対する全体的なイメージがつかめてくるはずです。



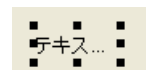
【基礎課題 2-3-1】

以下の I ～ V までの操作を行い、動作結果を確認してください。

I . text プロパティ

2-1 節でボタンの text プロパティを変更しました。実は、text プロパティは上の 5 つのコンポーネント全てにあります。そこで、全コンポーネントの text プロパティを「テキストの変更」という文字に変更してください。その際、次の点に注意してください。

- ① プロパティの変更は、目的とするコンポーネントを選択した状態で、インスペクタの当該プロパティを変更します。コンポーネントの選択は、デザイナ上でクリックするか、構造ペインで当該コンポーネントをクリックすることで行えます。
- ② コンポーネントのサイズが小さいと、右のように表示がはみ出る場合があります。そのときは、コンポーネントのサイズを適宜拡大してください。



II. font プロパティ

font プロパティも全てのコンポーネントに存在します。そこで、font プロパティを変更して、文字のスタイルを「斜体」あるいは「太字」に変更してください。

III. background プロパティ

background プロパティも、上の全てのコンポーネントに存在します。これは、コンポーネントの色を指定するプロパティです。それぞれのコンポーネントを適当な色に変更してください。

IV. enabled プロパティ

もう一つ、共通に存在するプロパティとして、enabled プロパティがあります。このプロパティの値としては、「True」か「False」かの2つしかありません。最初は「True」になっています。そこで、ボタンコンポーネントと、テキストフィールドコンポーネントの enabled プロパティを「False」に変更するとどうなるかを確認、担当の補助員あるいは指導員に（どのような状態になったか）説明してください。プログラムを実行してみると分かるはずです。

V. selected プロパティ(チェックボックスコンポーネント)

各コンポーネントには、特徴的なプロパティもあります。例えば、チェックボックスコンポーネントの selected プロパティは、ラベルやテキストフィールドにはありません。この selected プロパティも、enabled プロパティ同様「True」か「False」のいずれかの値しかとりません。そして最初は「False」になっているはずです。そこで、チェックボックスコンポーネントの selected プロパティを「True」に変更するとどうなるかを担当補助員あるいは指導員に説明してください。

以上、主立ったプロパティを眺めてみましたが、共通に存在するプロパティと、特殊なプロパティのイメージが大まかにつかめたと思います。

<補足ーコンポーネントの削除の仕方>

次のようにすれば、貼り付けたコンポーネントを削除することができます。

- ① 削除したいコンポーネントを選択する（構造ペインで当該コンポーネントを選択するか、デザイナー上で直接クリックする）。
- ② その後、JBuilder の「編集」メニューから「切り取り」を選択する。または、直接 [Delete] キーを押す。

【基礎課題 2-3-2】

これまでの学習の総復習です。実行時に次のような電子メール送信画面が現れるプログラムを作成してください。

＜理解度チェックのポイント＞

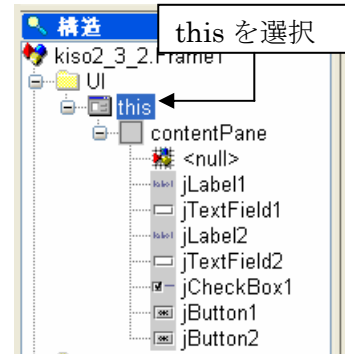
- ① 新規にプロジェクトおよびアプリケーションを作成する手順は大丈夫ですか？（まず、今開いているプロジェクトを閉じることを忘れないように）
 - ② フレームタイトルの指定はどこで行いますか？
 - ③ フレーム（上の contentPane）にコンポーネントを貼り付ける手順は大丈夫ですか？
- ※ プロジェクトの名前は、課題名「Kiso2_3_2」とすることを忘れないようにして下さい。

【基礎課題 2-3-3】

【基礎課題 2-3-2】のプロジェクトを開いた状態にしてください。このプログラムに次のようにメール本文の入力欄を加えましょう。

新しく追加したテキストエリア (JTextArea) コンポーネント  は、コンポーネントパレット **Swing** タブの左から 7 番目にあります。これは、複数行のテキストを入力する際に用いるコンポーネントです。そしてやはり、**text** プロパティに入力した文字が表示されるようになっています。

このフレームを作成する際に、フレームの大きさを拡大する必要があります。その際は、まず次のように構造ペインで「**this**」を選択してください。そうすると、他のコンポーネントと同様にフレームを拡大縮小する事ができます。「**this**」とは、フレームの事です。フレーム自体の大きさを変更する際は、このように「**this**」を選択してから行うようにしてください。「**contentPane**」が選択された状態では、フレームの大きさは変更できないので注意が必要です。



2-1 節のコラムで解説した様に、**contentPane** はフレームの上に張られた”膜”のようなものです。そして **Java** 言語ではその膜 (シート) の上にコンポーネントを配置する様になっています。フレーム (**this**) の大きさを変えると、その上の膜である **contentPane** の領域も (自動的に) 変更する様になっています。

注意 設計ビューでフレームの設計を行う際には、フレームタイトルは表示されていません。ところが、プログラム実行時にはフレームタイトルが自動的に挿入されます。そのため、フレームの設計時に (縦方向の長さを) その分だけ余分にとっておく必要があります。

なお、プロジェクト名は、「**Kiso2_3_2**」のままで結構です。

プログラムを実行したら、改めて次の 3 点を確認して下さい。

- ① 「件名」「宛先」「メール本文」の欄に、入力ができること。
- ② チェックボックス欄のチェックとその解除ができること。
- ③ 「送信する」「キャンセル」両方のボタンが押せること。

残念ですが、「送信する」ボタンを押しても、メールが送信されることはありません。その動作部分を、まだプログラミングしていないからです。

ボタンを押したら、自分のしたいことをしてくれるようなプログラムを作る。それが本講義 (演習) の目標なのです。これは次章以降で学習します。

2-4 コンポーネントのnameプロパティ

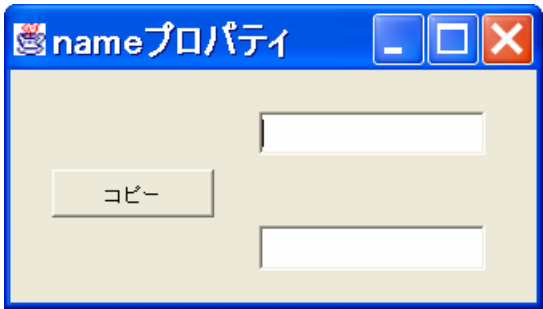
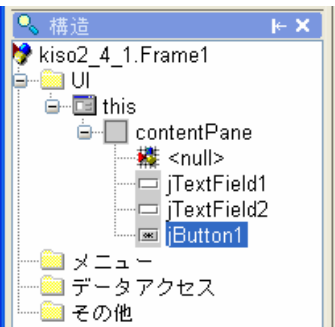
すべてのコンポーネントには、「name」プロパティがあります。name プロパティの値は、そのコンポーネントの名前を表します。

コンポーネントをフレームに貼ると、「jTextField1」、「jTextField2」、「jLabel1」、「jLabel2」など、コンポーネントの種類と数字を組み合わせた名前が自動的につけられます。しかし、名前ですから、自分で自由につけることもできます。コンポーネントの数が少ない場合はそのまま良いのですが、数が増えてくると、そのコンポーネントがどのような役割を果たすのかを考えて、できるだけ**わかりやすい名前**をつけることが必要になって来ます。

注意 「name」プロパティの値は何でもよいのですが、以下では、説明の都合上、当該コンポーネント名を指定する場合があります。

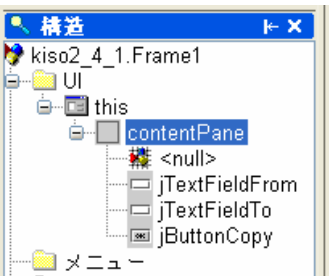
【基礎課題 2-4-1】

次のようなフレームを作ってください。



設計ビューでコンポーネントを配置した後、構造ペインを見ると、左のように表示されているはずです。ボタンが1つ、テキストフィールドが2つあることを確認して下さい。

ここで、ボタンと、2つのテキストフィールド name プロパティを、次のように変えて下さい。



コンポーネント	name プロパティ
「コピー」ボタン	jButtonCopy
上のテキストフィールド	jTextFieldFrom
下のテキストフィールド	jTextFieldTo

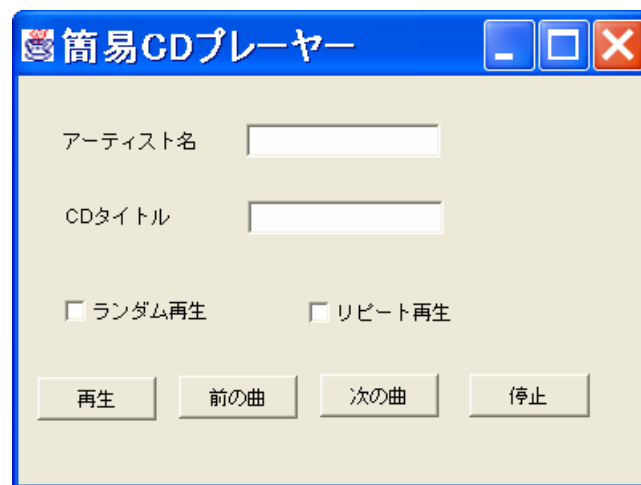
もう一度、構造ペインを見ると、右のように各コンポーネントの名前表示が変わっているはずです。これを確認してください。

2-5 章末課題

以下の各課題は、必ずきちんと課題毎に（プロジェクトとして）保存して下さい。

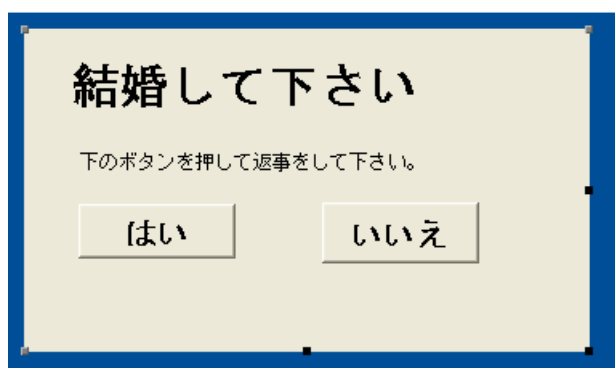
【基礎課題 2-5-1】

下のようなフレームを作して下さい。

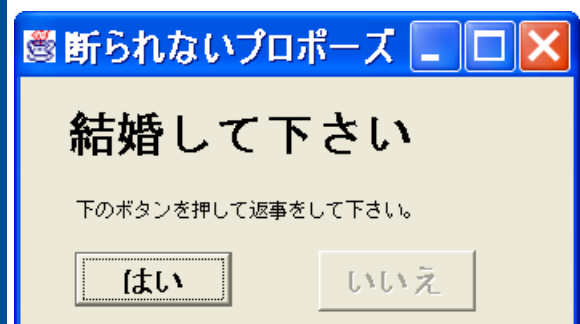


【基礎課題 2-5-2】

(1) 下のようなフレームを作ってください。



(2) ただし、実行したときに、「いいえ」ボタンが押せないようにしておきましょう。



ただし、もし断られたとしても、当方は一切の責任を負いません。

【基礎課題 2-5-3】

下のようなウィンドウが現れるプログラムを作ってください。

ストレス度診断テスト

ストレス度診断テスト

以下の該当する項目をチェックした後、診断ボタンをクリックして下さい。

- ☐ 家庭内で色々な問題があった。
- ☐ 日頃から楽しみにしている趣味などない。
- ☐ 気分が沈みがちで憂鬱である。
- ☐ ささいな事に腹が立ち、イライラする。
- ☐ 人に会うのがおっくうで、何でもめんどくさい。
- ☐ 前日の疲れがとれず、朝方から体がだるい。
- ☐ 朝気持ちよく起きられずに、気分が悪い。
- ☐ 頭がすっきりしなく、頭重感がある。
- ☐ 腹が張り、下痢や便秘を交互に繰り返す。
- ☐ 目が疲れたり、めまいや立ちくらみがある。

(参考) 関谷透『ストレス度自己診断テスト』より

結果

できたら実行して、あなたもストレス度チェックをしてみてください。
チェックされた項目の数と、メッセージの対応は、次のようになるそうです。

チェック数	メッセージ
9～10	かなりストレスがたまっています。医師による診察を。
6～8	少しストレスがたまっています。気分転換を。
3～5	ストレスの兆候がありますが、心配は不要。
0～2	正常です。

第5章までの学習を終えれば、「診断」ボタンを押すとチェック数を数えてメッセージを表示するようなプログラムを作れるようになります。

【基礎課題 2-5-4】

下の表の左にはコンポーネント名、上にはプロパティ名が並んでいます。インスペクタを見て、コンポーネントにそのプロパティがあれば○、なければ×をつけて下さい。

	background	font	enabled	text	selected	name
ボタン						
ラベル						
チェックボックス						
テキストフィールド						
テキストエリア						