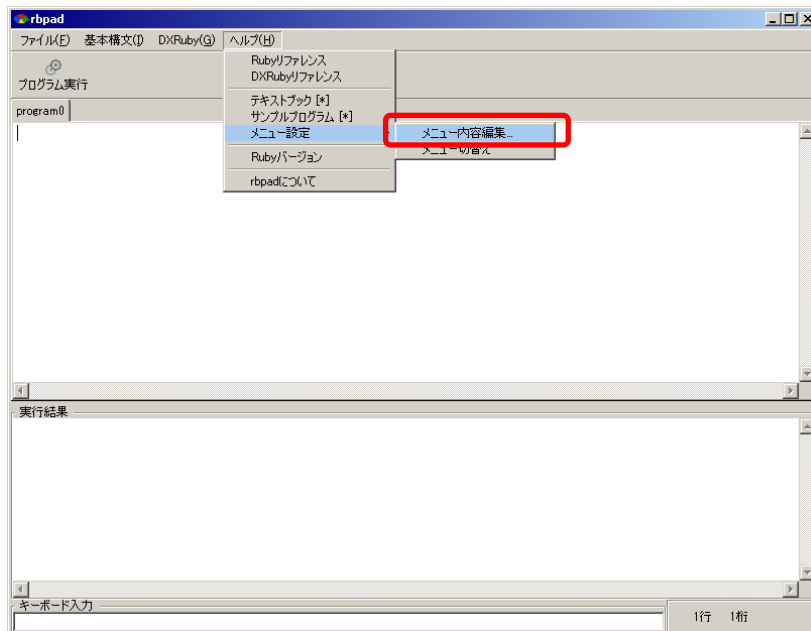


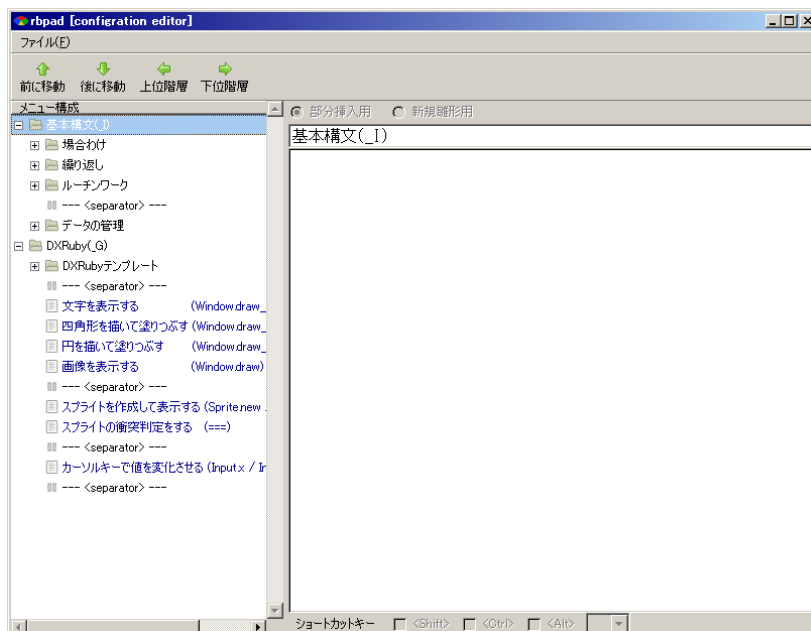
『rbpad』のメニューを使いやすい設定にしてみよう！

1. メニュー編集用ツール『Configuration Editor』を開いてみよう

(1)rbpad の[ヘルプ]-[メニュー設定]-[メニュー内容編集..]を選んでください。

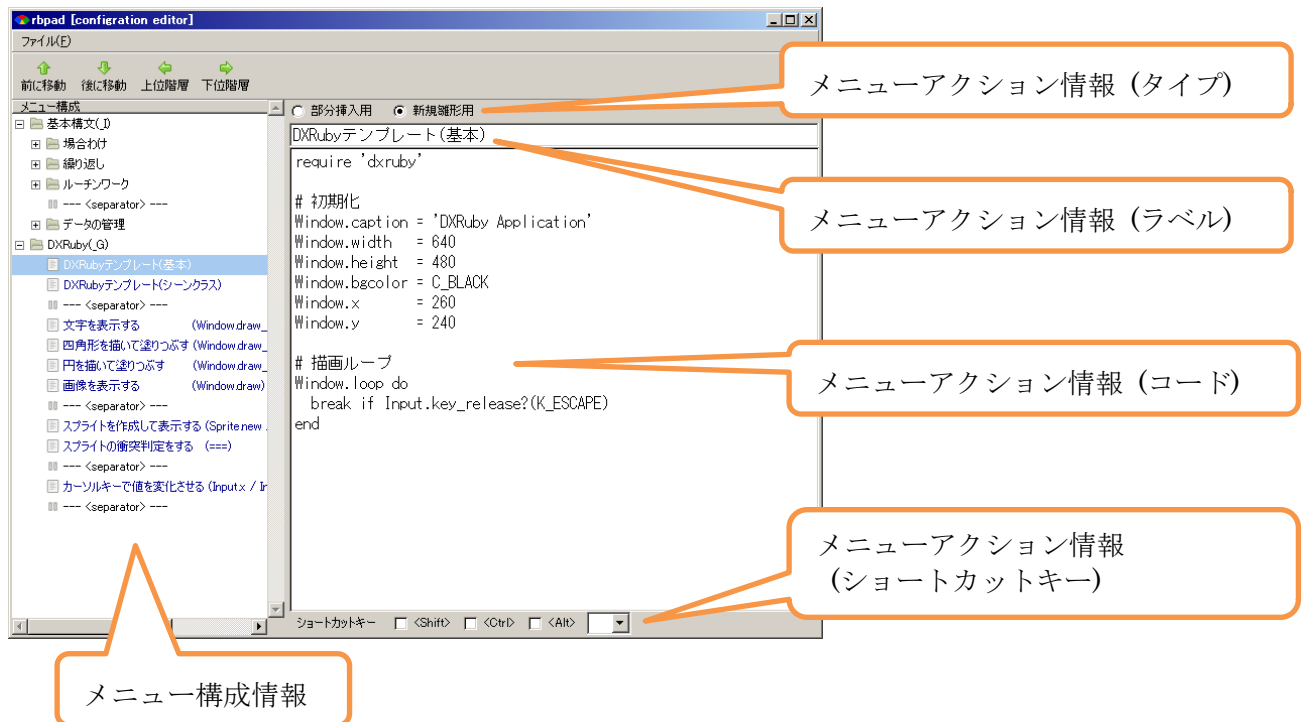


(2)『rbpad [configuration editor]』というタイトルのウィンドウが開きます。



2. 表示される内容を確認してみよう

(1) Configuration Editor では、rbpad のメニューバーに表示されるメニューのうち、
[ファイル]と[ヘルプ]以外のメニューに関する情報をすべて編集することができます。

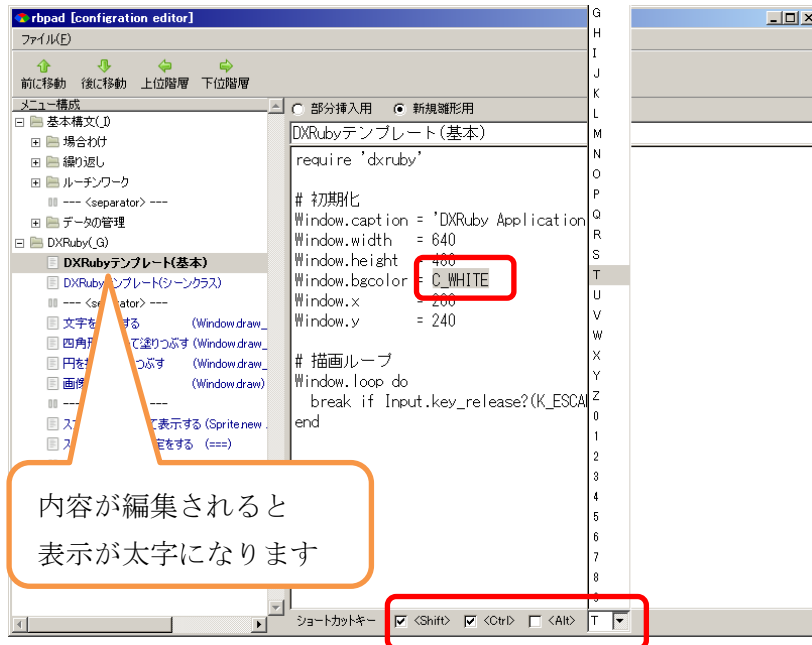


メニュー構成情報	rbpad に表示されるメニューの構造がツリー形式で表示されます
メニューアクション情報	メニュー(ツリー表示の枝の部分)や メニュー項目(ツリー表示の葉の部分)の内容が 下記の各セクションに表示されます
(1)タイプ	「部分挿入」か「新規雛形(テンプレート)」かを識別します <ul style="list-style-type: none"> ・部分挿入 ... エディタ画面のカーソル位置に挿入 ・新規雛形 ... エディタに新しいタブを追加して そこにテンプレートとして挿入
(2)ラベル	メニューやメニュー項目の名称が表示されます
(3)コード	挿入される Ruby のコードが表示されます
(4)ショートカットキー	挿入の操作をキーボードでおこなうための ショートカットキーの定義が表示されます

3. メニュー項目の内容を変更してみよう

(1)メニューアクション情報の表示エリアで、定義内容を自由に編集することができます。

下記の例では、Window.bgcolor = C_BLACK の記述を C_WHITE に変更し、ショートカットキーとして [Shift]-[Ctrl]-[T]で呼び出せるようにしています。



4. メニューの構成を変更してみよう

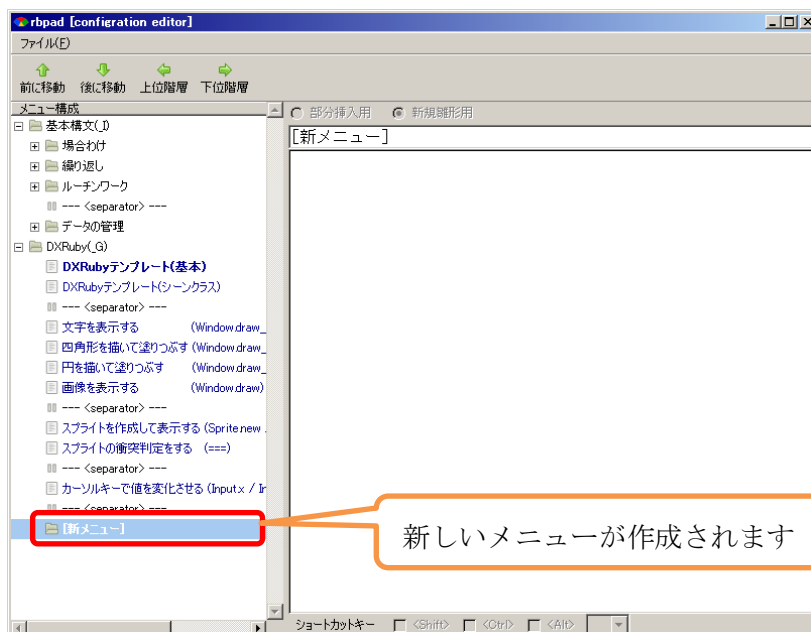
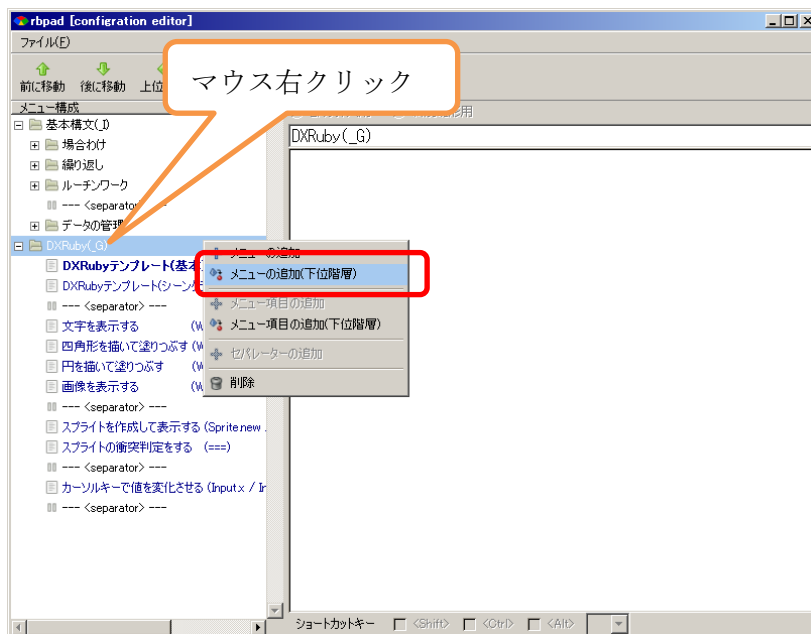
- (1)メニュー構成情報のツリー表示エリアで、メニューやメニュー項目の構成を自由に編集することができます。

下記の一連の操作例では、

[DXRuby]メニューの下に[DXRuby テンプレート]サブメニューを追加し、既存の2つのメニュー項目([DXRuby テンプレート(基本)]と[DXRuby テンプレート(シーンクラス)])をその中に移動させています。

- (2)メニューの下にサブメニューを追加

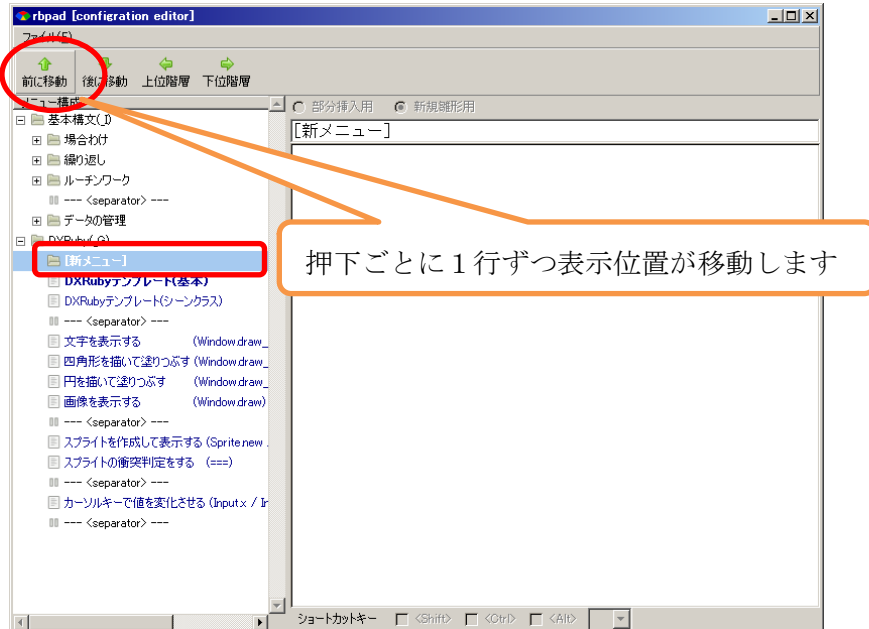
メニューを選択し、マウスの右クリックによって表示されるポップアップメニューから[メニューの追加(下位階層)]を選択することにより、新規のサブメニューが作成されます。



(3) メニュー表示位置の移動

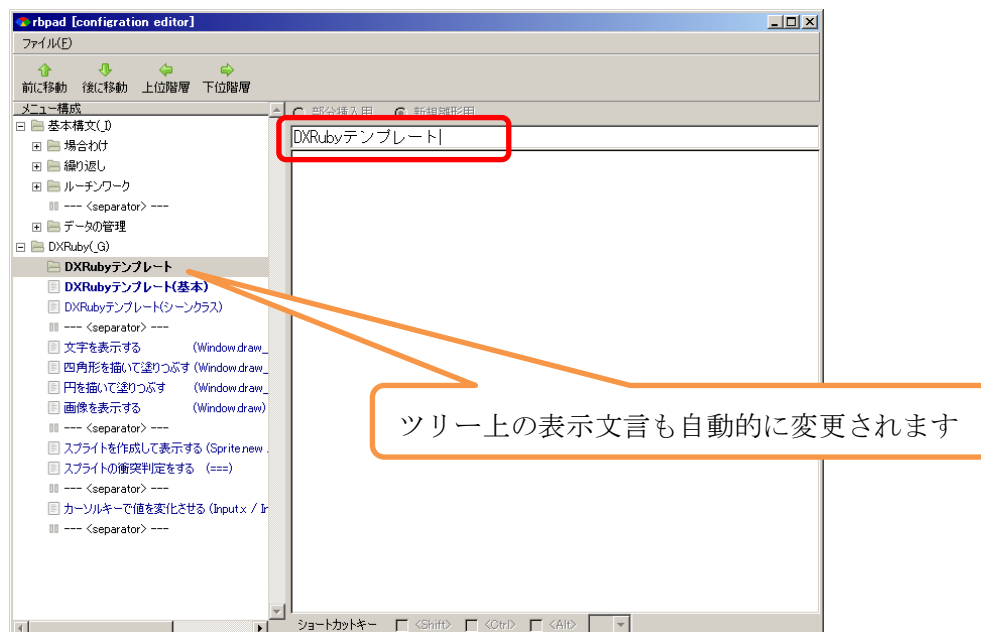
追加された[新メニュー]の項目を選択し、ツールバーの[前に移動]ボタンを押下することにより、表示位置が前に移動します。

(同様に、[後に移動]ボタンで表示位置が後ろに移動します)



(4) メニュー表示ラベルの編集

追加された[新メニュー]の項目を選択し、メニューアクション情報の表示エリアでラベルの内容を編集することができます。ここでは[DXRuby テンプレート]に変更しています。

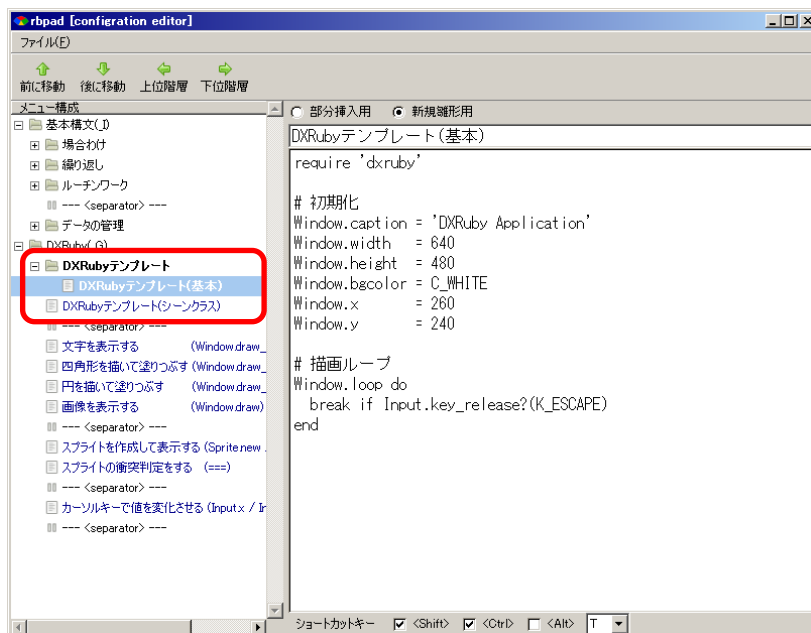
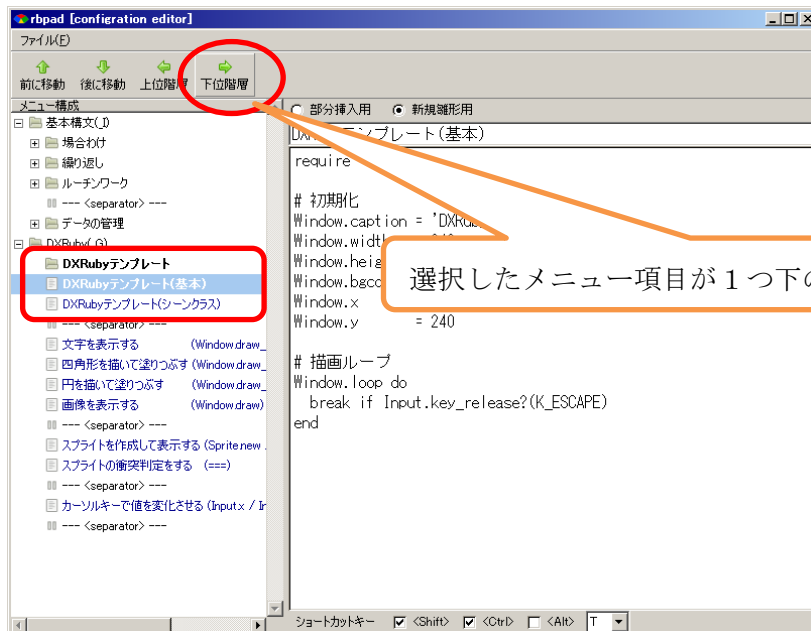


(5) メニュー項目をサブメニューの下に移動

メニュー項目を選択し、ツールバーの[下位階層]ボタンを押下することにより、1つ下の階層に移動します。

(同様に、[上位階層]ボタンで1つ上の階層に移動します。

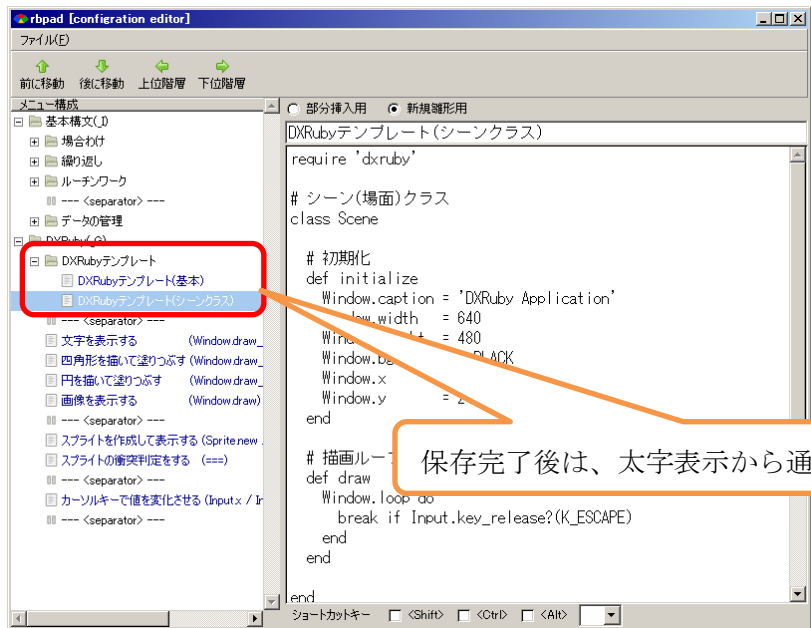
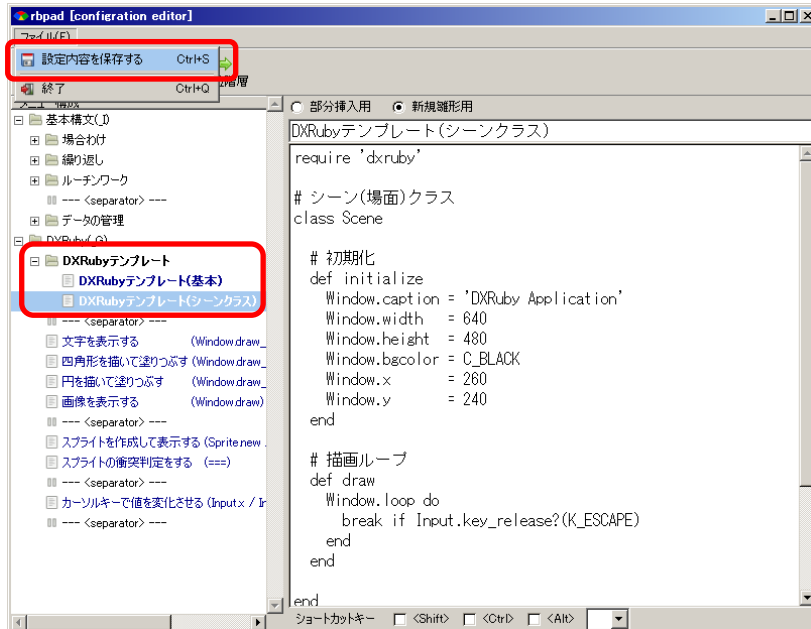
ただしメニュー項目は最上位の階層には移動することはできません)



5. 変更したメニュー情報を保存してみよう

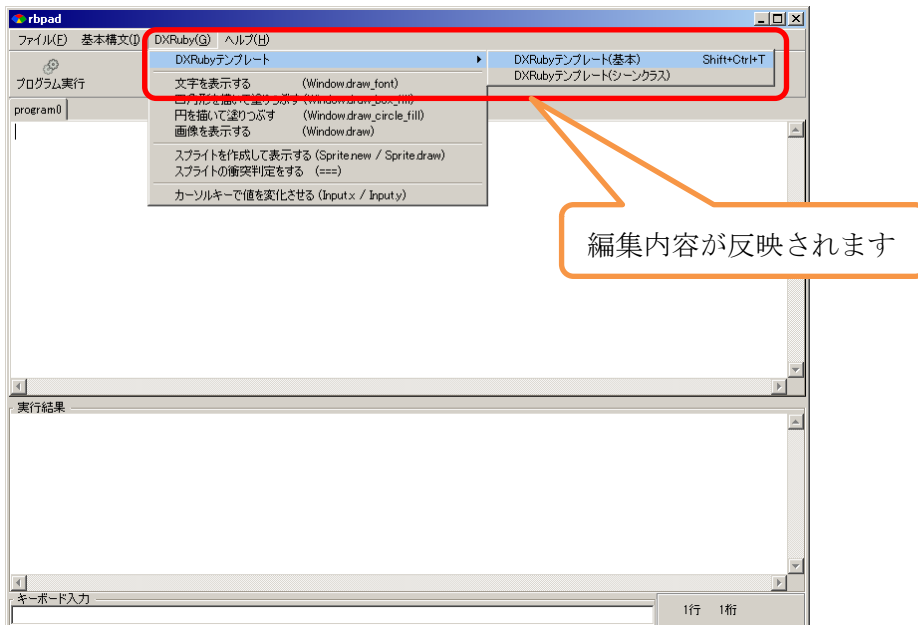
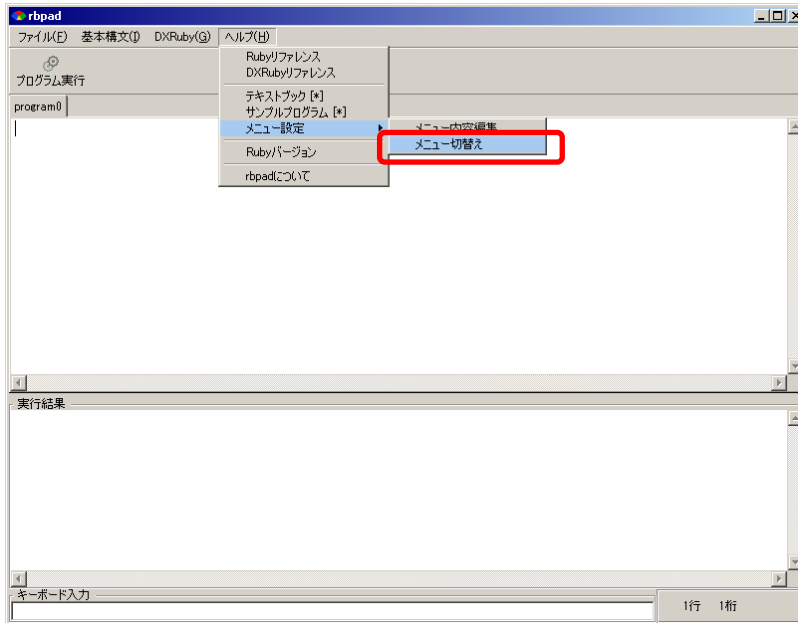
(1) メニュー編集情報の確認と保存

編集したメニュー情報の確認後、[ファイル]-[設定内容を保存する]メニュー項目を選ぶと、編集されたメニュー情報がファイルに保存されます。



6. 変更したメニュー情報を rbpad に反映させてみよう

- (1) Configuration Editor を閉じ、rbpad の[ヘルプ]-[メニュー設定]-[メニュー切替え]を選ぶと、Configuration Editor で編集した最新のメニュー情報が rbpad に反映されます。



7. 独自のメニューをどんどん作っていこう

メニューのカスタマイズ機能は操作を便利にするためだけではなく、自分だけの独自の「Ruby プログラミングライブラリ」を作っていくことができます。

プログラミングの習熟度や取り組んでいるプログラムの内容などに合わせて、どんどん便利なメニューを作っていくてください。