

Basicの基礎・はじめに

0. 目次

1. はじめに

- 1. 1 ファイル・フォルダとは
- 1. 2 「Tiny Basic」の起動と停止
- 1. 3 「Tiny Basic」の実行
- 1. 4 簡単なプログラム

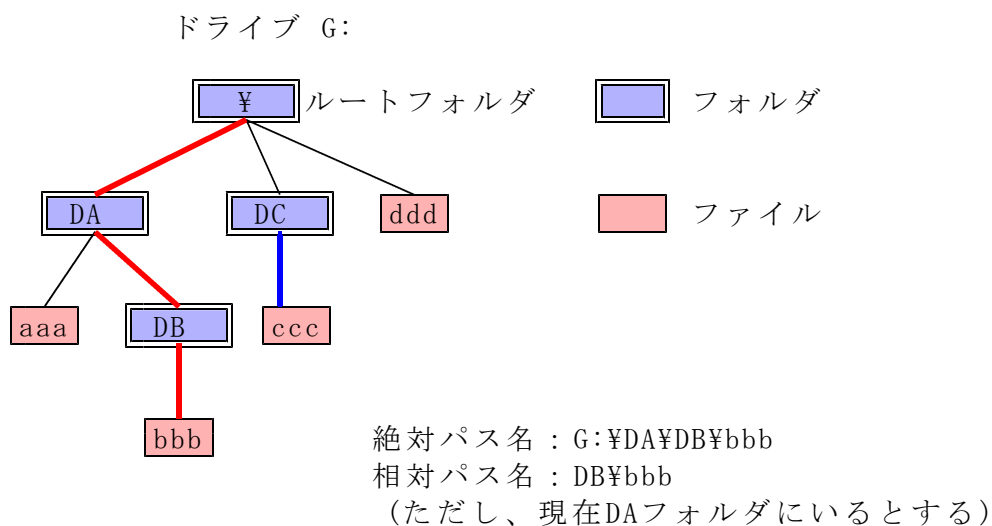
1. はじめに

1. 1 ファイル・フォルダとは

データの集まりをファイルといい、名前を付けて区別する。フォルダ（ディレクトリともいう）という特別なファイルは、ファイルが登録できるようになっている。フォルダも名前では区別する。

このフォルダにはファイルだけでなくフォルダも登録できるので、Windowsでは、つぎのような階層構造を利用してファイル管理を行っている。

USBメモリをコンピュータに差し込むと、差し込み口に対応するドライブ名とUSBメモリが対応づけられる。ここでは、ドライブ名をGとしておく。



ファイル（フォルダも含む）を区別し指定するためにパス名を使う。
絶対パス名と**相対パス名**がある。


絶対パス名 : ドライブ名から目的のファイル（フォルダも含む）までの経路をいう。
 USBメモリをパソコンに差し込むと、差し込んだ場所に割り当てられているドライブ名（「A:」はフロッピーディスク、「C:」はハードディスク、「G:」はUSBメモリ等）から始まり、ルートフォルダ「¥」を書き、以降経路にあるフォルダを順番に「¥」でつなぎ、最後に目的のファイル（フォルダも含む）を書く。

相対パス名 : 現在の位置から目的のファイル（フォルダも含む）までの経路をいう。
 経路にあるファイルを順番に「¥」でつなぎ、最後に目的のファイル（フォルダも含む）を書く。

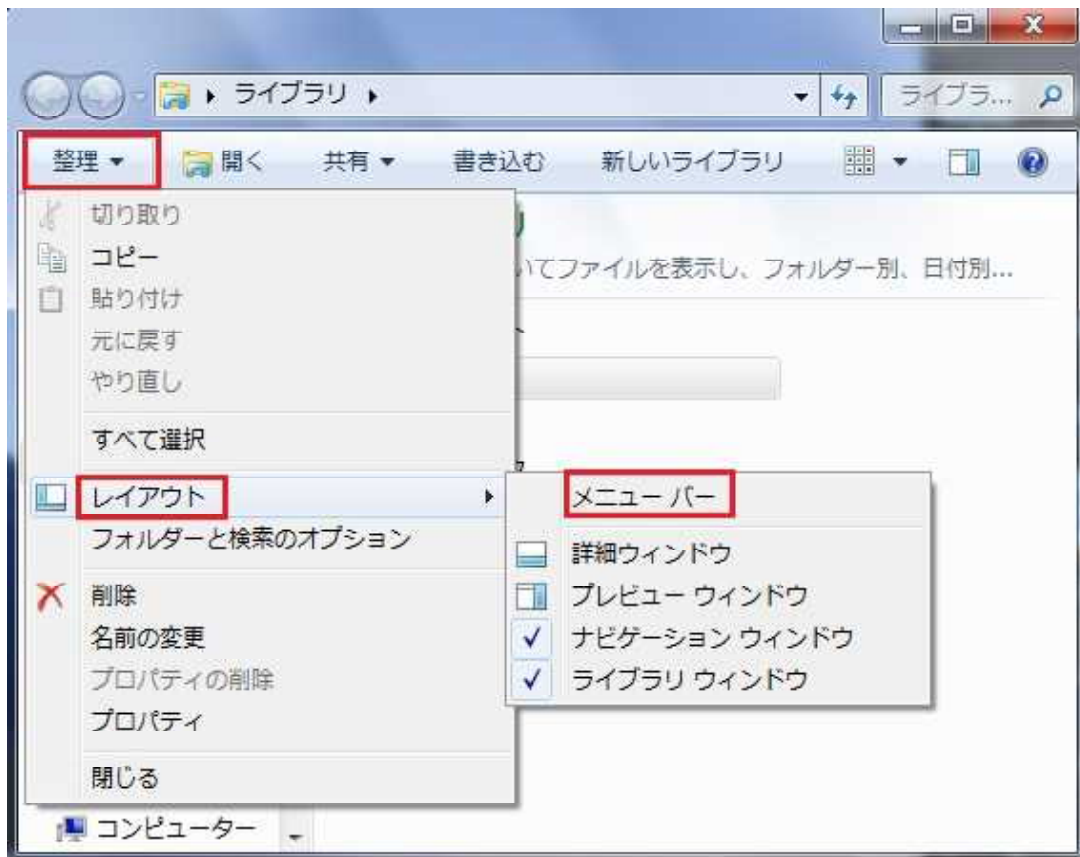
(参考) ファイルに付けられた拡張子（かくちょうし）をいつも表示される状態に設定する方法

拡張子とは、ファイル名の末尾につけられた文字列で、ピリオドより右の文字列で、ファイルの種類を識別するためのものである。

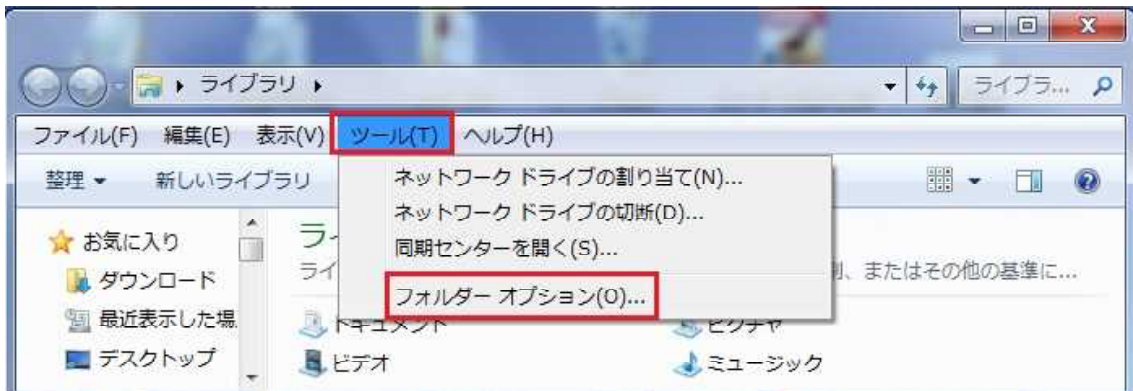
ファイル名	拡張子	ファイルの種類
abc.txt	txt	テキストファイル
fb543.doc	doc	Wordのファイル
excel.xls	xls	Excelのファイル
TEXT.pdf	pdf	PDFファイル
Sample.bas	bas	Basicファイル
Pic.jpg	jpg	画像ファイル

①画面左下隅にあるスタートボタン  を右クリックし、ポップアップウィンドウ内の **エクスプローラを開く** 項目をクリックする。

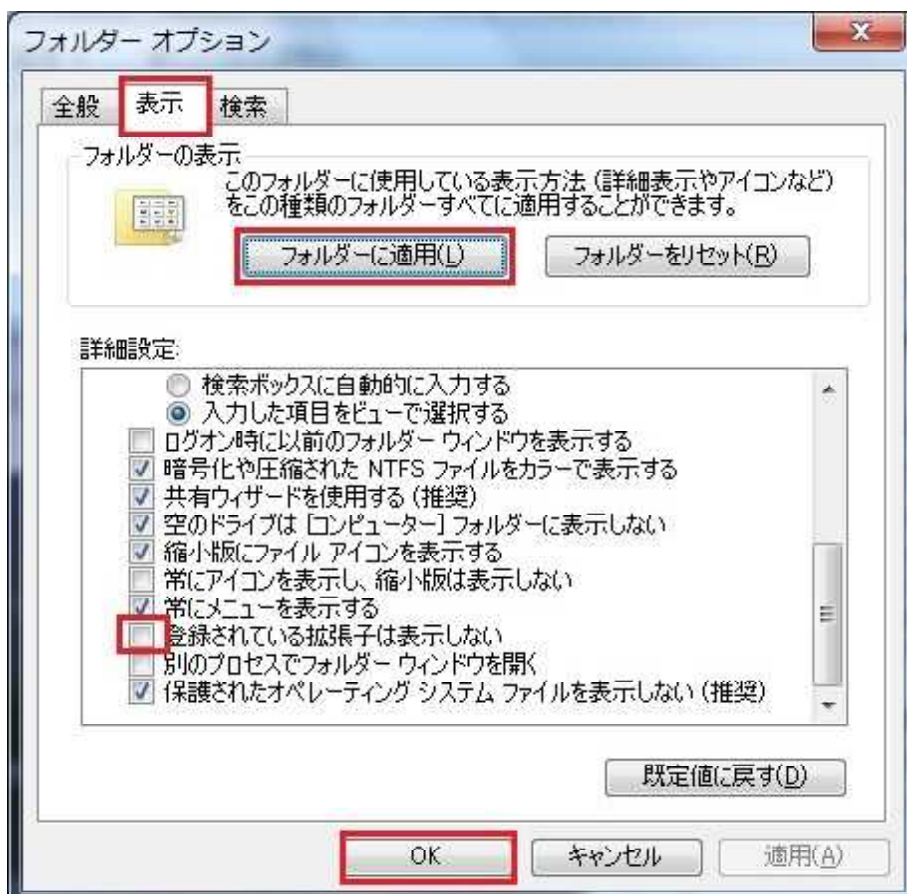
メニューバーが表示されていない場合、**整理** ボタンをクリックし、ポップアップメニューから **レイアウト** を選択し、**メニューバー** をクリックする。



表示されるウィンドウ内のメニューバー内から「ツール」をクリックし、「フォルダーオプション」をクリックする。




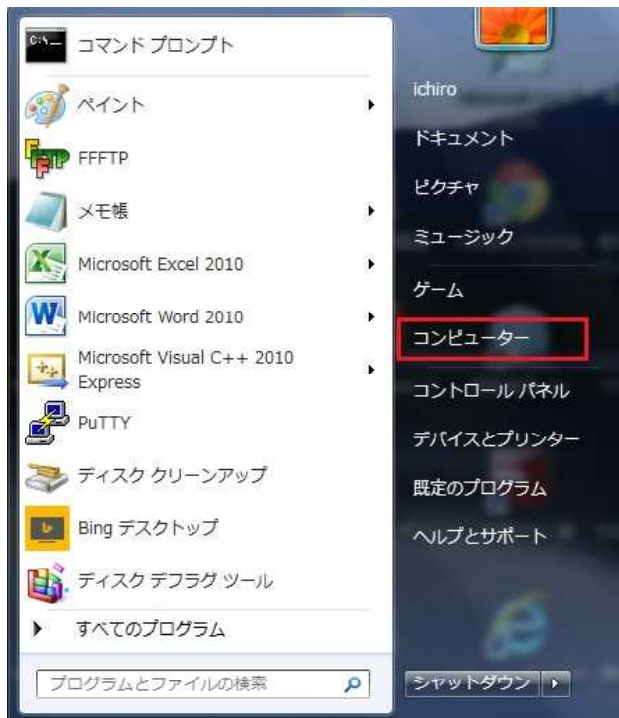
- ② 「表示」タブを選択しクリックする。
詳細設定欄のスクロールバーを下に移動し、「登録されている拡張子は表示しない」のチェックを外す。
その後、フォルダーの表示欄の「フォルダーに適用」ボタンをクリックする。
最後に「OK」ボタンをクリックする。



1. 2 「Tiny Basic」の起動と停止

「Tiny Basic」の起動と停止の方法を示す。

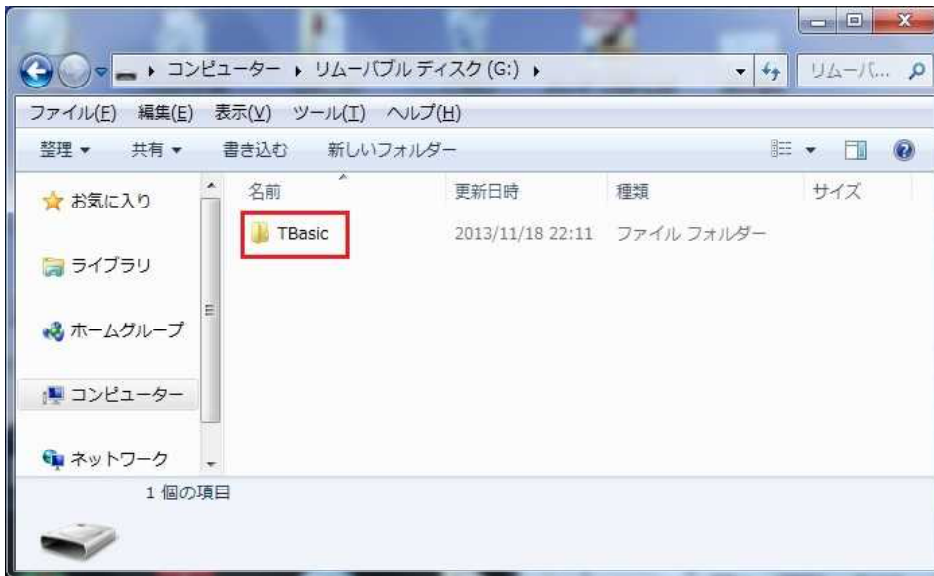
- ① 「Tiny Basic」の入ったUSBメモリをコンピュータに差し込む。
- ② 画面左下隅にあるスタートボタン  を左クリックして、ポップアップウィンドウを表示し、**コンピュータ**項目を選択しクリックする。



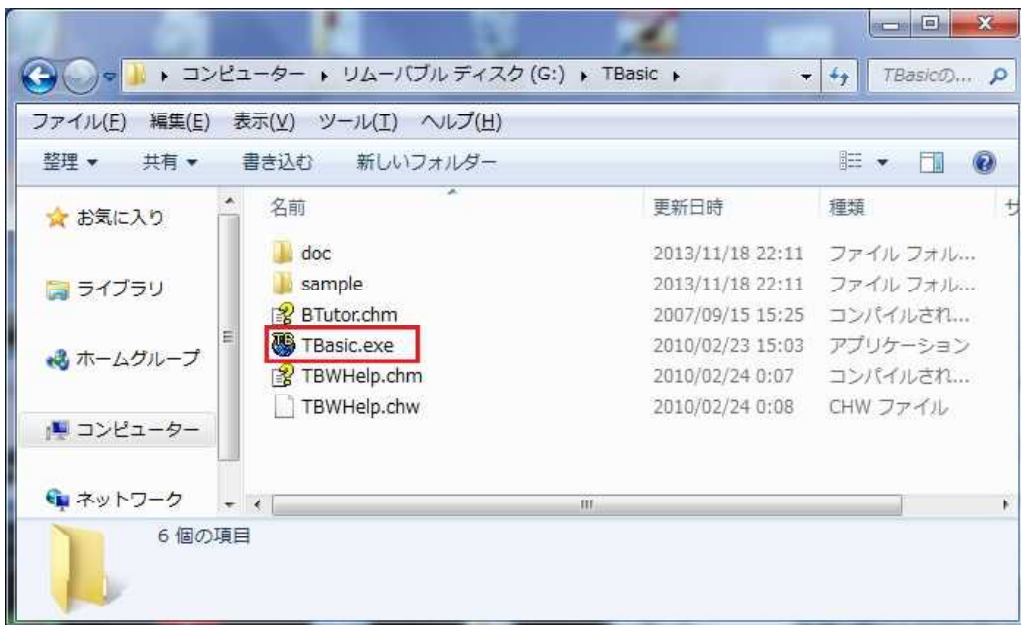
- ③ 表示されたウィンドウ内のリムーバブルディスク (G:) をクリックする。
ドライブGがUSBメモリに対応づけられている。



④表示されたウィンドウ内のフォルダ (TBasic) をダブルクリックする。



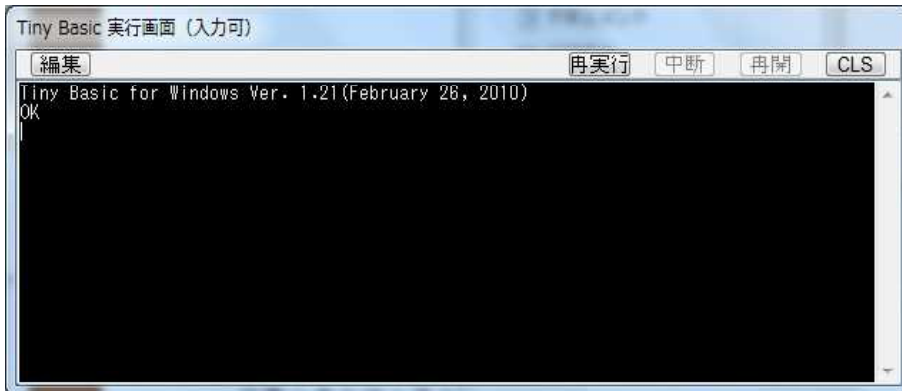
⑤表示されたウィンドウ内のファイル (TBasic.exe) をダブルクリックすると、「Tiny Basic」が起動される。



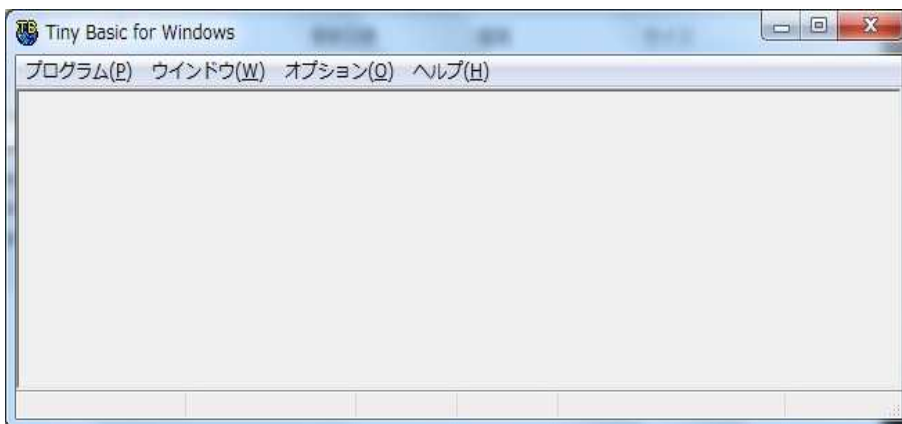
● 起動

Tiny Basicが起動後、「Tiny Basic 実行画面 (入力可)」画面と「Tiny Basic for Windows」画面とが表示される。


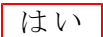
「Tiny Basic 実行画面 (入力可)」画面



「Tiny Basic for Windows」画面



● 停止

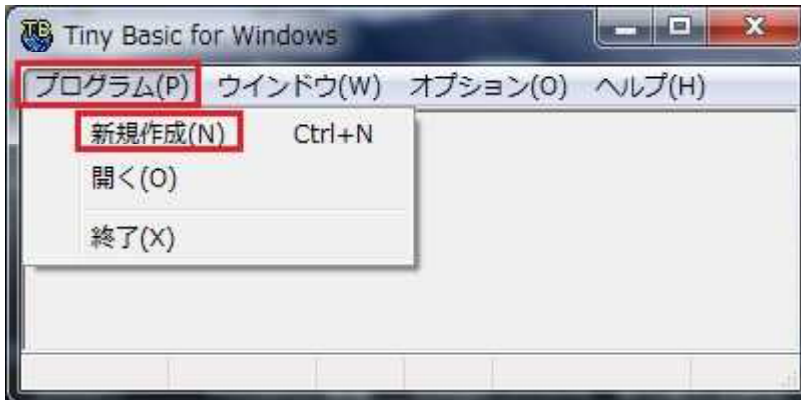
停止するには、「Tiny Basic for Windows」画面の右上隅にある  をクリックする。すると、「確認」ウィンドウが表示されるので、  ボタンをクリックする。



1. 3 「Tiny Basic」の実行

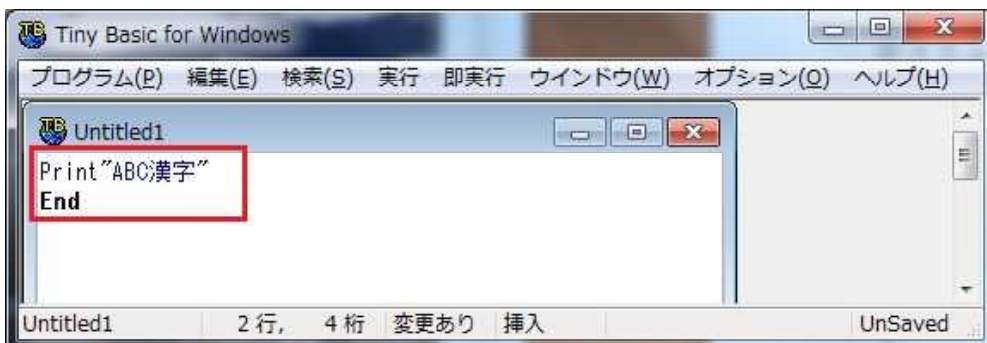
●新規ファイルの作成

- ①メニューバーの「プログラム」をクリックし「新規作成」を選択すると、「Untitled1」ウィンドウが表され、ウインドウ内にカーソル(|)が点滅する。



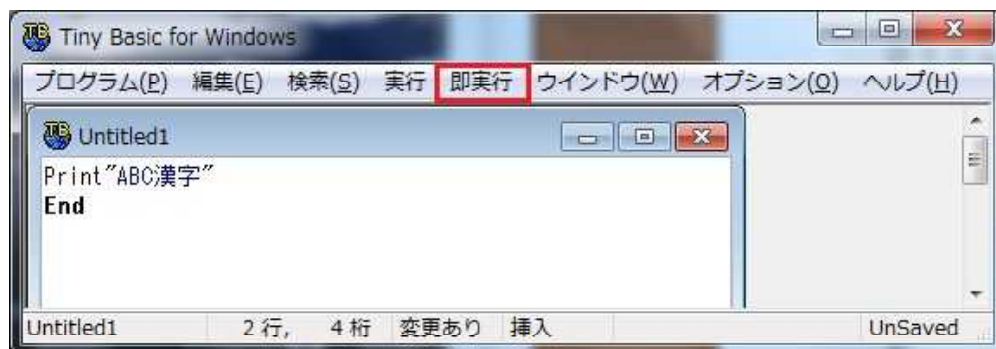
キーボードから入力した文字はカーソルの位置に表示される。
カーソルの移動は、矢印キー（↑↓←→）で行う。
このウインドウ内にBasicプログラムを入力する。

簡単なプログラムを入力する。



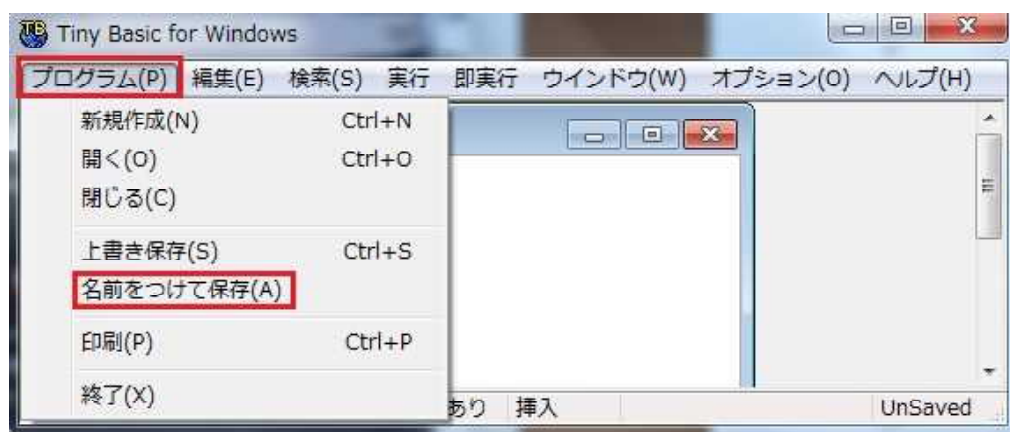
●プログラムの実行

- ②メニューバーから「即実行」をクリックすると、実行結果が「Tiny Basic 実行画面（入力可）」画面内に表示される。



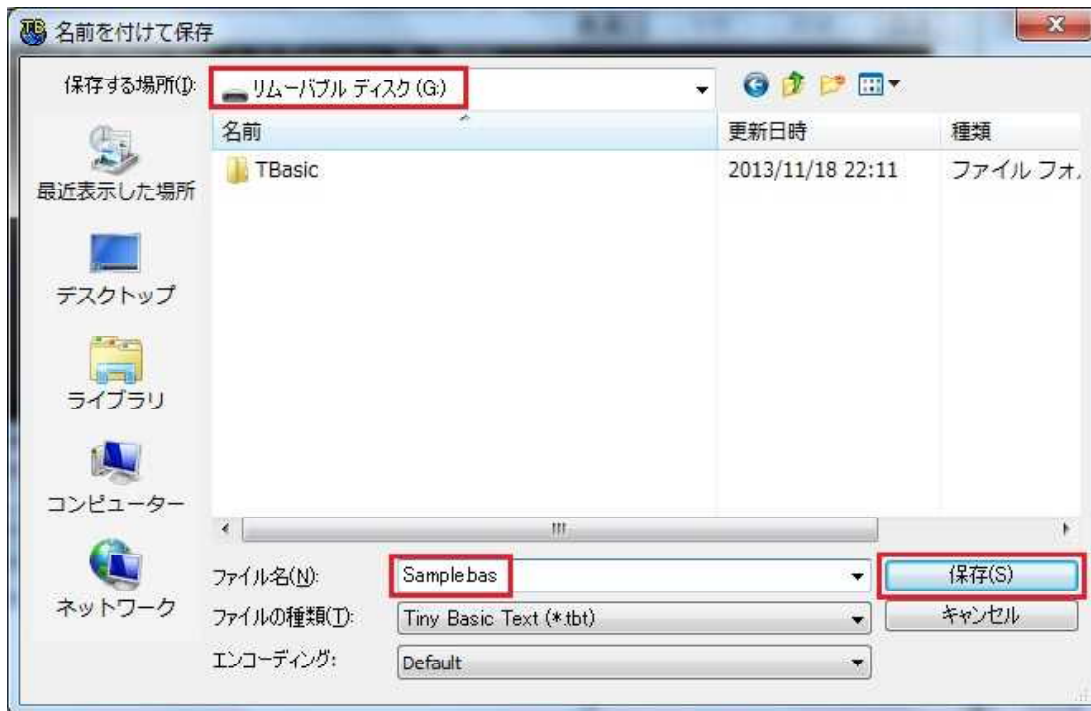
●プログラムの保存

- ③メニューバーの「プログラム」をクリックし、「名前を付けて保存」を選択する。



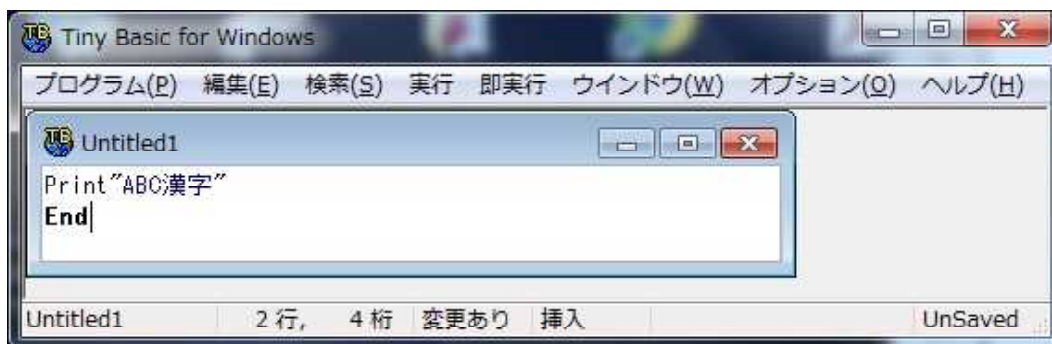
「名前を付けて保存」ウィンドウが表示されるので、保存する場所（リムーバブルディスク (G)）と、ファイル名 (Sample. bas) を指定して、「保存」ボタンをクリックする。

Basicプログラムを保存するファイルには、拡張子「.bas」をつけておく。



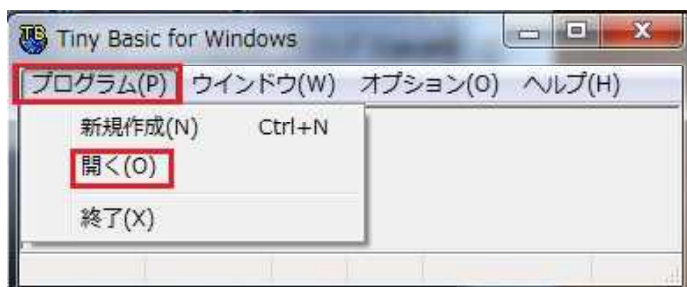
●プログラムの消去

- ④ 「Untitled1」 ウィンドウ右上の **×** をクリックすると、「Untitled1」ウィンドウが消え、同時に編集中のプログラムも消去される。



●プログラムの修正

- ⑤ 作成したBasicファイルを修正する場合、Tiny Basicを起動後、「Tiny Basic for Windows」画面のメニューバーの **プログラム** をクリックし、**開く** を選択する。

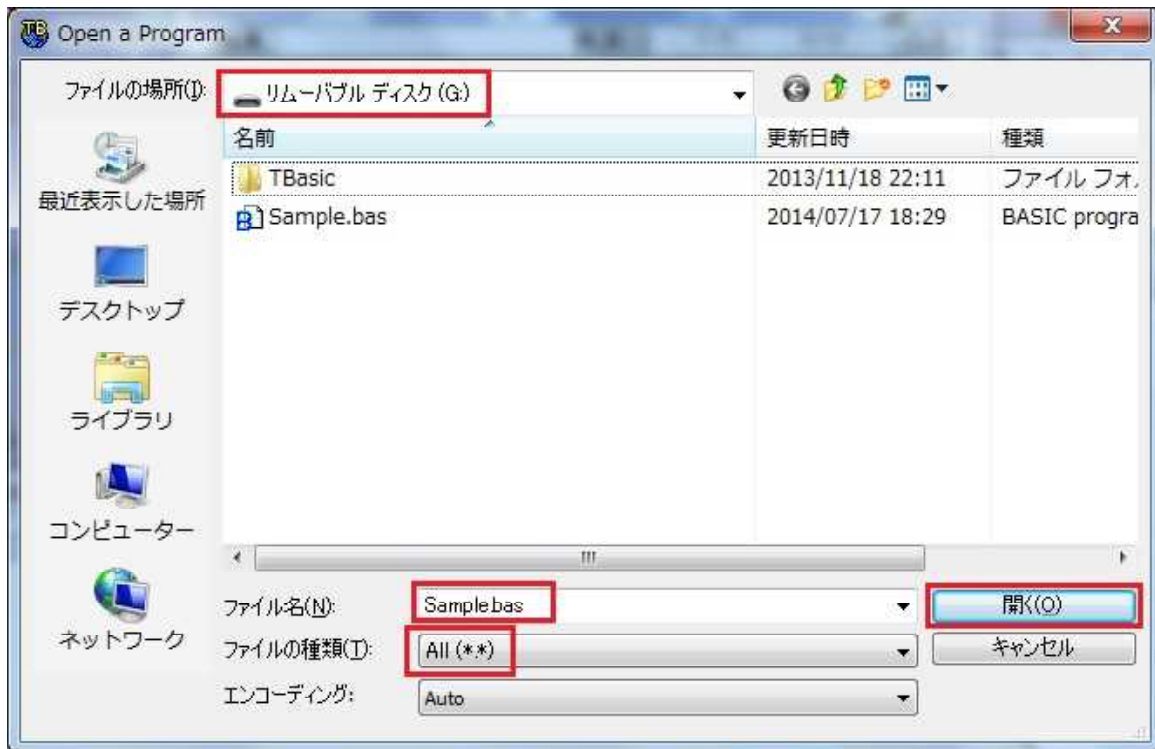


「Open a Program」ウィンドウが表示される。

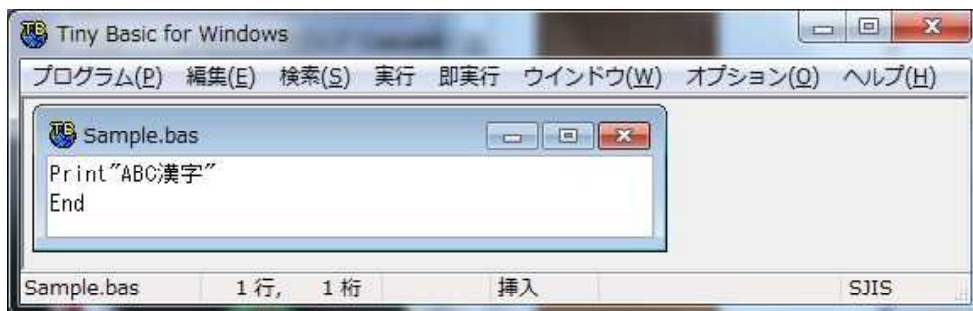
ファイルの種類を **All (*.*)** にしておく。

ファイルの場所（リムーバブルディスク (G)）と、ファイル名 (Sample.bas) を指定して Sample.bas をクリックし、ファイル名欄に表示しておく。

開く ボタンをクリックする。



⑥ 「Tiny Basic for Windows」画面にファイル (Sample.bas) が表示される。



⑦ 「Sample.bas」ウィンドウ内で修正を行う。

●プログラムの印刷

⑧ メニューバーの **プログラム** をクリックし、**印刷** を選択する。

「印刷」ウィンドウが表示されるので、必要な指定を入力後、**OK** ボタンをクリックする。

(プログラム作成・修正作業時の注意事項)

- 最初は、英数字等が入力できる状態になっている。
漢字の入力をするには、**半角/全角**キーを押す。
- 漢字の入力は、ローマ字で入力し（ひらがなで表示される）、最後に**スペース**キーを押すと漢字に変換される。
正しく変換されたなら、**Enter**キーを押す。
正しくない場合、正しい漢字が現れるまで**スペース**キーを押して探す。
ひらがなのままでよい場合は、**スペース**キーを押さずに**Enter**キーを押す。
漢字入力を解除するには、再度、**半角/全角**キーを押す。
- 間違って入力した文字は、削除したい文字の直前にカーソルを移動し、**Delete**キーを押すか、削除したい文字の直後の位置にカーソルを移動し、**BackSpace**キーを押して行う。

キー	機能
Delキー	<p>カーソルの直後の文字を消し、以降の文字列全体を1文字分左に詰める。</p> <div style="border: 1px solid black; padding: 5px; display: inline-block;"> <pre>abc defg Delキーを押す。 abc efg</pre> </div> <p> はカーソルを意味する。</p>
BackSpaceキー	<p>カーソルの直前の文字を消し、以降の文字列全体を1文字分左に詰める。</p> <div style="border: 1px solid black; padding: 5px; display: inline-block;"> <pre>abc defg BackSpaceキーを押す ab defg</pre> </div>

1. 4 簡単なプログラム

簡単なプログラムでBasic言語を紹介する。

プログラムに書かれた順に実行される。

●例1：出力処理

	プログラム	解説
1	' << EX111.bas >>	<ul style="list-style-type: none"> ・注釈は、「'」で始まり行末まで。実行に無関係である。 ・空行はプログラム中どこにでも入れることができ、プログラムを見やすくするために使われる。
2	Print"Basic"	<ul style="list-style-type: none"> ・Print文で画面に文字列Basicと出力する。文字列は2重引用符「"」で囲む。 ・二重引用符を入力するときは、半角文字を入力する状態であることを確認すること。全角文字の二重引用符「”」と間違えないように。間違えたときは、半角/全角キーを押して、半角入力状態に切り替える。
3	Print"こんにちは:";	<ul style="list-style-type: none"> ・セミコロン(;)をつけると、改行は起こらない。セミコロンがないと改行される。
4	Print"今日は"	
5	End	<ul style="list-style-type: none"> ・プログラム実行の終了を意味する。

実行結果

```
Basic
こんにちは:今日は
OK
```

●例2：変数と演算処理

	プログラム	解説
1	' <<EX121.bas >>	<ul style="list-style-type: none"> ・プログラム実行中に値が変わるデータを変数という。変数には名前をつけることができ、それを変数名という。 ・変数には、数値を扱う数値型変数と、文字列を扱う文字型変数がある。 ・数値型変数の名前は、英字から始まる英数字で構成される。文字型変数の名前は、英字から始まる英数字で構成され、末尾に\$を付ける。
2	A=123	<ul style="list-style-type: none"> ・代入文（変数=式）という。 ・Aは数値型変数。 ・B\$は文字型変数。
3	B\$="abc"	
4	A=A+456	<ul style="list-style-type: none"> ・「=」は、右辺の演算結果を左辺の変数に代入する。 ・数値型変数における「+」は加算を意味し、文字型変数における「+」は、文字列の連結を意味する。
5	B\$=B\$+"def"	
6	Print A,B\$	<ul style="list-style-type: none"> ・数値型変数Aと文字型変数B\$の値を出力する。
7	End	

実行結果

```
579          abcdef
OK
```

●例3：入力処理

	プログラム	解説
1	' << EX131.bas >> ,	<ul style="list-style-type: none"> ・Read文で数値型変数Aに数値データを文字型変数B\$に文字列を読み込む。 ・データはData文に読み込む順に記述する。
2	Read A,B\$	
3	Data 123,"abc"	
4	Print"A=";A	<ul style="list-style-type: none"> ・数値型変数Aの値を出力する。 ・文字型変数B\$の値を出力する。
5	Print"B\$=";B\$	
6	End	

実行結果

```
A= 123
B$=abc
OK
```