

SYU JOURNAL

VOL.0



HELLO WORLD

皆さんお疲れ様です。

文系出身で算数がまったく得意でないプログラミング初心者のSYUTENGUです。

つい最近小学生の娘に帯分数と整数の掛け算の間違ったやり方を教えてしまって怒られました💧

そんな記憶もまだ新しいなか、なんと今日は「**プログラミングのすゝめ**」をテーマに

SYU JOURNAL VOL.0をお送りします。

そもそもプログラムとは？とまったく未経験の方もいらっしゃるかと思いますが、

一言で説明するとPCが行うべき処理を順序立てて記述した作業指示書です。

例えば炊飯器に組み込まれている各種レシピを実現するための作業手順が身近な事例になります。

(PC上で)同じ作業の繰り返しが発生するとまず我々の頭の中でその手順を整理します。

さらにそれをPCが理解できるプログラミング言語でプログラム(作業手順書)にまとめます。

それ以降同じ作業の繰り返しであれば全部PCに任せられます。

しかもIT技術の進歩と普及に恵まれてこの実現過程はかつてないほどハードルが低くなってきました。

今日は具体的な事例を一つ取り上げて、プログラミングの魅力[?]をご紹介します。

作業準備

本番作業を開始するまえに用意したいものが二つだけ、制御台と作業台です。

A制御台とは最終的にプログラムを実行するため、コンピュータと会話(入力INPUTと出力OUTPUT)するユーザーインターフェース(UI)です。

今回はブラウザGoogleChromeに内蔵のChromeDevToolsの「**コンソール**」(CONSOLE)を使います。

B作業台とはプログラムの内容(コード)を書く場所です。

今回はWindowsPCの使用を想定して「**メモ帳**」を使います。

メモ帳の開き方・使い方は恐らく皆さんよくご存じかと思いますので、割愛します。

制御台 CONSOLE

下記の手順で準備します。経験のある方はこのページと次のページをとばしてください。

⑥プログラミングの世界では基本半角文字を用いて英数字と記号の入力をします。

①ブラウザ(GoogleChrome)のURL欄に「`about:blank`」と入力し改行(ENTER)して現在タブを空にします。

②ファンクションキーF12を押すかメニュー→その他のツール→デベロッパーツールをクリックします。

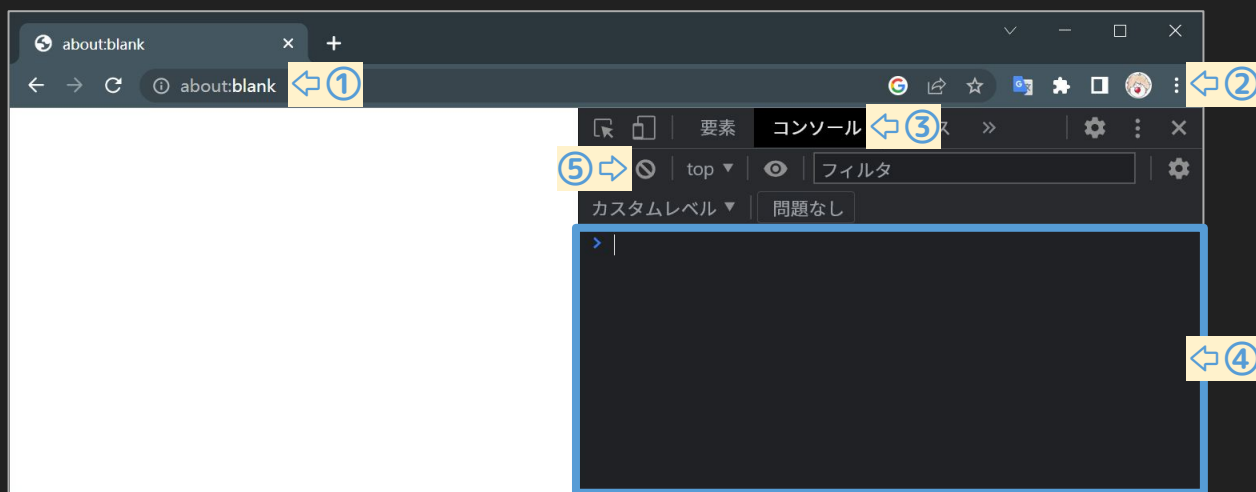
ウィンドウ右側にサイドバーの形で表示されるのはChromeDevToolsのUIです。

③左から2番目のタブ「コンソール」をクリックします。

④入出力が行われ、PCと会話する場所です。

⑤🚫をクリックするかCTRL+Lを押すと④のなかの内容がクリアされます。

もし④に何か既存の内容が表示されていればまず⑤の操作を実行して制御台の上をきれいにします。



制御台 CONSOLE

④で以下のような入力(INPUT)をすれば、基本的な算数演算が実行され、答えが次の行(LINE)に出力(OUTPUT)されます。

キーボードで「1+2」を入力して改行(ENTER)押すと、「3」が出力されます。

```
> 1+2
```

```
< 3
```

キーボードで「2*3」を入力して改行(ENTER)押すと、「6」が出力されます。

```
> 2*3
```

```
< 6
```

キーボードで「2**4」を入力して改行(ENTER)押すと、「16」が出力されます。※べき乗

```
> 2**4
```

```
< 16
```

キーボードで「1/3」を入力して改行(ENTER)押すと、「0.3333333333333333」が出力されます。

```
> 1/3
```

```
< 0.3333333333333333
```

課題提起「消費税計算」

社会人の皆さん、仕事のなかで「消費税の計算」はほとんど経験されているかと思います。
四則演算レベルの単純な算数問題で、計算方法は下記の通りです。

$$\text{消費税額} = \text{税込み金額} \times (\text{消費税率}^{\ast} \div (\text{消費税率} + 100))$$

※計算の便宜上、消費税率はすべて「%」を除く数字部分を直接用います。以下同じです。(e.g.「8%」の場合は「0.08」ではなく「8」)
e.g.スーパーで税込108円のアイスを一個買いました。消費税は 8円 = 108円 \times (8 \div (8 + 100)) です。

日常で電卓の税計算機能やExcel関数、ウェブアプリ等を用いて処理されている方も多いかと思います。
今回は道具を変えて、自作のプログラムで計算して答えを出してみませんか。

まずはコンソールを使って具体的に計算させてみましょう。

今回は税込11,000円の洋服の消費税額を計算してみます。10%の消費税率が適用されます。

前記の計算方法に従って税込み金額のところに「11000」を代入し、消費税率のところに「10」を代入します。

括弧さえつければ数式全体を一行にまとめて入力しても大丈夫です。

算数演算子や括弧の優先順位は基本的に小学校で習った算数通りです。

入力の最後は改行で入力内容を確定させることをお忘れなく。

```
> 11000*(10/(10+100))
```

```
< 1000
```

消費税額の計算結果「1000」が返り値として次の行に出力されます。

はい、これで完成です。一行でも立派なプログラミングです。

これじゃ暗算や電卓、Excel関数、ウェブアプリを使ったほうが早いよとの異見もたぶん出るかと思います。

しかし個人的には他人の作ってくれたアプリよりもネット環境一切いらずにPCや携帯の電卓アプリを起動する手間もなく、いつも開いているブラウザでただF12キーを押すだけで基本的な算数演算ができるほうが比較的便利だと感じます。

作業台 CODE EDITOR

でもこれしかできないとあまり面白くないですね。

プログラミングの面白さの一つはどんどん改善の蓄積ができることです。

課題に戻りますが、軽減税率の適否で税率が異なるのと、毎回数式全体の入力が非効率だと思う方がきっとおられます。

それなら、税込み金額、税率を入力するだけで税額を算出できる自動処理装置(関数)を作しましょう。

関数の概念については次のページで説明しますが、まずその前に作業台の登場です。

コードを一行に纏めるのは別に処理側(PC)にとっては難がないですが、

コードを書く人間にとって読み書きにくだけなので、

複数行のコードを記述・編集できるよう、B作業台を使います。

「**メモ帳**」等お手元にあるテキストエディタを開いてください。

関数と引数 FUNCTIONS & PARAMETERS

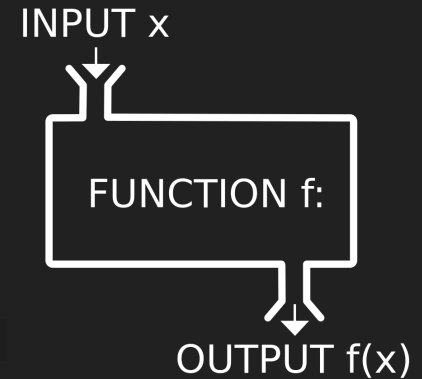
関数(FUNCTION)の概念についてWIKIPEDIAを読むほうが速いと思いますが、
喩えで説明するとデータ処理するためのブラックボックスです。

そういえばむかし日本語でもFUNCTIONのことを中国語の「函数」で呼んでいたね。
関数はまさにデータ処理の「函(はこ)」そのものです。

数学の授業で学んだ $f(x,y)=x^2+y$ のような関数式のことを思い出してみてください。

x,y は関数に数値を入力するための未知数(文字記号)で、**引数**(PARAMETER)と呼ばれます。

例えば引数 x,y にそれぞれ2と1を代入すれば $f(2,1) = 2^2+1 = 5$ となります。



プログラミングの世界でもこの関数の仕組みを利用して手順記述の省力化を図っています。

```
function 関数名(引数名1,引数名2,...){
```

```
    関数の中身
```

```
}
```

関数記法は下記の通りです。

「function」と半角スペースの直後に関数名を記述します。

関数名の直後に「()」を記述します。()の中に「,」(半角カンマ)で区切った引数名を記入します。

()の直後に「{ }」を記述します。{ }の間に空行を挿入します。

空行に関数の中身を記述します。中身が複雑で一行に纏められない場合、自由に行を増やしてください。

では、早速この関数の仕組みを利用して、オリジナル関数「消費税計算」を作ってみましょう。
前記の通り、まずは関数の容器・空箱から書きます。
開いたメモ帳に下記の内容をそのまま記入します。
漢字仮名以外、すべて半角文字で入力してください。基本的に単語と単語の間には半角スペースを入れます。
「function」直後の半角スペースも含めて慣れていないうちはそのままコピペしても大丈夫です。

```
function 消費税計算(金額,税率){  
  
  
  
}
```

今回の関数(FUNCTION)名を課題名の通り「消費税計算」と名付けます。
計算に必要な引数(PARAMETER)は二つあります。税込み金額と適用される消費税率です。
それぞれ「金額」 / 「税率」と名付けます。
関数名も引数名も(一部特例を除いて)基本的に名付けが自由です。
今回は分かりやすいように全角文字の漢字仮名にしました。
これで空っぽの関数容器ができました。

それから、中身の処理コードを `{ }` の間に記入します。

中身の内容作成は簡単です。さきほど消費税の計算方法に従ってA制御台(コンソール)に直接入力した「`11000*(10/(10+100))`」の具体的数値を決めた引数名に書き換えるだけです。

「`11000`」を「`金額`」に、そして「`10`」→「`税率`」に変えます。

さらに関数処理を終わらせ処理結果を返すため処理コード行の冒頭に「`return`」を付け加えます。

「`return`」の直後の半角スペースもお忘れなく。

```
function 消費税計算(金額,税率){  
    return 金額*(税率/(税率+100))  
}
```

これで中身も記入され、自作関数「`消費税計算`」の完成です。

制御台でこの関数を使ってみましょう。

A制御台(コンソール)に戻って、B作業台(メモ帳)に記載のコードをコピーして改行(ENTER)します。

「**undefined**」※と出力されますが、気にする必要がないです。

※未定義、コードをメモリに読み込んだだけで返すもの無しという意味です。

```
> function 消費税計算(金額,税率){  
    return 金額*(税率/(税率+100))  
}
```

< **undefined**

すると、関数がPCのメモリに読み込まれます。これで実行の用意ができました。

コンソールで試しに「消費税計算(11000,10)」を入力して改行(ENTER)すると、関数「消費税計算」が呼び出されて{}内のコードが逐行処理されます。

最後に処理結果「**1000**」は返り値としてコンソールに出力されます。

```
> 消費税計算(11000,10)
```

< **1000**

上記のように関数「消費税計算」の引数「**金額**」と「**税率**」にいろんな数値を代入して計算してみてください。

入力数値によっては割り切れない小数が返されることもあって、少し不都合ですね。

また別の機会でそのあとの端数処理機能の実装等もご紹介できればと思います。

Q&A

プログラムの保管

B作業台(メモ帳)に記入した関数は任意のファイル名(e.g. 「abc.txt」)をつけて保存してください。

A作業台(コンソール)に入力してメモリに読み込まれるまで、コードはただのプレーンテキストに過ぎないです。自分で書いたコードをやりやすいように編集・保管できて簡単に使いまわせるのが便利ですね。

メモリ内の関数

一度コンソールに関数のコードを入力すると、その関数がメモリに読み込まれてブラウザが閉じられるまで（または同名関数・変数に上書きされるまで）一時保存されます。ですのでブラウザを開いたままの状態では、毎回の関数実行にコンソールへの関数貼り付けは必要がないです。しかし作業台で**関数の内容が更新されたらもう一度関数全体をコンソールに貼り付けて**メモリ内に一時保存されている関数の内容を更新する必要があります。※

※コンソールはあくまでも実験台のような存在で主に処理結果の確認に使われます。本格的にコードを書く場合、関数や変数、定数をメモリへの読み込み方はコンソールと違いますが、更新に伴う再読み込みが必要であることは同じです。