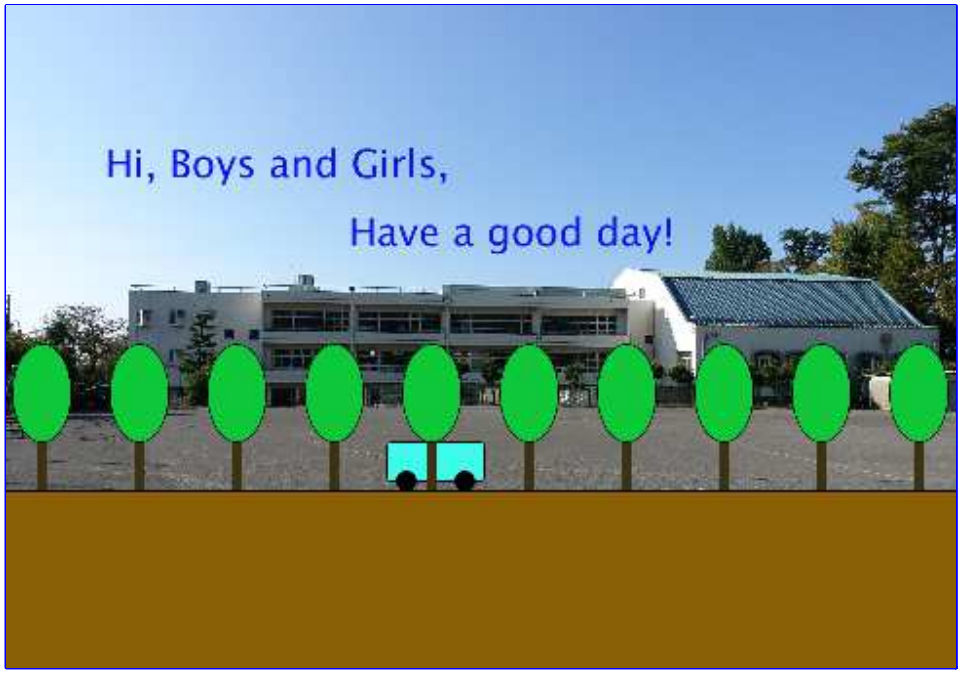
Processing

第１回

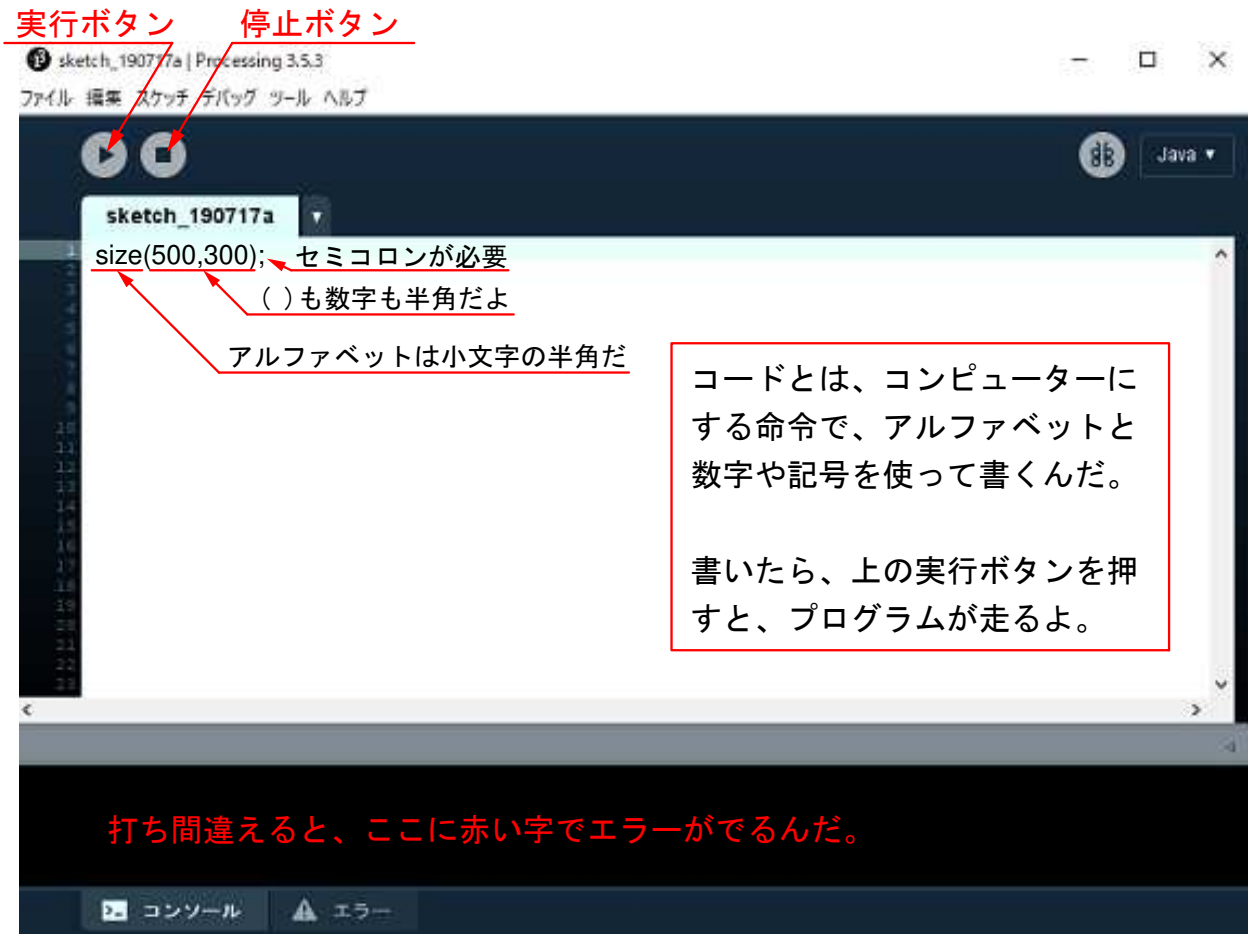


松田小学校／寄小学校

1ーステップ　1：プロセッシング入門

プロセッシング＝処理するとか、加工するって意味

プロセッシングとは、本格的なプログラム言語をつかって、
デザインやアニメーションを描くためのソフトウェアだよ。



コードとは、コンピューターにする命令で、アルファベットと数字や記号を使って書くんだ。

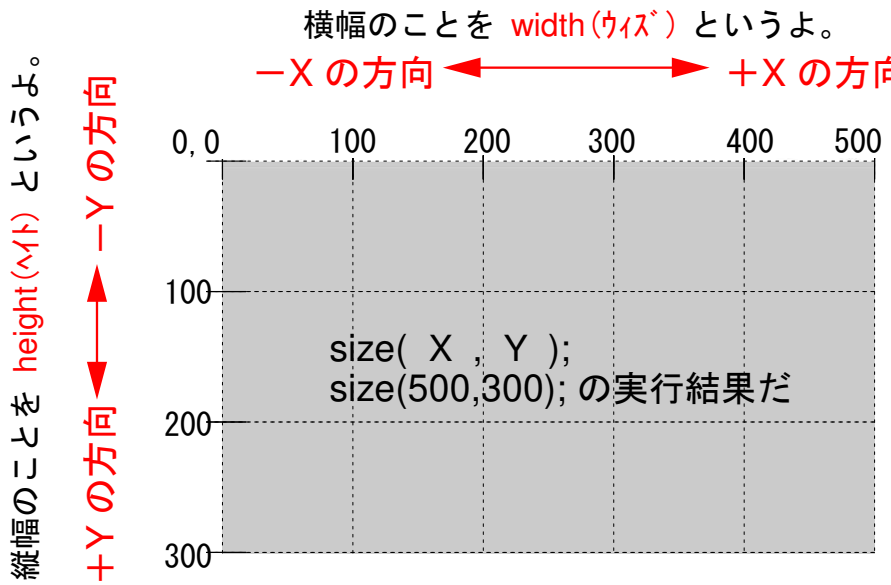
書いたら、上の実行ボタンを押すと、プログラムが走るよ。

エラーについて

左のようにセミコロンがないまま実行すると、エラーとなるんだ。
エラーがでたときは、何が間違っていたか考えてみよう。

セミコロンがないね！　赤い波線がでたら注意。

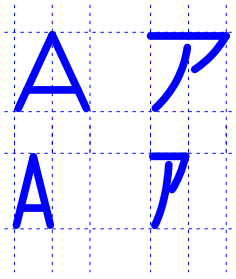
size(500,300); で横幅 500、縦 300のウィンドウ(=画用紙)がかけた。



ウィンドウには、上や左の数字、そして点線は描いてないよ。

文字の大きさには、半角(1ﾊﾞｲﾄ)と全角(2ﾊﾞｲﾄ)があるんだ。
processing では半角でコードを打つよ。

上が全角、下が半角



← このキーで
半角と全角を変えるんだ。
キーボードの上の左端にあるよ。

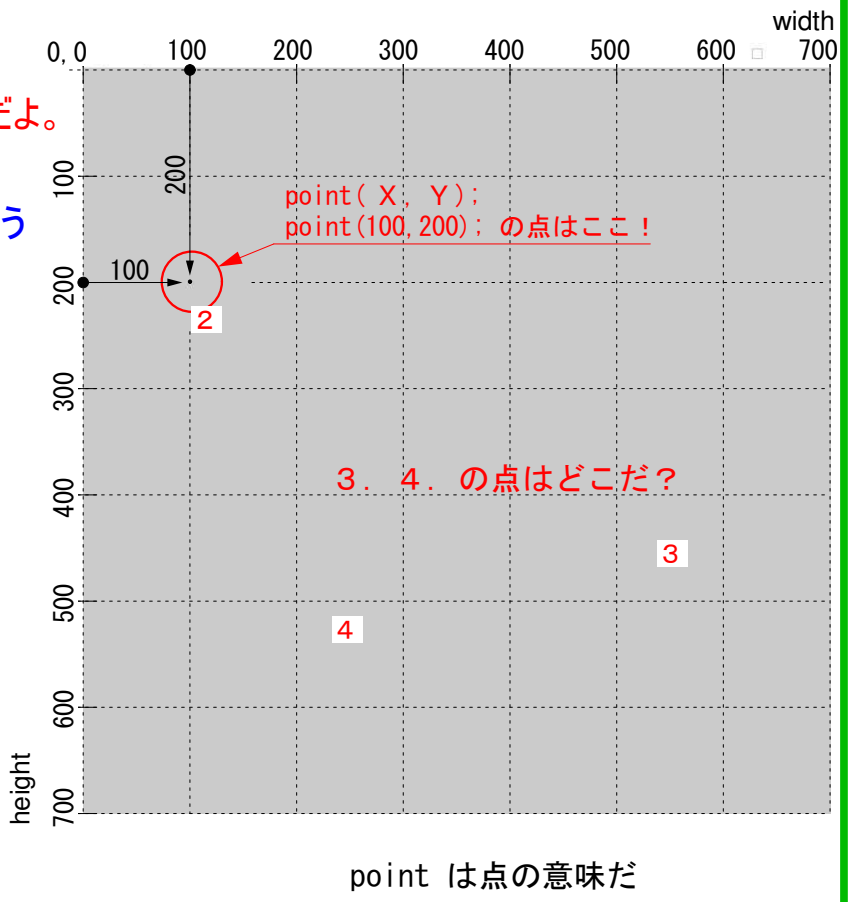
ミッション

1． size(1000, 1000)； のウィンドウを描いてみよう。

2． 前に書いたコードを消して、縦長のウィンドウを描いてみよう。

１ーステップ　２：コードを入力してみよう

size は画用紙の大きさだから、最初に１回だけ書くんだよ。



point は点の意味だ

点のコードを打ってみよう

```
１．size(700, 700);
２．point(100, 200);
```

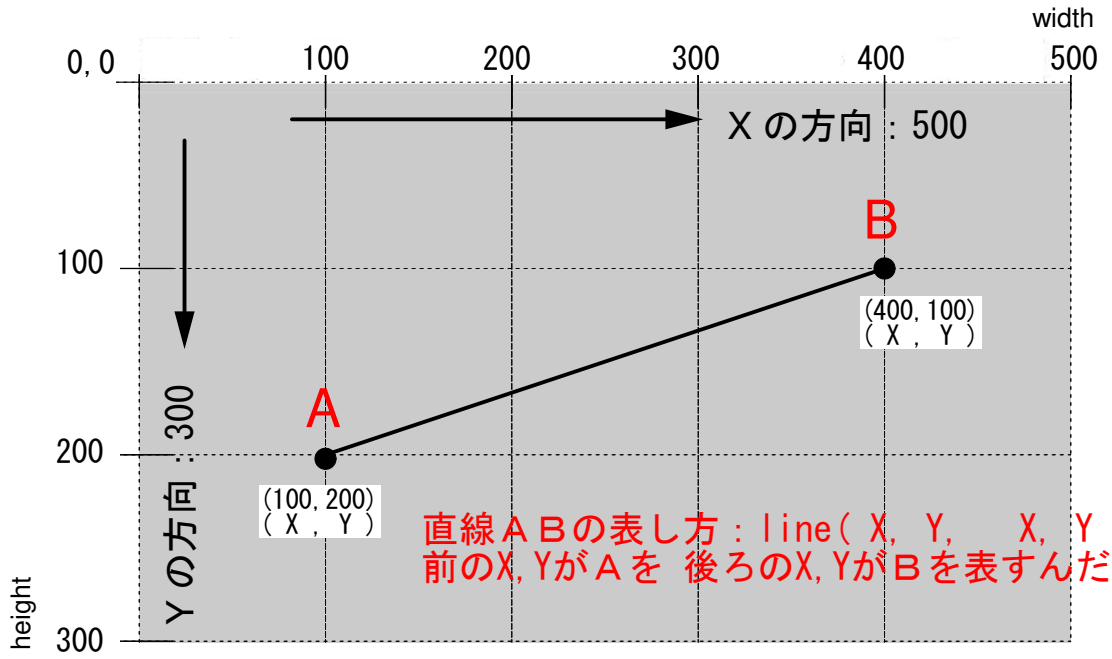
下の３行のコードを書き加えてみよう

```
３．point(600, 500);
４．point(200, 600);
５．point(800, 800);
```

５．point(800, 800);はウィンドウの外になったので、右のウィンドウsize(700,700)には表れてこないよ。

ミッション

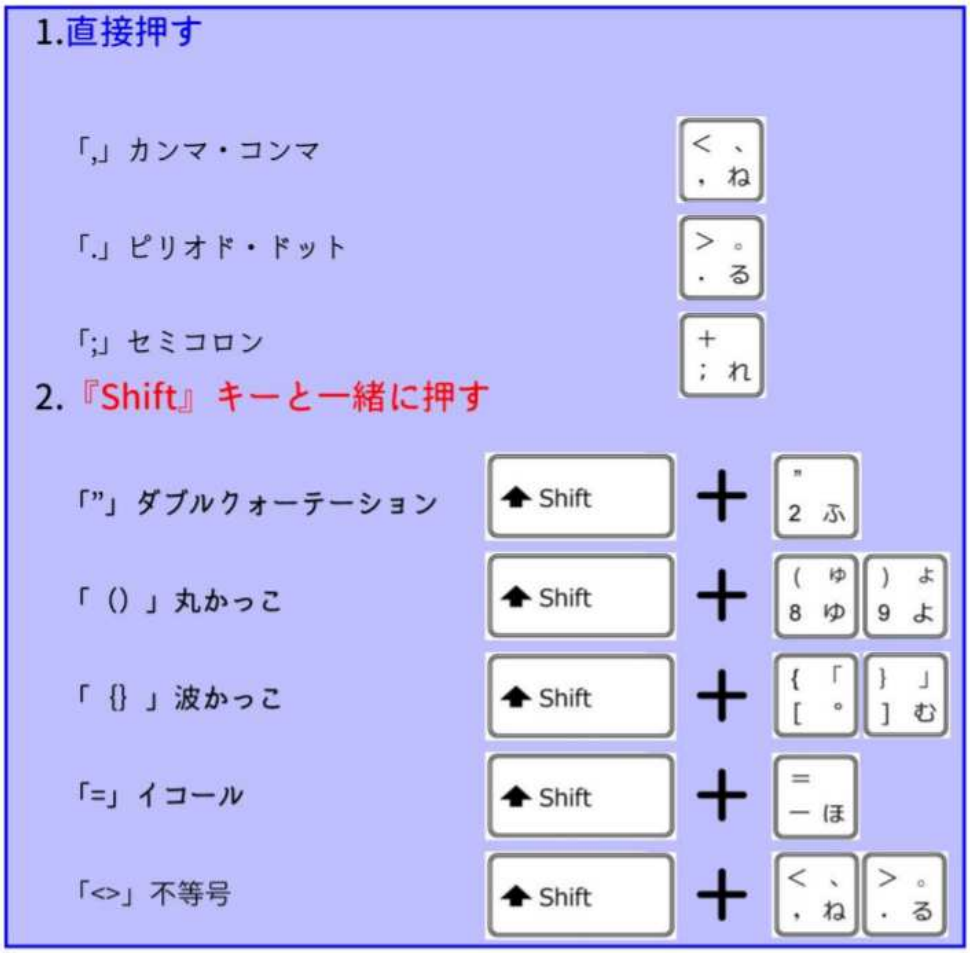
上を参考にして、直線 AB のコードを打って、実行してみよう。



１ーステップ　３：キーボードとマウスの使い方

アルファベットの大文字は、Shift キーを押しながら打つんだ。

特殊なキーは、下の１．と　２．だ。



ミッション　　size(400, 300)； のウィンドで

特殊キーを打ってみよう。エラーがでるけど気にしないでね！

```
1 size(700, 700);
2 line(100, 200, 600, 400);
3 line(600, 400, 300, 500);
4 line(300, 500, 100, 200);
5
```

マウスを使ったコピーの仕方

マウスの左ボタンを押したまま、右か左に滑らすと黄色くなって、範囲指定ができるんだ。
黄色い部分にマウスを置いたまま、右クリックしてコピーを選ぶ。マウスで縦棒の位置を指定し、右クリックして貼り付けを選ぶと、黄色く指定した部分がコピーできるよ。

キーを使ったコピーの仕方

範囲指定できたら、Ctrl＜ｺﾝﾄﾛｰﾙｷｰ＞を押しながら、Ｃをおすとコピーができるし、Ｖをおすと貼り付けができるよ。

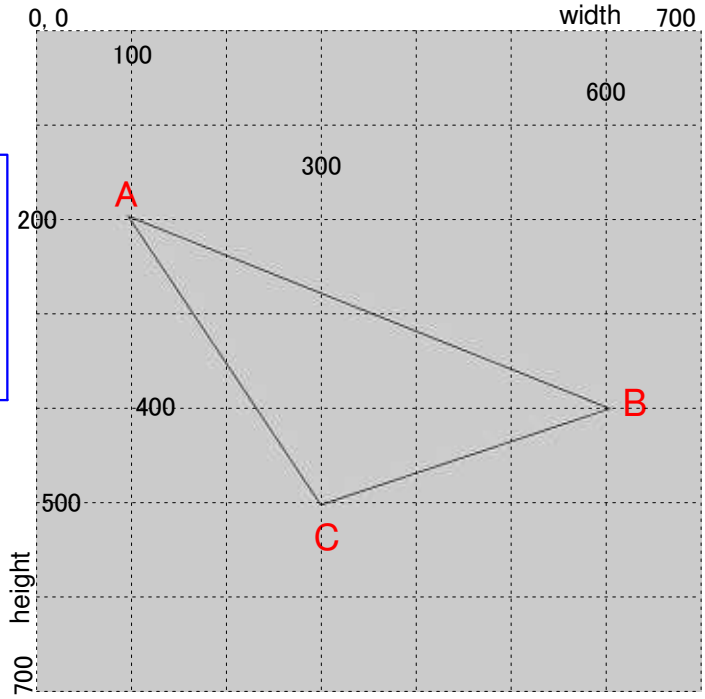
ここで右クリックして貼り付けを選ぶと、黄色い部分がコピーできる。

１ーステップ　４：線と三角形のコードの違いを確認しよう

ファイルから新規を開いて、
下のコードを打ってみよう。

```
size(700, 700);
fill(255, 0, 0);
line(100, 200, 600, 400);
line(600, 400, 300, 500);
line(300, 500, 100, 200);
```

小文字　size　line
大文字　SIZE　LINE
キーボードには大文字が書いてある。



ファイルから新規を開いて、
下のコードを打ってみよう。

```
size(700, 700);
triangle(100, 200, 600, 400, 300, 500);
```

トライアングル　A のX座標　A のY座標　B のX座標　B のY座標　C のX座標　C のY座標

小文字　fill　triangle
大文字　FILL　TRIANGLE

fill（フィル）は満たすって意味だ
fill(255,0,0); は赤で満たせ、
つまり赤く塗れという命令だ。

上のコードに２行目を加えて実行してみよう。

```
size(700, 700);
fill(255, 0, 0);
triangle(100, 200, 600, 400, 300, 500);
```

加えた

A　B　C

triangle1 で保存しよう。

３本の直線でつくる三角形と、triangle という命令で描く三角形の違いが分かったかな？
直線で書いた三角形は面積をもってないから、
fill 命令を使っても内部を赤く塗れないんだ。

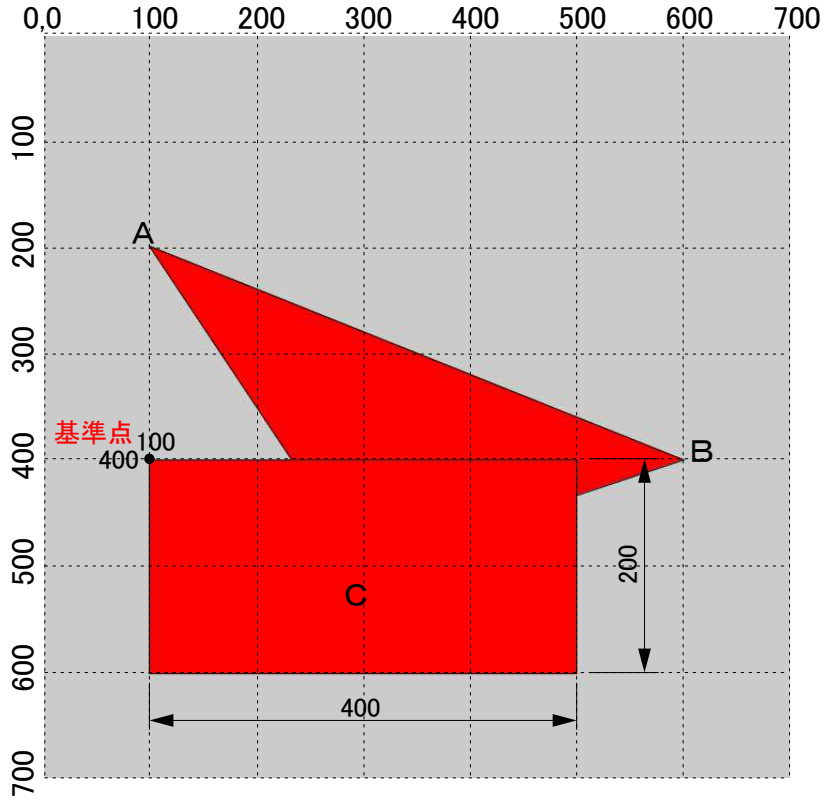
上のコードに５行目を加えて実行してみよう。

```
size(700, 700);
fill(255, 0, 0);
triangle(100, 200, 600, 400, 300, 500);

rect(100, 400, 400, 200);
```

加えた

１ーステップ　５：いろいろな図形を描こう



小文字　rect
大文字　RECT

上のコードに７行目を加えて実行してみよう。

```
1 size(700, 700);
2 fill(255, 0, 0);
3 triangle(100, 200, 600, 400, 300, 500);
4
5 rect(100, 400, 400, 200);
6
7 circle(450, 200, 200);
```

基準点のX座標　Y座標　横幅　高さ

中心点のX座標　Y座標　直径

加えた

レクト　レクタングル
rect は rectangle の省略形で、長方形の意味だよ

上のコードに９行目を加えて実行してみよう。

```
1 size(700, 700);
2 fill(255, 0, 0);
3 triangle(100, 200, 600, 400, 300, 500);
4
5 rect(100, 400, 400, 200);
6
7 circle(450, 200, 200);
8
9 ellipse(300, 200, 200, 300);
```

中心点のX座標　Y座標　横径　縦径

サークル
circle は 円の意味だね。

加えた

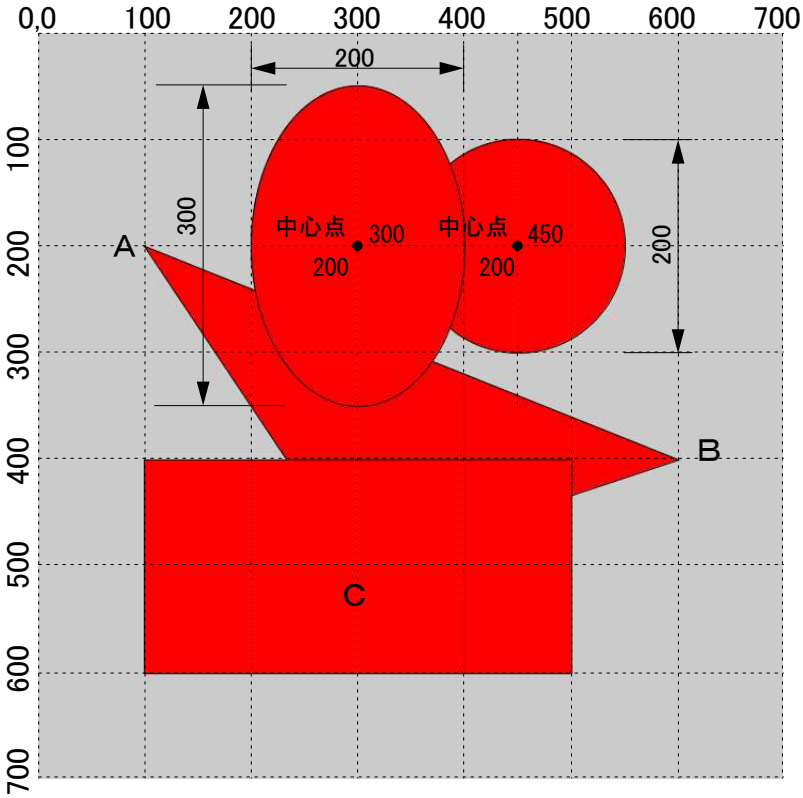
エリプス
ellipse は 楕円（だえん）の意味だね。

triangle2 で保存しよう。

１ーステップ　６：いろいろな図形を描こうの続き

ミッション　１

１. 上のコードの４行目に、
　　fill(255,255,0);
　と書いて実行してみよう。
２. 上のコードの６行目に、
　　fill(0,0,255);
　と書いて実行してみよう。
３. 上のコードの8行目に、
　　noFill( );これだけ大文字だよ
　と書いて実行してみよう。
４. ８行目の末尾にカーソル
　をおいて、Enterを押して
　１行くり下げて、９行目に
　　strokeWeight(10);
　と書いて実行してみよう。
５. ( )の数字を変えると、
　どうなる？



ｽﾄﾛｰｸｳｪｲﾄって読んで、W だけ大文字だよ

triangle3 で保存しよう。

ミッション　２

size(800,600); の新規ウィンドで

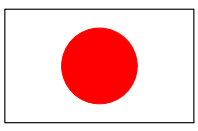
１．line(100, 200, 600, 400) ; と平行な線を描いてみよう。
２．line(400, 0, 400, 600) ; に直交する線を描いてみよう。
３．新規のウィンドウを開いて、日本の国旗を描いてみよう。

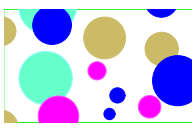
flag で保存しよう。

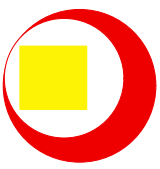
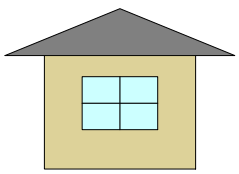
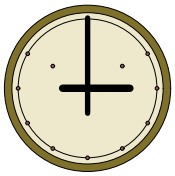
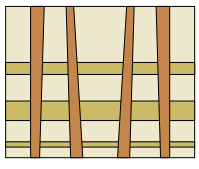
次回の予告

次回は、いろいろな図形に本格的な色をつけてみよう。
国旗も描いてみよう。何か色つきの模様を考えてきてくれるかな！
例を書いておくからね。

Bye-bye!



伴走の方へ　各回共通

この教材は、各回共通でステップ０～２まで（１回目だけはステップ１～３まで）を１時限、ステップ３以降を２時限とおおまかに考えています。タイピングに５分＋α、ステップ０の復讐に５～８分程度を配分して下さい。ミッションは進行具合を見て、抜粋して良いと思います。アンケートは２時限終了後に行って下さい。時間節約のため、ＰＣはボランティアが事前に立ち上げて、プロセッシングも開いておきます。

第１～３回目くらいまではコードに慣れていないので、丁寧な説明が必要です。とりわけ、コードが上から順に実行されることと、第３回の for 文の繰り返し、第４回の条件分岐は詳しく説明して下さい。しかし、４～５回目あたりでコードを打つことに慣れてきます。ですから、これ以降は一斉に説明してから、コードを打つことを繰り返すという進行形態は薦めません。これは児童の集中力を何度も切断してしまうこと気がつきました。

第４回目あたりから授業の最初に簡単な説明をしたら、あとは児童の教材を読む力に任せても大丈夫です。今まで講師は説明を止めて教室中を見ながら、困っている児童を素早く発見する役に徹しました。そして、講師から指示を受けたボランティアたちが、個別的に１対１で問題解決に当たりました。

小学生が自力で教材を読みながら、自分で考えてコードを打つことを、大切なことだと思うようになりました。その考えに従って本ガイドを纏めています。本ガイドはまずコードを書いてみてから、理由を考えるという体裁をとっています。

１０回の授業の中で消化できる範囲に限定しましたが、回を追う毎に情報量が多くなってしまいました。１回を９０分と前提していますが、９０分では納まらない回もあるかと思います。適宜抜粋して授業を進めて下さい。しかし、児童の自立力に任せると、想像以上にすすむものだというのが、４年間の経験によって得られた結論です。以下に、本文では触れることができなかった情報を纏めておきます。

一般項目

① ＵＳＢがＰＣに認識されないと、保存対象ドライブとしてＵＳＢが表示されません。ＵＳＢが認識されるには少し時間がかかり、時間のロスが発生します。それを防ぐため、授業冒頭でタイピングを始める前にＰＣにＵＳＢを差して、ＵＳＢのド

ライバーを読み込ませて下さい。

ＰＣとＵＳＢの接触不良を起こす差込口がありますので、差込口を変える、もしくはＵＳＢにオイルを吹くなどの確認をしてください。オイルの用意があります。

② タイピング中にＰＣがフリーズしたら、モニターの余白部分でマウスをクリックし下さい。それでもダメなら、Ctrl+Shift+Esc　の同時押しでタスク・マネージャーを起動し、タイピングを一度終了させて再スタートして下さい。

③ キーボードの中央やや左にあるＦと、やや右にあるＪには、小さな突起があります。Ｆには左手人差し指を、Ｊには右手人差し指をのせてください。ここをホームポジションと言って、いつもここに指を置く習慣を付けると、タイピングが早くなります。

④ 大文字と小文字の異同を知らない児童がいます。キーボードは大文字で描かれていますが、コードは小文字ですから、大文字と小文字の確認をおこなって下さい。大文字と小文字を書いたカードを渡すのも良いかもしれません。

⑤ ＵＳＢに保存するには、ファイル→名前を付けて保存→ＰＣ→ＵＳＢです。プロセッシングでの保存は、ASCII　文字(半角英数のみ)で行って下さい。－(ハイフン)は、自動的に＿（アンダーバー)に変換されます。プロセッシングは sketch_000.pde の保存形式を推奨していますが、sketch なしで 000.pde だけでも保存できます。

⑥ プロセッシングを終了する前にＵＳＢを抜いてしまうと、警告文が残ってしまいますので、まずプロセッシングを終了します。次に＜ device 取り出し＞をＯＫにした後で、ＵＳＢをＰＣから抜いて下さい。

警告文がでた場合は、Ctrl+Shift+Esc　を同時に押して、タスク・マネージャーを開き、Processing を選んでタスクの終了をすれば、終えることが出来ます。

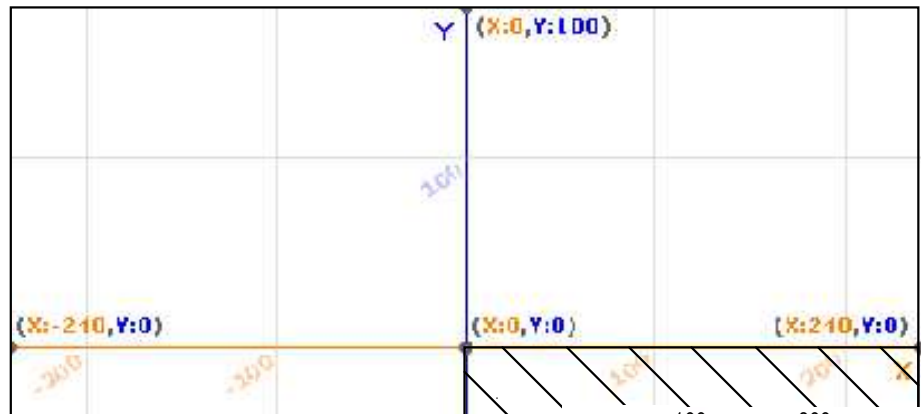
⑦ ＰＣの基本的な操作は、ウィンドウズの一般的なものと同じです。たとえば、保存と読み込み、ファイル管理、コピー＆ペースト(貼り付け)や、大文字と小文字の切り替え、キーボードの配置、やり直し(Ctrl ＋ Z キー)などは、ウィンドウズの一般原則と同じです。マウスで範囲指定して、右クリックでコピー＆ペーストも使えます。

第１回　　座標とコードの理解

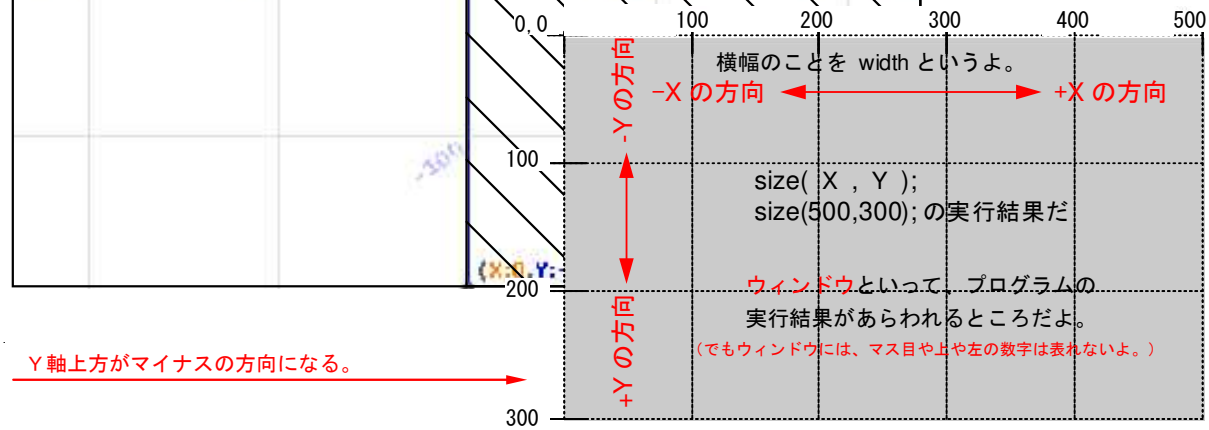
① 座標の理解は不可欠です。５年生でやったスクラッチは、ＸＹ軸をもった直交座標ですから、右側に行くに従ってＸの値は大きくなり、左に行くに従ってＸの値はマイナスがついて小さくなります。Ｙ座標も上にむかって数字が大きくなるし、下は数字は大きくなりますがマイナスが付くので値は小さくなります。(下の白い座標がスクラッチ)

プロセッシングは原則として、スクラッチの右下に相当する部分(下図の斜線部分=第４象限)しか使いません。Ｘ軸こそスクラッチと同じですが、Ｙ軸はマイナスがつかず下に行くに従って、数字も値も大きくなっていきます。そのため反対に、Ｙ軸にそって登る場合は、マイナス方向ということになります。

スクラッチのウィンドウ



プロセッシングのウィンドウ



② プログラムの実行によって表れるウィンドウには、座標値やマス目が書かれていません。本ガイドで書いているウィンドウのマス目や数字は大凡のものです。ページレイアウトの都合で適宜な縮尺で縮小しています。

マス目や座標値はその都度、提示すると理解しやすくなりますから表示していますが、本ガイドではページが違うと、同じ図形でも大きさが違うとこがあり得ます。なお size(X,Y);は、(X － 1,Y － 1)という情報もありましたが、確認できなかったので (X,Y)で描いています。

③ コードを打ち込んでいる途中では、エラーメッセージが頻出しますが、打ち上がるとエラーメッセージは消えます。それでもエラーメッセージが残るときは、コードの打ち間違いですから、丁寧に見直して下さい。

見落としやすいのは、「,」と「.」の間違い、「;」と「:」の間違い、「(　)」と「{　}」の間違いや打ち忘れ、半角の空白と全角の空白の間違い(全角の空白はエラー)などが多いものです。また、命令(コマンド)はスペルを間違えると、色が変わらずに黒いま

までです。

④ 日本語を打ったときウィンドウの文字が文字化けするときは、ファイル→設定を開き、言語を日本語にして、エディターとコンソールのフォントをＭＳゴシックにして下さい。松田小学校のＰＣは日本語に設定済みです。

ステップ ６ ミッション－１

```
size(700, 700);
fill(255, 0, 0);
triangle(100, 200, 600, 400, 300, 500);
fill(255, 255, 0);
rect(100, 400, 400, 200);
fill(0, 0, 255);
circle(450, 200, 200);
noFill();                //塗らない
strokeWeight(10);        //縁線の太さ
ellipse(300, 200, 200, 300);
```

ステップ ６ ミッション－２

```
size(800, 600);
line(100, 200, 600, 400);
line(100, 300, 600, 500);    //平行線は無数にある

line(400, 0, 400, 600);
line(0, 300, 800, 300);      //直行線は無数にある
```

日本の国旗

```
size(900, 700);
rect(100, 100, 700, 500);
fill(255, 0, 0);
noStroke();
circle(450, 350, 240);
```

色を指定しないと白になる。
白地は横：縦は、３：２

図形の縁線を表示しない命令

日の丸は縦の3／5で、位置は中央