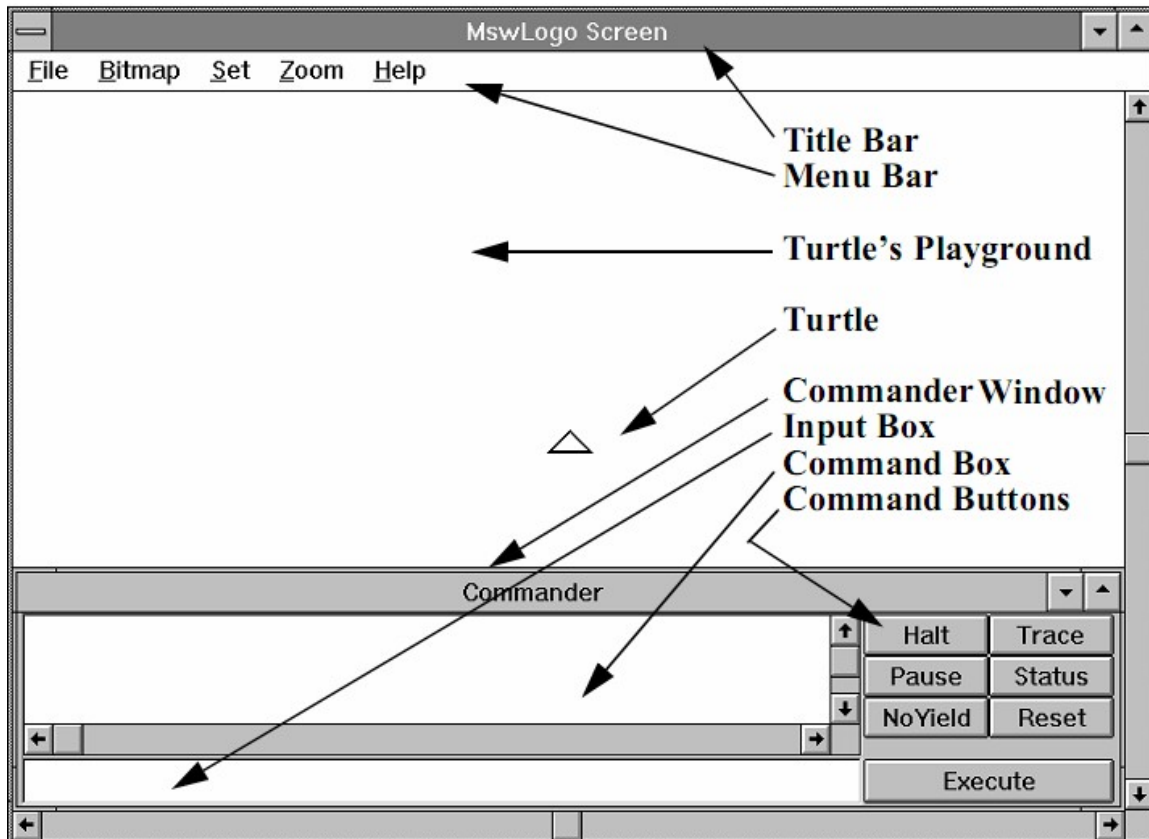


MSW LOGO – TUTORIAL ONE - INTRODUCTION

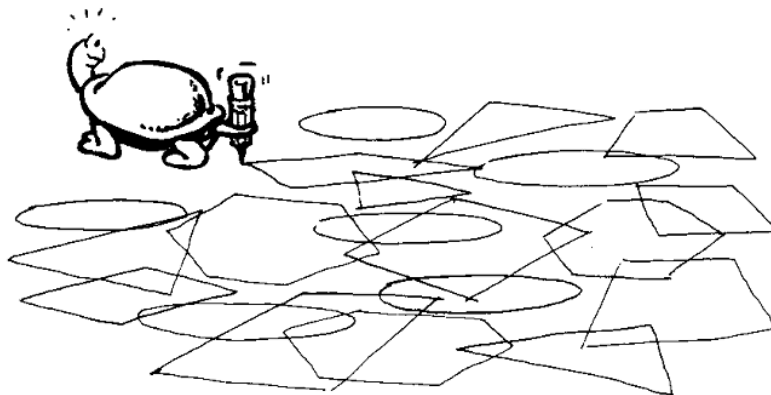
MSW logo is the perfect introduction to computer programming. You will give **commands**, you will draw shapes and you will write **procedures**.

Your LOGO workspace looks like this:



You will type your instructions into the **input box**. Your instructions are recorded in the **command box**.

The turtle can draw shapes with it's pen.



TURTLE DIRECTIONS

The turtle can go **forward**, **back**, **right** and **left**.

Type **FORWARD**, press space and enter a big number like **200**. Press Enter.

Try using the **BACK** command.

To turn the turtle, we have to tell it to turn right or left.

Try : **RIGHT** _____

LEFT _____

Question: What number should you enter to:

- 1) Turn the turtle to face exactly the side of the screen.
- 2) To turn the turtle from facing the top of the screen to facing the bottom of the screen.

So, when using the **FORWARD** and **BACK** commands the turtle moves around the screen, but using **LEFT** and **RIGHT** only turns the turtle and does not move it. Got it?! Practice moving the turtle around the screen!

TURTLE SHORTHAND

We can write Turtle commands in shorthand :

FD =

BK=

RT=

LT=

CS=

THE TURTLE'S PEN



PENUP or PU

What if you wanted to move the turtle without drawing a line? That's easy. You can tell the turtle you want it to put its pen down or pick its pen up.



PENDOWN or PD

The turtle can do three commands with the **pen down**.

Write them next to the following pictures:



HIDING AND SHOWING THE TURTLE

If you don't want to see the turtle, type `HIDETURTLE` or `HT`.

To show the turtle again, type `SHOWTURTLE` or `ST`.

ERASING

What if you make a mistake when drawing a shape?

Type the following:

```
FD 50 RT 120  
FD 50 RT 120  
FD 60
```

What happened and how can it be solved? (check the pictures above...). Just change what the pen is doing. Since you want to erase that last line, you just move the turtle back over it. Try the following:

**PENERASE or PE in shorthand
BACK or BK 60**

Now, try to complete the shape correctly. What happened? What do you need to do?

Task:

Try to write your name using logo commands.

MSW LOGO – TUTORIAL TWO – MAKING SHAPES

Let's think about how to draw a square using MSW logo. Can anyone complete the following instructions:

FD _____ RT _____ FD _____ RT _____
FD _____ RT _____ FD _____ RT _____

But surely there is an easier way than just repeating the same thing four times....yes, there is!

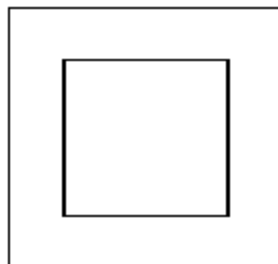


In Logo we can use the REPEAT function. Try this :

REPEAT 4 [FD ____ RT 90]

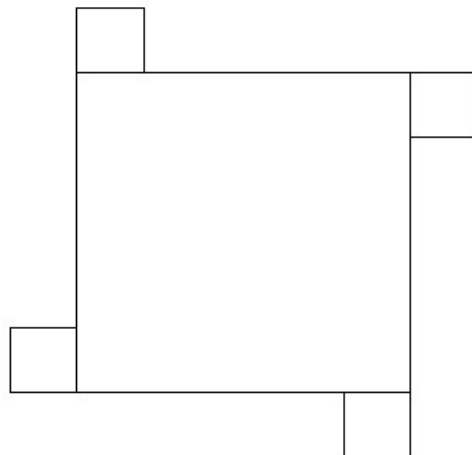
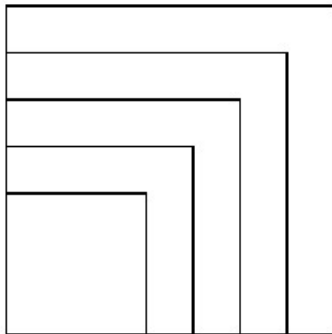
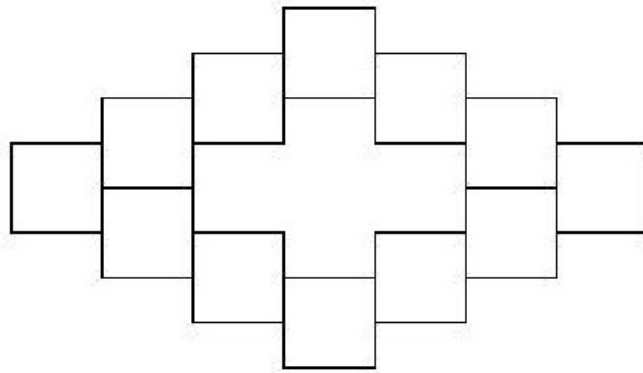
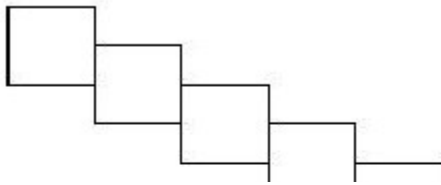
What happens?

Now try to draw this, and use the repeat function.



ADVENTURES WITH SQUARES

What other shapes can you make with squares? Try some of these:



TRIANGLES

Now you can try to draw a triangle using the REPEAT command.

Think about the rules you have learnt in Math about various kinds of triangle :

1. If you have three equal sides, you'll have three equal angles.
2. If you have two equal sides, you'll have two equal angles.
3. If you have no equal sides, none of the angles will be equal.
4. The sum of the three angles in a triangle equals 180 degrees.

RECTANGLES

Who can create a command to draw a rectangle using the REPEAT function? Try this:

```
REPEAT 2 [FD _____ RT 90 FD _____ RT 90]
```

TRY THESE....

```
REPEAT 12 [REPEAT 4 [FD 100 RT 90] RT 30]
```

```
REPEAT 6 [FD 100 REPEAT 6 [FD 10 BK 10 RT 60]BK 100 RT 60]
```

```
REPEAT 6 [FD 100 REPEAT 60 [FD 20 BK 20 RT 6]RT 60]
```

```
REPEAT 8 [RT 45 REPEAT 6 [REPEAT 90 [FD 2 RT 2] RT 90]]
```

You can change the number of REPEATS to make more interesting patterns.

To understand these '**oneliners**', work from the first brackets outwards. See if you can figure out how we made these patterns. Can you make one yourself?

Now try these :

```
PX REPEAT 10000 [FD 200 RT 179]
```

```
PX REPEAT 1000000 [FD 80 BK 80 RT 1]
```

Let's take a look at one of the one-liners as an example:

```
REPEAT 12 [REPEAT 4 [FD 100 RT 90] RT 30]
```

Is this right? Let's see.

Logo starts from the left and reads the first word, the command REPEAT. To run correctly, REPEAT needs a number to tell it how many times to repeat, followed by a list of the instructions to be repeated.

So, Logo reads to the right. Yes, there's the number 12.

So the next step is to look for a list that will tell REPEAT what it is going to repeat twelve times.

The brackets — those are the things that look like square parentheses [] — tell you that the things inside them are a Logo list. In Logo, lists can be groups of words, numbers, letters, or even other lists. Among other things, lists can include spaces.

And that's just what you find after REPEAT 12, another list. There's that REPEAT command again. And, yes, it is followed by the number 4 and a list. The list tells the turtle to do what's inside the brackets — go forward 100 steps and turn 90-degrees to the right.

That's OK. This list is followed by the command RT 30.

So it seems that there's a perfectly good list for the first REPEAT command. This is what Logo repeats 12 times. [REPEAT 4 [FD 100 RT 90] RT 30]

MSW LOGO TUTORIAL THREE : WRITING PROCEDURES

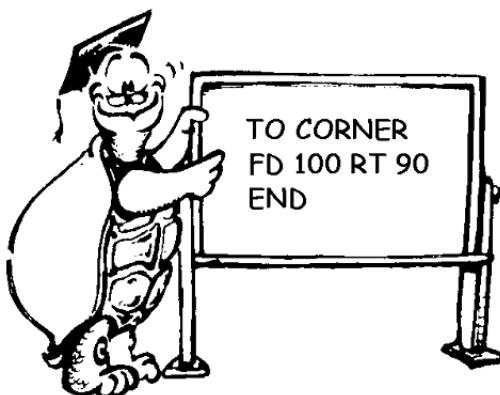
To write a procedure in Logo is to teach the turtle how to draw something.

Let's write a procedure on how to draw a corner:

Type **to corner**.

You will see the to mode dialogue box appear.

Enter:



FD _____ RT 90

Then type **End** to close the to mode dialogue box.

You should the message **Corner Defined** in the commander box.

Now type **Corner** into the command line and see what happens...

WRITING A SQUARE PROCEDURE

Try this : REPEAT 4 [CORNER]

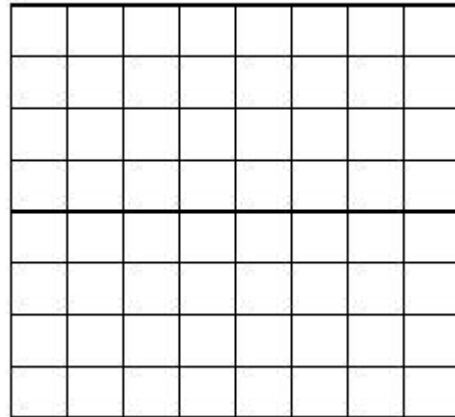
Now how would you write a procedure for a square?

To clear Logo's memory of current procedures, type **erall**.

Activity : Now try to make your own procedure, step by step, to create this checkerboard:

Hints:

Make the sides of the square **20** long.
Now write the procedure for square.
Think about where the turtle finishes after you have 'squared', and how can you use the **repeat** command to first create a column, and then two columns, and then the entire board using one command.

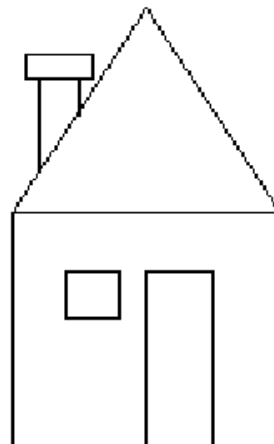


TRIANGLE PROCEDURES

Let's define triangle now as well : Enter

```
TO TRI  
REPEAT 3 [FD 100 RT 120]  
END
```

Now try this :



```
TO HOUSE
SQUARE
FD 100 RT 30
TRI
END
```

Now try writing a procedure that can draw a house like above...maybe add a chimney, a door, a window etc....

Check this one out for some more ideas:

```
TO HUT
REPEAT 4 [FD 60 RT 90]
FD 60 RT 30 FD 60 RT 120
REPEAT 20 [BK 6 FD 66 BK 60 RT 3]
END
```

Check out the **Print** command, try this:

```
REPEAT 12 [REPEAT 4 [FD 100 RT 90] RT 30
PRINT "WOW]
```

SNOWFLAKE

Who can write a procedure for the best snowflake. Here's an example of a snowflake procedure:

```
TO S
REPEAT 5 [FD 10 RT 144]
END
```

```
TO SNF
FD 20 RT 45 S LT 180 S RT 135
END
```

```
TO SNOW
SNF FD 30 RT 45 SNF SNF
FD 10 BK 50 LT 90 SNF
SNF FD 10 BK 50 RT 45
FD 50 LT 45 S RT 45 BK 100
END
```

```
TO STARFLAKE
HT REPEAT 6 [SNOW RT 60]
END
```

TIPS :

1. Draw a snowflake.
2. Pick up the pen and move the turtle to another part of the screen.
3. Draw another snowflake and then move again.
4. Do this several times and the screen will look like a snow storm.

MSW LOGO TUTORIAL 4 : ADDING COLOR

On the screen, you have 256 colors. That is 256 mixes of Blue, Green and Red. Each color has a number. Look at the following:

Red = [255 0 0]

Green = [0 255 0]

Blue = [0 0 255]

What does this tell you? How can we get yellow?

What if a mixture of all colors, so what would the numbers for black and white be?

Black = [] White = []

LOGO COLOR COMMANDS

We have three main color commands :

SETPENCOLOR [number number number] or setpc

SETFLOODCOLOR [number number number] or setfc

SETSCREENCOLOR [number number number] or setsc

You can use the **set** menu to set these properties.

Now load the procedure file colors.lgo from the procs folder, chapter 5. Take a look at the colors that have been defined.

A FLOOD OF COLOR

Try this :

To fillit

```
setpc black
repeat 4[fd 100 rt 90]
pu rt 45 fd 40 pd
setfc blue fill
pu home pd
end
```

Cyan is a light blue color, let's set the screen to cyan, set the pen to red and fill the square with yellow.

```
To fillit2
setsc cyan
setpc red
repeat 4[fd 100 rt 90]
pu rt 45 fd 40 pd
setfc yellow fill
pu home pd
end
```

We can change the size of lines using the SETPENSIZE command. This command takes two numbers, and both must be the same. Obviously, the bigger the number, the fatter the line.

```
SETPENSIZ [500 500] fd 100
```

Too big huh?!

Try this:

```
To fillit3
setsc cyan
setpc red
setpensize [5 5]
repeat 4[fd 100 rt 90]
pu rt 45 fd 40 pd
setfc yellow fill
pu home pd
end
```

Now try this one! First make a triangle :

```
To tri
rt 30 repeat 3[fd 100 rt 120]
end
```

```
To pizzaz
setsc white
setpensize[120 120] setpc purple tri
setpensize[100 100] setpc blue tri
setpensize[80 80] setpc magenta tri
setpensize [60 60] setpc red tri
```

```
setpense [40 40] setpc orange tri
setpense [20 20] setpc yellow tri
setpense [5 5] setpc green tri
end
```

```
To flash
Repeat 20 [cs setsc black pizzaz]
end
```

And finally:

```
TO TRI
REPEAT 3 [FD 100 RT 120]
END
```

```
TO PIZZAZZ2
SETSC CYAN CS HT
PU LT 45 FD 100 RT 45 PD
SETPENSIZ [120 120] SETPC PURPLE RT
PU HOME LT 45 FD 80 RT 45 PD
SETPC BLUE RT 30 TRI
PU HOME LT 45 FD 60 RT 45 PD
SETPC MAGENTA RT 30 TRI
PU HOME LT 45 FD 40 RT 45 PD
SETPC RED RT 30 TRI
PU HOME LT 45 FD 20 RT 45 PD
SETPC ORANGE RT 30 TRI
PU HOME PD
SETPC YELLOW RT 30 TRI
PU HOME RT 135 FD 20 LT 135 PD
SETPC GREEN RT 30 TRI
END
```

Could you make a neon sign that flashes your name????!!!

MSW LOGO TUTORIAL FIVE: VARIABLES

'Variable' is an important concept in Computer Science and programming.

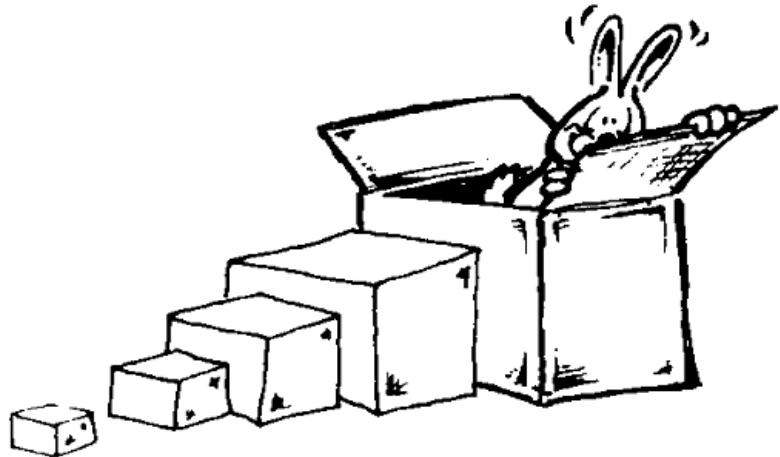
How many people are in your family? Or in your class? Are they all the same?
What can be different about them?

Let's talk about variables in LOGO. What would this procedure look like?

```

to boxes
repeat 4 [fd 100 rt 90]
rt 90 pu fd 120 pd lt 90
repeat 4[fd 100 rt 90]
end

```



So, what if you wanted to draw 20 boxes, each one bigger than the last? This is where we use variables.

When you start a variable, you have to provide an INPUT.

For example, we will start the procedure **boxes** and use the input **size**.

We write it like this “ **to boxes :size**. Note the use of the **:** before the variable.

Try to understand this procedure.....

```

to boxes :size
repeat 4 [fd :size rt 90]
rt 90 pu fd :size + 20 pd lt 90
repeat 4[fd :size rt 90]
end

```

Now, how do you run the procedure in the command box? Can you do it by typing **boxes** only?

Let's add a variable to the triangle procedure.

```

to tri :n
repeat 3 [fd :n rt 120]
end

```

Look at this example, made by a **7 year old student**. Try it!

```

to square :n
repeat 4 [fd :n rt
90]
end

```

```

to squares
square 60
square 80
square 100
square 120
end

```

```

to tables
squares
lt 90
squares
end

```

```

to mirror
tables
lt 90
tables
end

```

```

to mirrors
mirror

```

lt 45
mirror
end

Now try your own design using variables.....(you can also add color from the last tutorial).