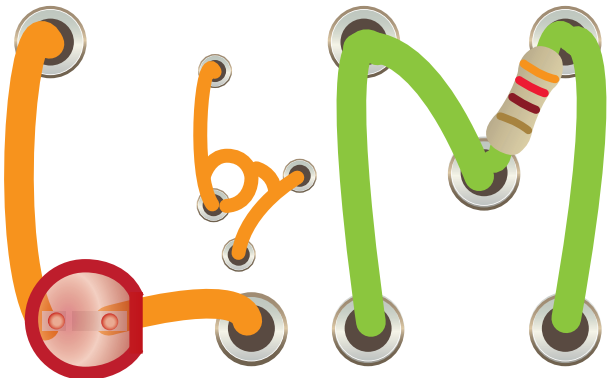


Learning by Making

UNIT 1

TurtleLogo

Starting TurtleLogo Project

Starting a new TurtleLogo project

Answer each question as shown below. If you do not see a black full screen window with questions, (1) click on the **black icon** at the bottom of your screen, (2) type **start** and (3) hit the **enter** key.

Terminal

```
Would you like to load an existing experiment? (y/n)
> n

Would you like to create a new experiment (y/n)
> y

Please select an experiment by typing its complete name
(eg BasicBoard.tar)
> TurtleLogo.tar

Provide a name for the new experiment folder
(eg. MyBasicBoard)
> TurtleLogo-YourName
```

This will open three windows:

- The command center (white),
- The drawing field (blue), and
- A text editor (white behind the other two windows)

Starting a previously created TurtleLogo project

Answer each question as shown below. If you do not see a black full screen window with questions, (1) click on the **black icon** at the bottom of your screen, (2) type **start** and (3) hit the **enter** key.

Terminal

```
Would you like to load an existing
experiment? (y/n)
> y

Which experiment would you like to
load?
> TurtleLogo-YourName
```

This will open three windows:

- The command center (white),
- The drawing field (blue), and
- A text editor (white behind the other two windows)

1.1 Introducing TurtleLogo

Learning Objectives:

- Students will be able to identify the three different windows used by `TurtleLogo`
- Students will learn how to clean the drawing field window using the command `clean`.
- Students will learn to drive the turtle using the commands: `fd`, `bk`, `rt` and `lt`.
- Students will learn how to draw specific geometric shapes using the turtle.

Materials

For each group of 2 students

- Computer
- Turtle Logo guide
- Worksheet
- Challenge sheet
- Double Dare sheet
- Graph paper

Vocabulary:

- Command Center window
- Drawing Field window
- Degree
- Polygon
- Text Editor window

Tasks you need to perform

Answer questions in your Worksheet, Challenge & Double Dare sheets

Getting Started

Imagine a turtle walking around on a sheet of paper with a pen in its mouth. Now imagine you could control that turtle's mind and drive it around. `TurtleLogo` is a computer program that does just that.

Type a command that the turtle *understands* and it will draw a picture on your screen.

Type in something the turtle *doesn't understand* and it will tell you what's wrong.

What words does the turtle understand? What can you draw with `TurtleLogo`?

Instructions

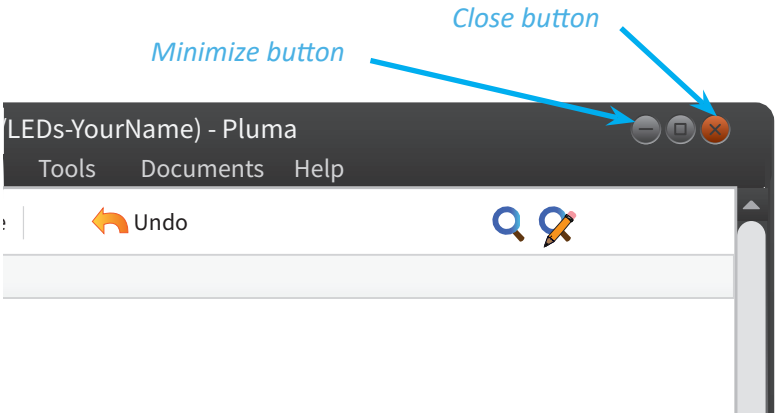
STEP 1. Start `TurtleLogo` Project

Turn on your computer. Create and launch your own personal `TurtleLogo` project following the instructions "Starting `TurtleLogo` Project" on p.3.

On your worksheet, write the words you see in the white **command center window** and describe what you see in the blue **drawing field window**, when `TurtleLogo` first starts.

STEP 2. Close `TurtleWords.txt` in the text editor window

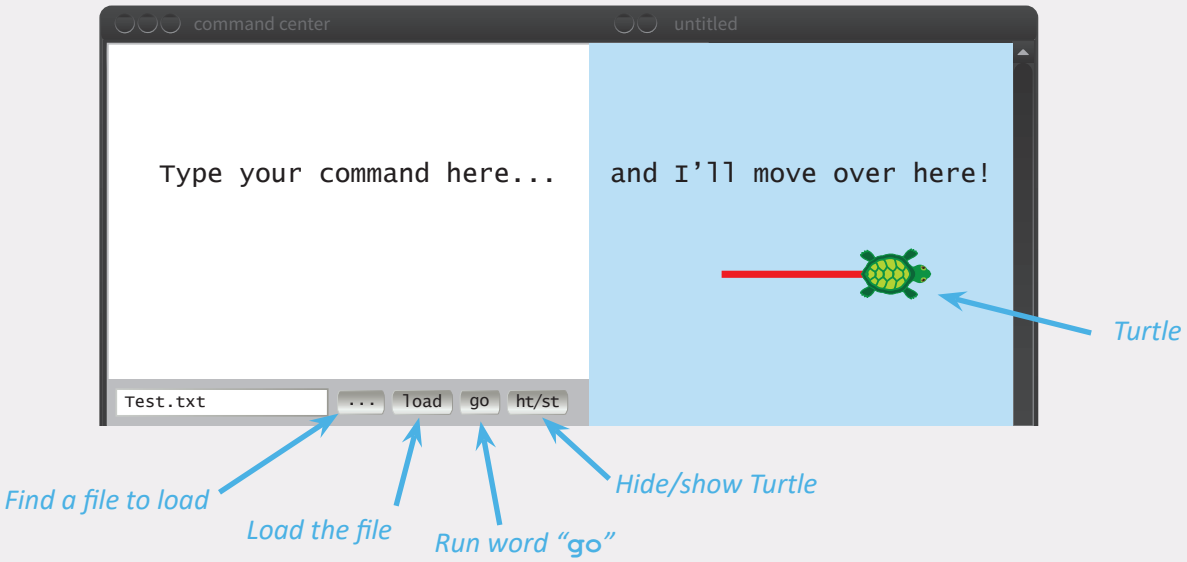
Behind the command center and the drawing field is a white window showing the file `TurtleWords.txt`. This is the **text editor window**.
Minimize the command center and drawing field windows by clicking the button in each window's title bar. Then close the remaining window with `TurtleWords.txt` by clicking the button.



On your worksheet, write down the names of the two remaining windows.

STEP 3. Reopen the minimized windows

Click on the penguin icon at the bottom of the screen.



On your worksheet, describe what happened.

STEP 4. Moving the turtle forward

Start over with a clean drawing field by typing `clean` and hitting the **enter** key. Move the turtle forward 200 steps by typing `fd 200` and hitting the **enter** key.

On your worksheet, write the direction that the turtle is facing after it is done moving.

> Embrace error messages. You cannot break the turtle. If something doesn't work, try typing a new line.
>
> If you get an error message, see if you can figure out what you did wrong by asking a classmate for help. If all else fails, ask your teacher.

STEP 5. Making the turtle move

In the command center window, type the following commands one at a time, and hit enter after each one. Then observe what happens.

Command Center

```
clean
fd 100
bk 100
```

Fill in the table on your worksheet. *[Note that the first two entries are completed, and you can use these as an example for the other command in the table.]*

STEP 6. Turn the turtle

Clean the drawing field again. In the command center window, type the following commands one at a time and hit enter after each one. Then observe what happens.

Command Center

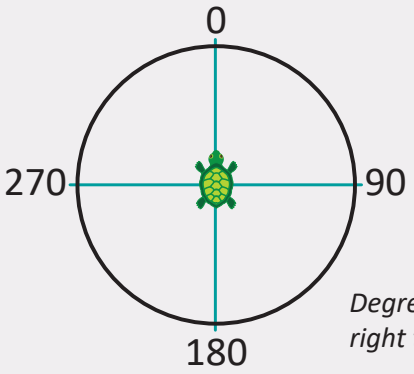
```
clean
fd 100
rt 90
```

On your worksheet, write the direction that the turtle is facing after it is done moving (up, down, left, right)?

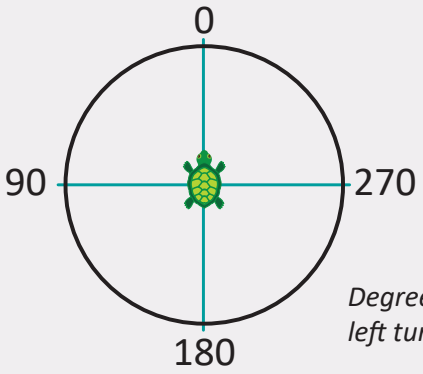
> You can copy, paste, and delete any lines in the command center at any time.
>
> If you mess up your drawing screen, type clean to start over

STEP 7. Discover other ways to move the turtle

The words **fd** and **bk** take input numbers that tell the turtle how many steps to take.
The words **lt** and **rt** tell the turtle how many **degrees** to turn, starting from the initial direction that the turtle is facing.
Clean the drawing field again. Then experiment with the words **fd, bk, lt** and **rt** to make the turtle move in different directions.



*Degrees for right turn (**rt**)*



*Degrees for left turn (**lt**)*

On your worksheet, fill out the table with at least five words that you used. Be sure to explain what the turtle did and also sketch a small picture of what the turtle drew.

> You cannot erase lines. You can only start over using **clean**. If you use the up arrow key on your keyboard, you will find commands you typed earlier.

STEP 8. Draw a line that changes directions

Clean the drawing field again. Now, experiment with all four commands: **fd, bk, lt** and **rt**. The goal is to draw a line that turns and goes in a different direction.

On your worksheet, write down the sequence of commands that you used to draw a line that changes direction. Also sketch the results.

STEP 9. Complete the challenges

Congratulations! You have now learned the basics of turtle driving. Continue on to complete the challenges for this lesson

Names:_____

STEP 1 Write the words you see in the white command center window and describe what you see in the blue drawing field window.

_____

_____

_____

_____

_____

STEP 2 Write down the names of the two remaining windows.

STEP 3 Describe what happened when you clicked on the penguin icon.

_____

_____

_____

_____

_____

STEP 4 What direction is the turtle facing after it is done moving (up, down, left or right)?

Name:_____

STEP 5 Fill in the table below.

If I use the command...	The turtle will...	Examples and notes
`clean`	Erase the drawing field and return to the center	This command is handy when I've driven the turtle somewhere off screen or when I make a mistake and want to start over.
`fd 100`	Move forward 100 steps	Change 100 to a different number to move the turtle forward by a different number of steps
`bk 100`		

STEP 6 What direction is the turtle facing after it is done moving (up, down, left or right)?

STEP 7 Fill out the table with at least five words that you used. Be sure to explain what the turtle did and also draw a small picture of what the turtle drew.

Command	What did the turtle did?	Your drawing

STEP 8 Write down the sequence of commands that you used to draw a line that changes direction. Also sketch the results.

Name:_____

For each Challenge, write the code you used to accomplish the task.

C-1

Draw a square with sides that are 75 steps long. Record the list of commands.

C-2

Draw a rectangle. Record the list of commands.

C-3

Complete the Geometry worksheet.

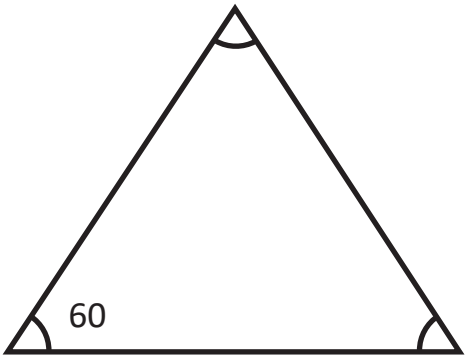
Names:_____

To draw using a turtle, it is helpful to review the geometry of simple **polygons** (shapes made of straight line segments)

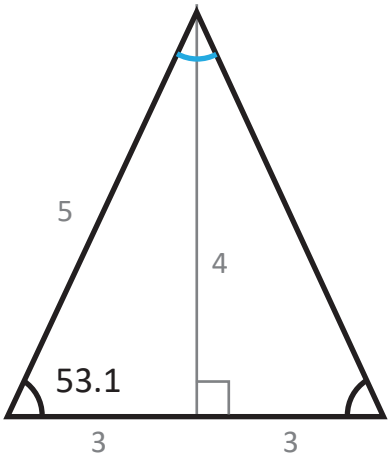
Triangles: The sum of the 3 internal angles adds to 180 degrees.

- Equilateral triangle: each side is the same length, and all 3 angles are equal.
- Isosceles triangle: 2 sides are the same length and one side is different. The angles opposite the 2 equal sides are equal.
- Right triangle: one internal angle is 90°

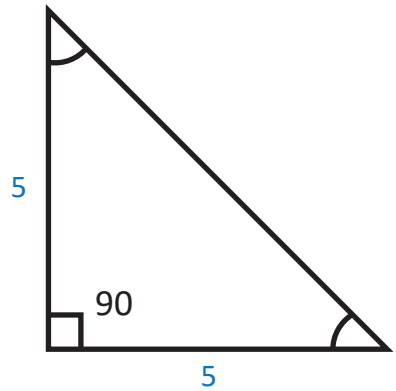
Fill in the missing angles on the drawings below:



Equilateral



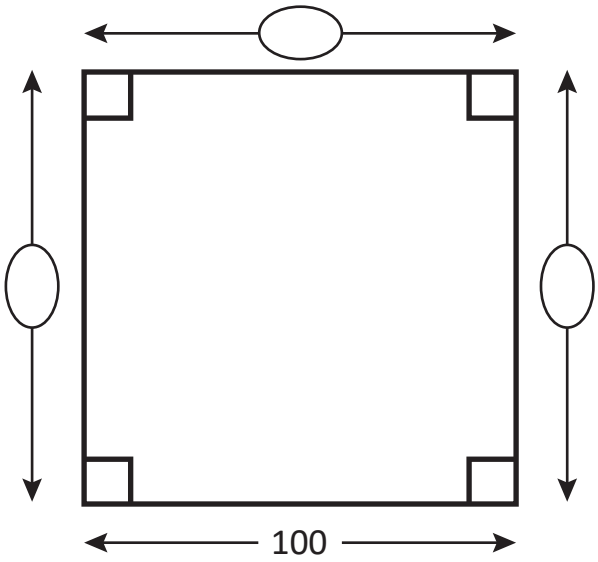
Isosceles



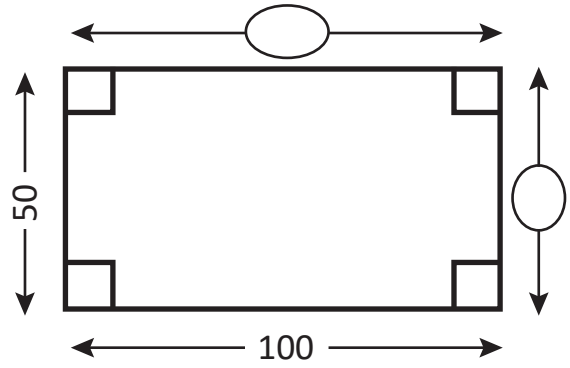
Right Isosceles

Names:_____

Rectangles: A rectangle has 4 sides, each at a 90° (right) angle. A square has 4 equal length sides.
Fill in the missing side lengths on the drawings below:



Square



Rectangle

Regular polygons: Regular polygons have at least 3 sides, and all the sides have the same length.
To draw a regular polygon, turn the turtle by an angle equal to $\frac{360}{\text{number of sides}}$
before drawing the next line segment.
Fill in the missing commands to make the turtle draw a square:

Code	
`fd`	`100`
`rt`	`90`
`fd`	
	`90`
	`100`
`rt`	

Name:_____

List the commands you used and draw a sketch of the results for each double dare. Feel free to use graph paper to plan your commands.

D-1

Draw any triangle.

D-2

Draw an equilateral triangle (all sides equal length)

D-3

Draw an isosceles triangle (two sides equal length)

D-4

Draw a right triangle (one 90˚ angle)

D-5

Draw a polygon with more than 4 sides

D-6

Draw any regular polygon with more than 4 sides

1.2 TurtleLogo Drawings

Learning Objectives:

- Students will learn how to fill the screen with a single color using **`fillscreen`**
- Students will learn how to change the colors drawn by the turtle using **`setcolor`**
- Students will learn how to change the width of the lines using **`setpensize`**
- Students will learn how to move the turtle to specified screen locations using **`setxy`**
- Students will learn how to start and stop the turtle from drawing lines using **`pendown`** and **`penup`**

Materials

For each group of 2 students

- Computer
- Turtle Logo guide
- Worksheet
- Challenge sheet
- Double Dare sheet
- Graph paper

Vocabulary:

- Argument
- Hue
- Pixel
- Saturation

Tasks you need to perform

Answer questions in your Worksheet, Challenge & Double Dare sheets

Getting Started

In *Lesson 1.1*, you learned how to move the turtle around the screen in straight lines, each of which was colored red. What if you wanted to change the line color or draw a line of different thickness? What if you wanted to move the turtle to a different location without drawing a connecting line?

It turns out the turtle knows how to do these things and much more! In this lesson, you will be introduced to other turtle commands that will allow you to draw shapes that are much more colorful.

H

You do not need to remake your personal **`TurtleLogo`** project from scratch as you did previously

Instructions

STEP 1. Reload your TurtleLogo project

Reload your **`TurtleLogo`** project from *Lesson 1.1* using the instructions on the bottom of page 3 "Starting a previously created **`TurtleLogo`** project". As in *Lesson 1.1*, close the text editor window (white window behind the command center window) by clicking the X in the red circle in the upper right hand corner.

STEP 2. Change the background color

Enter the following commands into the command center window:

Command Center

```
fillscreen 70 0
fillscreen 70 50
fillscreen 70 99
fillscreen 30 0
fillscreen 30 50
fillscreen 30 99
```

Colors available in `TurtleLogo`

Color-Code	Color
0	Red
10	Orange
20	Yellow
30	Lime Green
40	Blue Green
50	Sky Blue
60	Light Blue
70	Navy Blue
80	Purple
90	Pink
100	red

The **`fillscreen`** command takes two **arguments** (input numbers): the first input number specifies the color (or **hue**) and the second number specifies the shade (or **saturation**) for that hue.

Here is a table of colors that the turtle knows. You can use any number between 0 and 99 to specify the color and the shade. Shades vary from 0 (darkest) to 99 (lightest).

On your worksheet, fill out the table to describe what happened to the drawing field window.

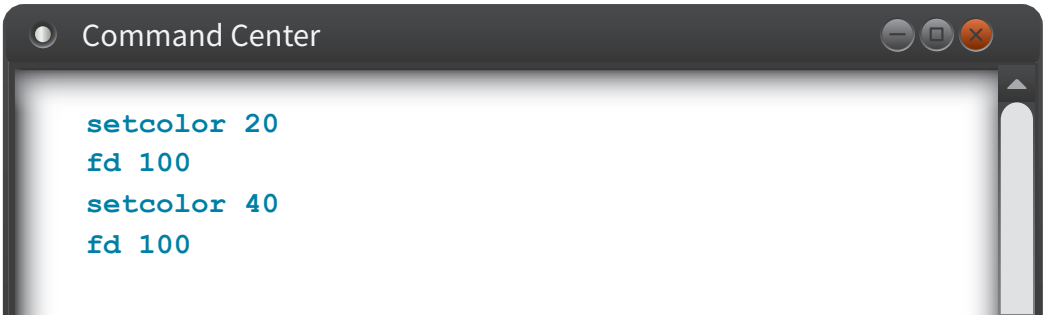
STEP 3. Experimenting with colors

Use the **`fillscreen`** command to experiment with at least five other color and shade values.

On your worksheet, write down each command you tried and describe what you observed.

STEP 4. Drawing lines of different colors

You can use the same color codes to draw lines of different colors using the **`setcolor`** command. Clean the screen first, then type in the following commands:

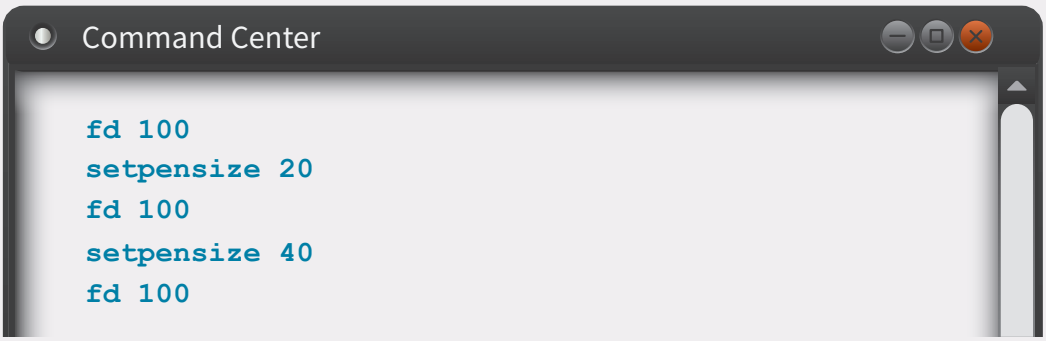


On your worksheet, fill out the table to describe what happened to the drawing field window.

STEP 5. Drawing lines of different widths

You can change the width of the pen that the turtle uses with the command **`setpensize`**. This command takes one argument, which is the width of the pen in **pixels**. Note that the default **`pensize`** is 5 pixels.

To try out this word, clean the screen, then type in the following commands:

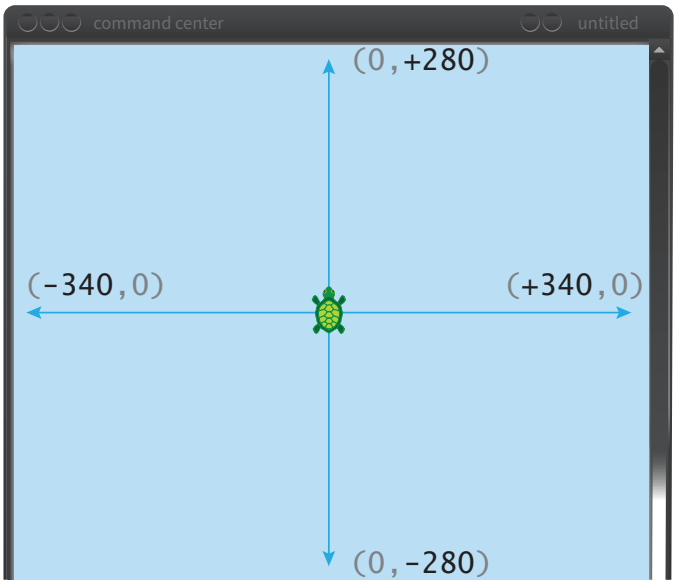


On your worksheet, fill out the table to describe what happened to the drawing field window.

STEP 6. Moving around the screen

In STEP 5 you learned how to change the width of the pen, by specifying the pen width in pixels. Each pixel corresponds to an individual position on the drawing field. The default position of the turtle is defined as the origin (0,0) in x-y coordinates. The turtle drawing field has pixels numbered from -340 to +340 in the x-direction and from -280 to 280 in the y-direction.

You can use the command **setxy** to tell the Turtle to move to a different position in the drawing field. However, the turtle will remember the state of the pen when it moves. If the pen was down, the turtle will draw a line connecting its old location to the new location. To avoid this problem, use the command **penup** before moving the turtle.



Clean the screen, and then type in the following sequence of commands:

Command Center

```
penup
setxy 200 70
pendown
fd 100
penup
setxy -200 70
pendown
fd 100
rt 90
fd 400
```

If you mess up your drawing field, type `clean` to start over and try again.

On your worksheet, sketch the shape that has resulted from this command sequence. Then figure out how to complete the missing side of the rectangle, using the **setxy** command and any other commands that may be needed. Write these commands on your worksheet.

STEP 7. Complete the challenges

Now that you have learned more turtle commands, continue on to the challenges for this lesson.

Name:_____

STEP 2 Fill out the table to describe what happened to the drawing field window.

Code	What happens?
`fillscreen 70 0`	
`fillscreen 70 50`	
`fillscreen 70 99`	
`fillscreen 30 0`	
`fillscreen 30 50`	
`fillscreen 30 99`	

STEP 3 Write down each command you tried and describe what you observed.

Code	What happens?

STEP 4 Fill out the table to describe what happened to the drawing field window.

Code	What happens?
`setcolor 20`	
`fd 100`	
`setcolor 40`	
`fd 100`	

Name:_____

STEP 5 Fill out the table to describe what happened to the drawing field window.

Code	What happens?
`fd 100`	
`setpensize 20`	
`fd 100`	
`setpensize 40`	
`fd 100`	

STEP 6 Sketch the shape that has resulted from this command sequence. Then figure out how to complete the missing side of the rectangle, using the `setxy` command and any other commands that may be needed. Write these commands below.

Code	What happens?

Name:_____

Write the commands you used and sketch the results in the space provided for each challenge.

C-1

Draw a square that uses a different color on each side.

C-2

Draw a rectangle that uses a different pen size on each side.

C-3

Change the color of the screen and draw a square that has one corner located at (x,y) = (100, 100). Do not draw any other lines on the screen except for the square.

Name:_____

Write the commands you used and sketch the results in the space provided for each double dare.

D-1

Use `penup`, `pendown` and `setxy` to create squares in each of the four quadrants of the screen. Make each square a different size.

D-2

Use `penup`, `pendown` and `setxy` to create two squares, each having a different pensize.

D-3

Change the color of the screen and draw 8 lines of different colors. Each line should start at (0,0) and go outwards for 100 steps. The 8 lines should be evenly spaced around the origin.

1.3 TurtleLogo Words

Learning Objectives:

- Students will learn how to create loops using the **repeat** command
- Students will learn how to use the text editor to modify commands and add new ones
- Students will learn how to control the speed of the turtle using the **wait** command

Materials

For each group of 2 students

- Computer
- Turtle Logo guide
- Worksheet
- Challenge sheet
- Double Dare sheet
- Graph paper

Vocabulary:

- Procedure
- Debug

Tasks you need to perform

Answer questions in your Worksheet, Challenge & Double Dare sheets

Getting Started

Wouldn't it be nice if you could shorten a long list of **TurtleLogo** commands for a complex drawing?
TurtleLogo can do that!

In this lesson, instead of typing repetitive words, you will learn how to create loops and learn how to save and load **TurtleLogo** programs.

Instructions

STEP 1. Reload your TurtleLogo project

Use the instructions at the bottom of page 3.

> You do not need to remake your personal TurtleLogo project from scratch as you did previously.

STEP 2. Code Comparison

Consider the following three **TurtleLogo procedures**. Try each option, cleaning the drawing window in between your experiments.

Option A

```
fd 100
rt 90
fd 100
rt 90
fd 100
rt 90
fd 100
rt 90
```

Option B

```
fd 100 rt 90
fd 100 rt 90
fd 100 rt 90
fd 100 rt 90
```

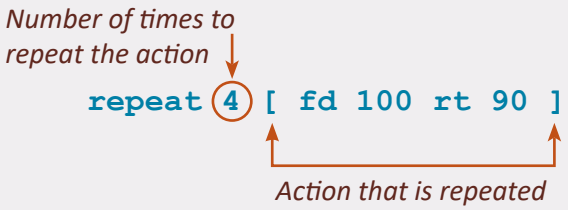
Option C

```
repeat 4 [ fd 100 rt 90 ]
```

On your worksheet, compare the code for each option. Were the drawings the same or did they differ? Explain your results by discussing the code.

STEP 3. Understanding the `repeat` command

In STEP 2, you used a new command: **`repeat`**. This command takes an argument – in this case, the number of times that some action should repeat. The action that is repeated is everything inside the square brackets [ ].



Use the **`repeat`** word to draw an equilateral triangle with sides that are 100 steps long.

On your worksheet, write the steps you used to draw a triangle with sides that are 100 steps long.

The spaces are useful to read the code. The square brackets are necessary for the **`repeat`** word to execute correctly.

STEP 4. Finding the `TurtleWords.txt` file

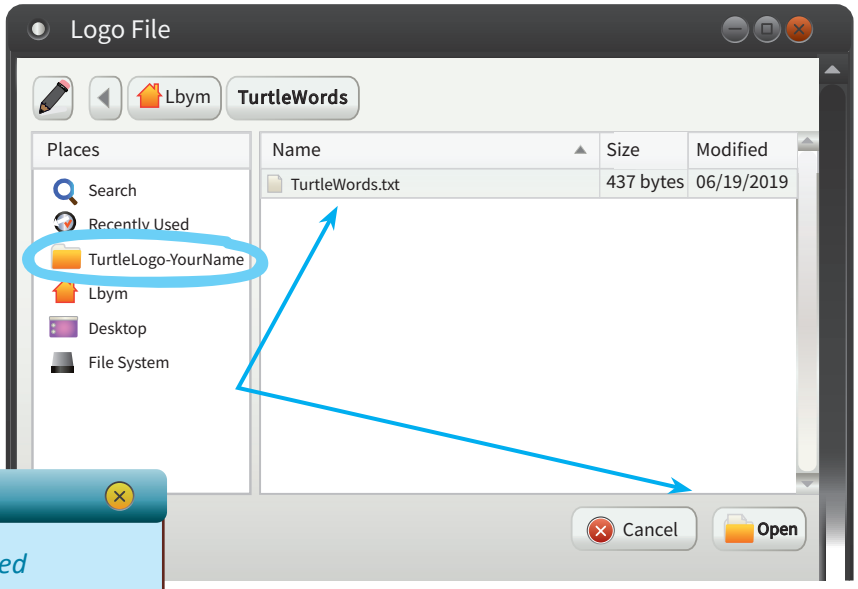
Near the bottom of the command center, you will see an icon with three dots. Clicking on this icon will bring up a new window that will show you the names of files that you can open.

a. Click on "icon with three dots" `...`

b. Click on the folder with your project name

c. Click on **`TurtleWords.txt`** as the file to load

d. Click Open



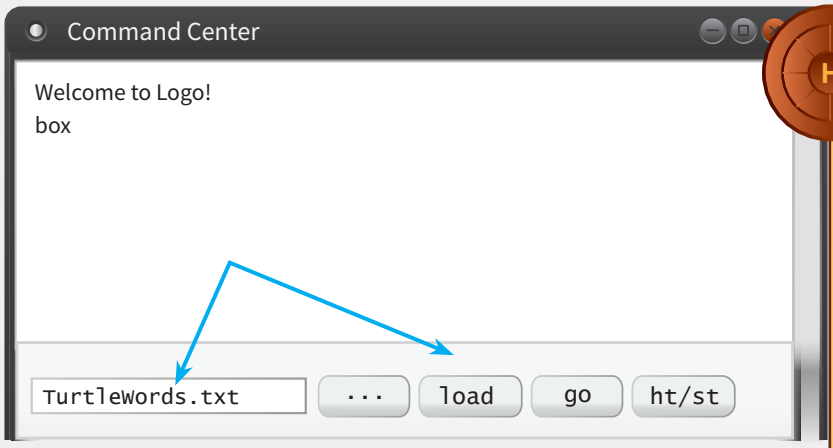
Do NOT click on *Recently Used*

Do NOT double click on the *file name*

On your worksheet, write down the name of the other file in your project folder (that is not **`TurtleWords.txt`**).

STEP 5. Loading the `TurtleWords.txt` file

TurtleLogo should now open the file. You will see its name in the space to the left of the icon with the three dots. Now click the load button to load the code in the **TurtleWords.txt** file. The file should also be open in the Pluma editor. Once the file is loaded, all the words in the file can be used to drive the turtle.



Type the command **box** in the command center.

On your worksheet, describe what happens in the drawing field after typing the command **box**.

> **H**
>
> Embrace error messages. You cannot break the turtle. If something doesn't work, try typing a new line.
>
> If you get an error message that says I **don't know how to** _____, check your typing. You may have misspelled the command, or you may not have loaded the file.
>
> If you get an error message, see if you can figure out what you did wrong by asking a classmate for help. If all else fails, ask your teacher.

STEP 6. Looking at the words in the `TurtleWords.txt` file

If it isn't already open, use the instructions in STEP 4 or the hint below to open the **TurtleWords.txt** file in the Pluma text editor. Find the place in the file where the word **box** is defined. It should look like this:

```
to box ; sample line comment
  repeat 4 [ fd 100 rt 90 ]
end
```

A word must be sandwiched between **to** and **end**

Give your word a useful descriptive name

Indent your commands to make the code human readable

On your worksheet answer:

1. On what line in the text editor do you see the code for this command?

2. After you've cleaned your drawing field and typed the command **box** what shape and color appears in the drawing field?

> **H**
>
> Comments within code are used to explain what will happen.
>
> Comments start with a semi-colon (;). Anything after the ; is ignored.

Do NOT delete any code in **TurtleWords.txt**.
All changes to this file should be additions only.

The text editor window may be hiding behind the command center window. If so, drag command center down and then click in the Pluma window to bring it to the front.

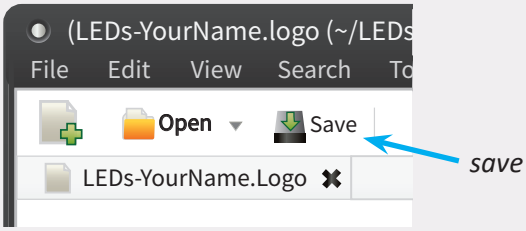
Another way to open the **TurtleWords.txt** file is to find the Filing Cabinet icon at the bottom of the screen and click on it. Find the folder named **TurtleLogo-YourName** and double click on it. Find the icon labeled **TurtleWords.txt** and double click on it. The file should now be open the Pluma text editor window.

STEP 7. Create a new word in TurtleWords.txt

Add the following piece of code to **TurtleWords.txt**. The word **wait** tells the turtle how many tenths of a second it should pause. The command, **wait 10**, means pause for 1 second (10 tenths of a second).

```
to turtle-wait
  repeat 10 [ fd 20 wait 10 ]
end
```

After you have typed in the code, save your file by clicking the save icon in the editor window.



Then minimize the editor window, click the load button in the command center window and run the new word by typing **turtle-wait**. If you see an error message, **debug** your code by looking for a typo in **TurtleWords.txt**.

If you mess up your drawing screen, type **clean** to start over.

You can copy, paste, and delete any lines in the command center at any time.

On your worksheet answer: how did the turtle move in the drawing field after you ran the **turtle-wait** command?

STEP 8. Add a comment to your code

Add at least one comment to the code for **turtle-wait** that describes what happens when you run it.

On your worksheet, write down the comment that you added to **turtle-wait**.

STEP 9. Complete the challenges

Continue on to the challenges, where you will practice creating and debugging words that are typed into your **TurtleWords.txt** file.

Name:_____

STEP 2 Compare the code for each option. Were the drawings the same or did they differ? Explain your results by discussing the code.

STEP 3 Write the steps you used to draw a triangle with sides that are 100 steps long.

STEP 4 Write down the name of the other file in your project folder (that is not **`TurtleWords.txt`**).

Name:_____

STEP 5 Describe what happens in the drawing field after typing the command **box** in the command center.

_____

_____

_____

_____

_____

STEP 6 1. On what line in the text editor do you see the code for this command?
2. After you've cleaned your drawing field and typed the command **box** what shape and color appears in the drawing field?

STEP 7 How did the turtle move in the drawing field after you ran the **turtle-wait** command?

STEP 8 Write down the comment that you added to **turtle-wait**.

1.3 Challenges

Name:_____

Write the commands you used and sketch the results in the space provided for each challenge.

C-1

Draw a square with sides that are 200 steps long using the `repeat` command.

C-2

Draw a triangle using the `repeat` command.

C-3

Draw any geometric shape using `repeat` and `wait`.

Name:_____

For each Challenge, write the code you used to accomplish the task.

C-4

What other words can you find in the `TurtleWords.txt` file? Remember, word definitions start with "**to**" and end with "**end**". Try running the other words you find and describe what happens in the drawing field for each one.

C-5

Edit the `TurtleWords.txt` file to change the size of the square drawn by `box` to have sides with length 200.

1. Make sure the file is open in the text editor window
2. Change the box word so that the sides are now 200 steps
3. Click **save** at the top of the text editor window
4. From the command center, click the **load** icon to reload your new code
5. Type `box` in the command center to try your new word
6. Write down the new code for the word `box`

Name:_____

Write the commands you used and sketch the results in the space provided for each double dare.

D-1

Draw multiple squares using the repeat command.

D-2

Create a drawing using the repeat command and at least two geometric shapes.

D-3

Draw a 4-, 5-, or 6-pointed star using the repeat command.

1.4 Words Within Words

Learning Objectives:

- Students will learn how to use the **go** icon in the command center window
- Students will learn how to make new commands out of simpler ones
- Students will gain familiarity with the text editor and will use it to edit the `TurtleWords.txt` file to modify old words and to create new words

Materials

For each group of 2 students

- Computer
- Turtle Logo guide
- Worksheet
- Challenge sheet
- Double Dare sheet

Vocabulary:

- Default

Tasks you need to perform

Answer questions in your Worksheet, Challenge & Double Dare sheets

Getting Started

Wouldn't it be nice if you could combine `TurtleLogo` commands for a complex drawing so that you could pull them up with ease? **`TurtleLogo` can do that!**

In this lesson, you will learn how to edit a file so that you can create more complicated commands based on previously defined words.

Instructions

STEP 1. Reload your TurtleLogo project

Reload your `TurtleLogo` project. Use the instructions at the bottom of page 3.

H: Embrace error messages. You cannot break the turtle. If something doesn't work, try typing a new line.

W: If you are trying to open an existing project, make sure to use the instructions on the **bottom** of page 3

STEP 2. Loading the `TurtleWords.txt` file

Near the bottom of the command center, you will see an icon with three dots. Clicking on this icon will bring up a new window that will show you the names of files that you can open.

a. Click on "icon with three dots" `...`

b. Click on the folder with your project name

c. Click on `TurtleWords.txt` as the file to load

d. Click Open

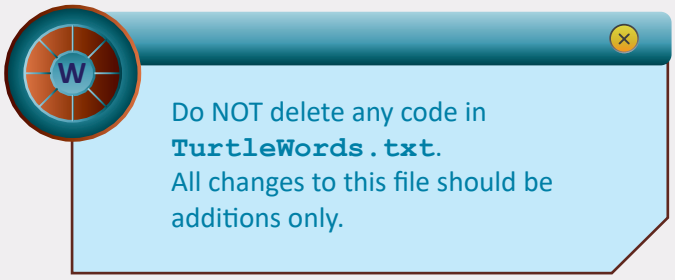
W: Do NOT click on *Recently Used*

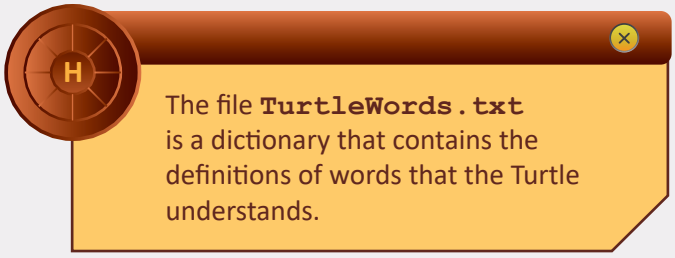
Do NOT double click on the *file name*

STEP 3. Running the word go

After you have reloaded **TurtleWords.txt**, click on the **go** button.

On your worksheet, write a description of what happened to the drawing field window when you clicked the **go** icon.





STEP 4. What does go do?

Read the **TurtleWords.txt** file to find the definition of the word **go**. This word has a special connection to the icon buttons at the bottom of the command center window. When you click the **go** icon, the computer will execute the words and instructions that are part of the definition of **go** in the **TurtleWords.txt** file.

On your worksheet, write down the **default** actions that take place when you click the go icon. Modify these actions by editing the **TurtleWords.txt** file. For example, you can choose to change either the pen color or the pen size, or to insert a new command to change the background color. Then save the file by clicking the save icon at the top of the editor window. Then click the **load** and **go** buttons in the command center window.
Did the turtle follow your new instructions? Write a description of what happened on your worksheet.

STEP 5. Code Comparison

Compare the definition of the two words **box** and **box-stack**, given below or in **TurtleWords.txt**. Once you create a new word for the Logo language and enter it in a dictionary file like **TurtleWords.txt**, it becomes part of the language and can be used inside more complex **procedures**. (A procedure is just a collection of command words to be executed.)

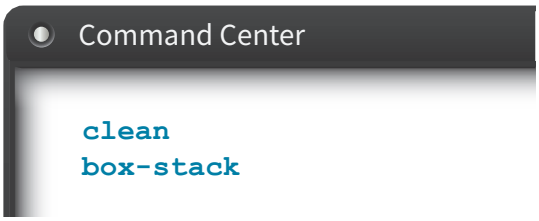
```
to box
  repeat 4 [ fd 100 rt 90 ]
end
```

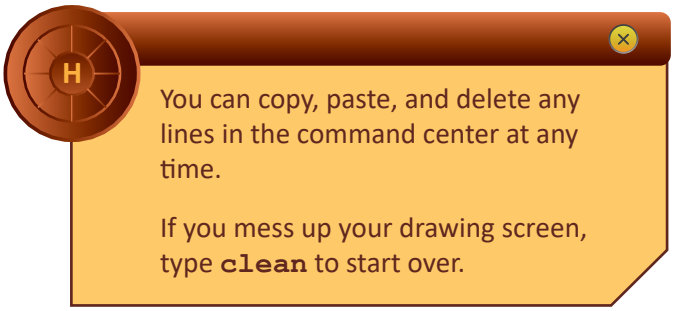
```
to box-stack
  repeat 2 [ box fd 100 ]
end
```

On your worksheet, sketch what you think the turtle will do for each word (**box** and **box-stack**).

STEP 6. Running the words

Now clean the screen and then run the **box-stack** word through the command center window.



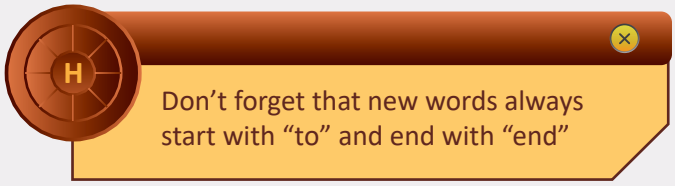


On your worksheet, indicate what **box-stack** actually did and compare it to your sketch in STEP 5.

STEP 7. Create your own word

Edit **TurtleWords.txt** to create your own word. Think of a name for your word, and make sure that your new word draws at least one square box. Save the **TurtleWords.txt** file, load it and run your new word in the command center window.

On your worksheet, write the code for your new word and sketch the results.



STEP 8. Complete the challenges

If you can easily use the text editor, please move on to the challenges in the next section. If not, spend some more time editing words in **TurtleWords.txt** and running them.

Name:_____

STEP 3 Write a description of what happened to the drawing field window when you clicked the go icon.

_____

_____

_____

_____

_____

STEP 4 Write down the **default** actions that take place when you click the **go** icon. Modify these actions by editing the `TurtleWords.txt` file. For example, you can choose to change either the pen color or the pen size, or to insert a new command to change the background color. Then save the file by clicking the save icon at the top of the editor window. Then click the **load** and **go** buttons in the command center window.
Did the turtle follow your new instructions? Write a description of what happened.

_____

_____

_____

_____

_____

_____

STEP 5 Sketch what you think the turtle will do for each word (`box` and `box-stack`).

Name:_____

STEP 6 Indicate what `box-stack` actually did and compare it to your sketch in STEP 5.

_____

_____

_____

_____

_____

_____

STEP 7 Write the code for your new word and sketch the results.

Name:_____

For each Challenge, write the code you used to accomplish the task.

C-1

Edit the `go` word in `TurtleWords.txt` so that clicking on the **go** icon makes the turtle draw two stacked boxes.
Hint: what existing word knows to draw 2 stacked boxes?

C-2

Edit the `go` word so that clicking on the **go** icon will run every word that is defined in `TurtleWords.txt`.

C-3

Create a new **TurtleLogo** word that uses at least two words that were previously defined in `TurtleWords.txt`. Write your new word below.

Name:_____

Write the commands you used and sketch the results in the space provided for each double dare.

D-1

Use at least five existing words from **`TurtleWords.txt`** to draw an interesting pattern.

D-2

Create a word that uses **`box`** to draw squares in all four corners of the drawing field.
Hint: don't forget to use **`penup`** *and* **`pendown`** *commands.*

D-3

Draw any object by creating a word that uses at least two other words and also uses the **`repeat`** command.

1.5 Variable Inputs to Words

Learning Objectives:

- Students will learn how to use variable inputs to words in `TurtleLogo`
- Students will learn how to run `TurtleLogo` commands that require inputs

Materials

For each group of 2 students

- Computer
- Turtle Logo guide
- Worksheet
- Challenge sheet
- Double Dare sheet
- Graph paper

Vocabulary:

- Input
- Variable

Tasks you need to perform

Answer questions in your Worksheet, Challenge & Double Dare sheets

Getting Started

What if you can't decide how big of a box to draw?

What if you want to draw boxes in different colors without editing, saving and reloading your code?

The size of the box and the color of the turtle's ink can each be specified with a number. This number can be used as **input** to a Turtle word.

Running the word with different input numbers will produce different drawings on the screen.

In this lesson, you will learn how to write `Turtle` words that accept **variable** input numbers.

Instructions

STEP 1. Reload your `TurtleLogo` project and your `TurtleWords.txt` file

Near the bottom of the command center, you will see an icon with three dots. Clicking on this icon will bring up a new window that will show you the names of files that you can open.

a. Click on "icon with three dots" ...

b. Click on the folder with your project name

c. Click on `TurtleWords.txt` as the file to load

d. Click Open

H

The text editor window may be hiding underneath the command center window. Move the command center window out of the way so that you can edit the `TurtleLogo.txt` file when needed.

STEP 2. Code Comparison

Compare the two words `box` and `draw-box` below. The new piece, `:n`, is called a **variable**. In the `Logo` computer language, variable names always start with a colon (:). Variables are used to hold input numbers. Add this new word, `draw-box`, to your `TurtleLogo.txt` then save the file and load it.

```
to box
  repeat 4 [ fd 100 rt 90 ]
end
```

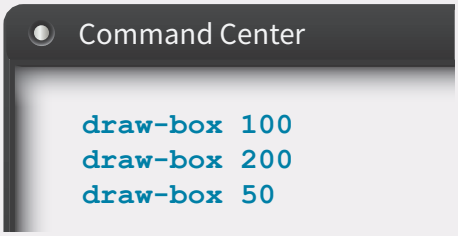
```
to draw-box :n
  repeat 4 [ fd :n rt 90 ]
end
```

On your worksheet, answer:

1. In `box`, what is the size of the box that the turtle draws?
2. Do you know the box size in `draw-box`? Explain.

STEP 3. Running `draw-box` with variable input

From the command center, type **`draw-box 100`** and hit enter. Type **`draw-box 200`** and hit enter. Type **`draw-box 50`** and hit enter.



If you get an error _____ needs more inputs, you have forgotten to provide the numerical value of your variable.

Don't forget that new words always start with "to" and end with "end"

On your worksheet, write a few sentences that explain the difference between **`box`** and **`draw-box`**. Your explanation must include a discussion of how variables work.

STEP 4. Modify words for input

`draw-box` is similar to **`box`** but it allows the user to change the box size when the word is run.

Find two other words in **`TurtleWords.txt`** that you can modify to add input variables. Copy each word, give it a new name and change it so that it takes a variable input. For example, you could decide to vary the size of one side of a box in the **`box`** command. Or you could decide to vary the number of boxes that are stacked up in **`box-stack`**.

On your worksheet, write the new code definitions that you wrote for each word and sketch the resulting drawings.

A variable can be more than one letter. A word can also accept multiple variables. The following variable names are valid -

`:size` **`:length`**
`:width` **`:apple`**

STEP 5. Running a word with more than one variable input

There is no limit to how many input variables you can define for a given word. To use more than one input variable, you need to name each variable with a different letter or description. For example, you could use **`:size`** instead of **`:n`** to represent the number of steps in one side of a box, while using **`:nbox`** to represent how many boxes to draw. In this case, **`:nbox`** would be used as the number which tells the repeat command how many times to draw another box.

Write code for a new word that will take two or more input variables. Type it into **`TurtleWords.txt`**, save the file, load it and then run the new word in the command center window.

On your worksheet, write the new code that you have created and explain the role of each variable in your code. Also sketch the result.

If you get an error message that says **`I don't know how to`** _____, check your typing. You may have misspelled the command.

You can't define new words with the **`to … end`** format within the command center. These must be in TurtleLogo files that you need to save, load, and run.

STEP 6. Complete the challenges

If you successfully creatd your own words with variable inputs, please move on to the challenges in the next section. If not, spend some more time creating your own variable-input words and running them.

1.5 Worksheet

Variable Inputs to Words

Name:_____

STEP 2 1. In `box`, what is the size of the box that the turtle draws?
2. Do you know the box size in `draw-box`? Explain.

STEP 3 Write a few sentences that explain the difference between `box` and `draw-box`. Your explanation must include a discussion of how variables work.

_____

_____

_____

_____

_____

_____

STEP 4 Write the new code definitions that you wrote for each word and sketch the resulting drawings.

STEP 5 Write the new code that you have created and explain the role of each variable in your code. Also sketch the result.

Name:_____

Write the commands you used and sketch the results when appropriate for each Challenge.

C-1

Create a `TurtleLogo` word **`draw-triangle`** that uses a variable to define the size of one side of an equilateral triangle. Run the word a few times with different inputs and describe the results.

C-2

Modify the **`draw-box`** word (that you previously typed into **`TurtleLogo.txt`**) so that you can change the color of the box by using a variable input number. Run the word a few times with different inputs and describe the results.

C-3

Modify either the **`draw-triangle`** or the **`draw-box`** word to use a second input variable. Run the word a few times with different inputs and describe the results.

Name:_____

Write the commands you used and sketch the results when appropriate for each Double Dare. Also, show your teacher the drawing field after completion.

D-1

Create a word that can change the size, color, and pen size of a square using input variables.

D-2

Create a word that uses a variable amount of time to wait between drawing each side of a box, and a second variable that changes the size of each side of the box.

Name:_____

Write the commands you used and sketch the results when appropriate for each Double Dare. Also, show your teacher the drawing field after completion.

D-3

Create a word that uses three variables as input to draw any regular geometric shape (such as a square or an equilateral triangle). The variables should describe the color, size and number of sides of the object. *Hint: the angle to turn the turtle before drawing the next side is equal to 360/(number of sides).*

1.6 Turtle Art

Learning Objectives:

- Students will learn how to use the **arc** and **setshade** commands
- Students will use a variety of **TurtleLogo** words to create a unique work of art that incorporates geometric shapes

Materials

For each group of 2 students

- Computer
- Turtle Logo guide
- Worksheet
- Challenge sheet
- Double Dare sheet
- Graph paper

Vocabulary:

- arc

Tasks you need to perform

Answer questions in your Worksheet, Challenge & Double Dare sheets

Getting Started

In this final exercise, you will use a variety of **TurtleLogo** words to create a geometric art project.

Instructions

STEP 1. Reload your TurtleLogo project and your TurtleWords.txt file

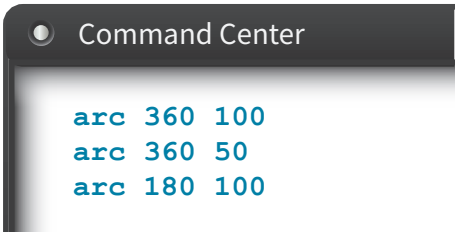
Use the instructions at the bottom of page 2 to reload your project, and then click on this icon to see a list of files that you can load. Choose **TurtleWords.txt** by clicking on it. Then click on the load icon at the bottom of the command center window. Make sure that the **TurtleWords.txt** file is open in a text editor window.

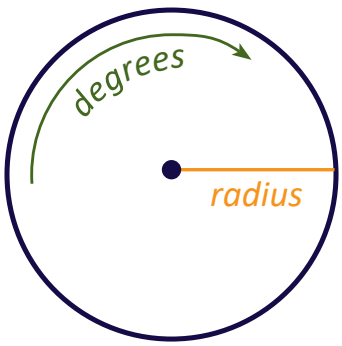
> The text editor window may be hiding underneath the command center window. Move the command center window out of the way so that you can edit the **TurtleLogo.txt** file when needed.

STEP 2. Using the arc word

arc is a word that is very useful for creating Turtle Art. The **arc** word takes two inputs: the number of degrees in the arc, and the number of steps that make up the radius of a circle.

Type the following into the command center to see how **arc** works. Clean the screen between each trial.





On your worksheet, fill in the table with a description of what happened when you typed in the words.
When the turtle draws a complete circle, on what side of the turtle is the circle (left or right?)

STEP 3. Using the `setshade` word

setshade is another word that can be useful in creating Turtle Art. It is similar to **setcolor**, but the shades that are available vary from 0 (black) to 100 (white) for each color. Type the following into the command center to see how **setshade** works. If needed, clean the screen between each trial.

On your worksheet, fill in the table with a description of what happened when you typed in the words.

Command Center

```
setcolor 50
fd 100
setshade 10
fd 100
setshade 90
fd 100
```

STEP 4. Practice Art

Use the **arc** word and other **TurtleLogo** words to draw three letters in a single color and pensize. For example: the letters can be your initials or those of a friend. Do not let the letters touch each other.

On your worksheet, write the new code that you have created, and show your output to your instructor.

Remember to use **penup**, **pendown** and **setxy**. Graph paper may help you plan out your drawings.

STEP 5. Complete the challenges

Please complete the art challenge. If needed, spend some more time practicing and reviewing the different word choices.

Name:_____

STEP 2 Fill in the table with a description of what happened when you typed in the words.
When the turtle draws a complete circle, on what side of the turtle is the circle (left or right?)

Code	What happens?
`arc 360 100`	
`arc 360 50`	
`arc 180 100`	

STEP 3 Fill in the table with a description of what happened when you typed in the words.

Code	What happens?
`setcolor 50`	
`fd 100`	
`setshade 10`	
`fd 100`	
`setshade 90`	
`fd 100`	

STEP 4 Write the new code that you have created, and show your output to your instructor.

1.6 Challenges

Name:_____

Code Requirements:

1. Your code must contain at least three unique words of your own creation. These can be anything that gives the turtle a series of directions to draw a shape, perform an action, or follow any kind of direction you choose.
2 You must use at least one repeat somewhere in your code.
3. You must include at least one geometric shape in your drawing.
4. You must change either the color of the pen, the width of the pen or the color of the background at least once.
5. You must have at least one word that requires a variable input.
6. Your entire project should display when you type its name into the command center or when you click the **go** icon

C-1

Create a **`TurtleLogo`** art project including each of the requirements listed above.

Present your **`TurtleLogo`** art project to the class.

Presentation Instructions:

1. Connect your Computer to the projector and demonstrate your final product to the class.
2. Display your code and point out the specific parts of code that fulfill the requirements listed above.
3. You will describe the parts of code and explain how they affect the outcome or actions of the turtle.
4. Answer questions from the class about your project.

1.6 Double Dare Challenges

Name:_____

Write the commands you used and sketch the results when appropriate for each Double Dare. Also, show your teacher the drawing field after completion. Graph paper may be helpful.

D-1

Write a word that draws your name or another word on the screen

D-2

Write a word that draws a dashed line where the number of dashes and the size of the dashes are both variables.

1.6 Double Dare *Continued*

Name:_____

Write the commands you used and sketch the results when appropriate for each Double Dare. Also, show your teacher the drawing field after completion. Graph paper may be helpful.

D-3

Write a word that draws a smiley face on the screen

D-4

Write a word that draws a spiral on the screen

D-5

Be creative! Make an amazing Turtle Art drawing.

Appendix A - Glossary

Use this glossary to write down the definitions of all the Vocabulary words for each lesson. Feel free to add additional words that you have learned.

A

Arc:

Argument:

B

C

Command Center window:

D

Debug:

Default:

Degree:

Drawing Field window:

E

F

G

H

Hue:

I

Input:

J

K

L

M

N

O

P **Pixel:**

Procedure:

Polygon:

R

S **Saturation:**

T **Text Editor window:**

V **Variable:**

W

X

Y

Z