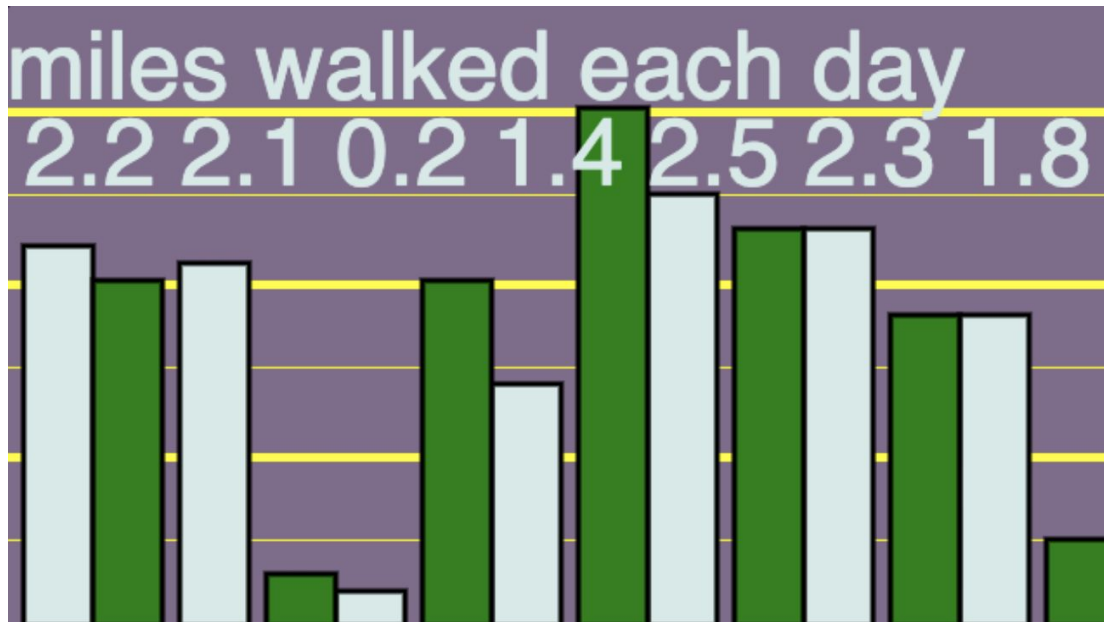




Data Visualization

What girls do:

Complete a data tracking app using JavaScript on Vidcode!

What girls learn:

Data visualization can help people to change their habits. Girls will learn how to code arrays to store and display data before they complete a prototype for a personal data app.

Steps:

Cadettes complete the “Personal Data” activity on Vidcode.

Share the Things to Know to get Cadettes started on the plugged activity.

Cadettes can login into Vidcode by logging into their account on Vidcode. Then, girls can navigate to 'start coding', then click on 'cadette', then 'cadette app'. Once the Vidcode platform loads, Cadettes can begin their projects. The projects are designed for girls to work in pairs, but they also work if girls are working alone.

If Cadettes want to take a break and complete their project later, they can save their project at any time by clicking the 'save' button at the bottom of the code editor.

When Cadettes have finished their project, or completed as much as they can or want to, they can click 'publish,' title their work, and then share their project with their friends or family.

Background

App— stands for application, used by computer programmers to refer to self-contained software that a user interacts with on different devices. Apps can be used for different purposes like organizing information (such as research notes, passwords, or to-do lists), providing a service (such as giving map directions, calculating compound interest, searching for flight information, shopping online, tracking the number of steps you walk each day, or checking a bank balance), or providing entertainment (such as playing a game or a video).

Array— a type of list that has a particular order, like an index or a shopping list. Each part of an array is called an element. Each element is separated by a comma. At each end of an array are square brackets '[']'.

Here is an example array made of words:

```
var ourConcerns = ["the environment", "access to education", "mutual respect"];
```

Here is an example array that lists how many Girl Scout cookies people ate in one sitting:

```
var cookiesEaten = [2, 3, 6, 2, 2, 0, 1, 4] // one person ate 2, one ate 3, etc.
```

The "var" at the beginning stands for variable, which is part of JavaScript syntax.

Bar chart— a graphical way to display data using bars of different heights. The type of bar chart Cadettes will be creating is a histogram where each bar groups numbers into ranges and taller bars show that more data falls in that range.

Data– any set of facts or statistics collected and analyzed or used for reference. Data can be in many forms and include information like steps taken, photos shared, or emails and messages sent. It can also include information collected from your cell phone or other device such as location history, internet browsing history, or login names and passwords.

Data visualization– a visual way data scientists, computer programmers, designers, and others communicate information clearly and efficiently. Data visualization uses statistical graphics, plots, information graphics, and other tools. Effective visualization helps users analyze and think about data. It makes complex data more accessible, understandable, and usable.

Health/Habit tracking app– self-contained software that helps people develop healthy habits. Users can set fitness goals, track nutrition, exercise, sleep, or even moods. They can also share progress on social media to encourage healthy behavior changes.

Prototype– a first version of a product which is built to be tested so that changes can be made before production.

Code Challenge:

- In the App Development badge, you write arrays and create data visualizations with personal data. You then used data to create a prototype for a habit-tracking app.
- Remember, by tracking your data, you can use it to form new habits.
- Now, pair up with another Cadette to finish an incomplete app for tracking personal data.
- In the activity, you're going to store your data with arrays, and then display it with graphics. This is going to track your sleep patterns. It's really important to get good sleep!
- If you want to use other ideas for the data, you can do that, too!

Sample Solution:

<https://www.vidcode.com/share/C36rF7Phi1>

Sharing:

Troop:

Publish all the projects to the Vidcode gallery. Study the arrays that they've created.

Have Girl Scouts explain how changes in these values affect the way their graphs looks.

Partners:

Share on individual computers with neighbors. Give feedback.

Step by Step Guide:

This section walks the volunteer through one possible way to get to a working program. Use this as a guide for helping you understand how the code develops. Try not to give away the solution if at all possible – Girl Scouts learn best from figuring out the problem themselves!

Step: 1.1-1.4

```
// this section stores the sleep data
var sleep1 = [13, 8.6, 7.5, 8, 6, 7, 12];
var sleep2 = [];
var unit = movie.height / Math.max(...sleep1, ...sleep2) / -1.5;

// this displays the webcam
movie = webcam();
tint("red", 50);

// this section displays the histogram
var mon1 = rect(0, movie.height, 30, unit * sleep1[0], "green");
var tues1 = rect(30, movie.height, 30, unit * sleep1[1], "green");
var wed1 = rect(60, movie.height, 30, unit * sleep1[2], "green");
var thurs1 = rect(90, movie.height, 30, unit * sleep1[3], "green");
var fri1 = rect(120, movie.height, 30, unit * sleep1[4], "green");
var sat1 = rect(150, movie.height, 30, unit * sleep1[5], "green");
var sun1 = rect(180, movie.height, 30, unit * sleep1[6], "green");
```

Notes:

The code above is given and the first part of the activity involves customizing it as a way of learning what it does. This code may be intimidating right now, but it will become much clearer.

Step: 1.5-1.7

```
var sleep1 = [5, 7.5, 9, 8.5, 5, 7, 9];
```

Notes:

The second line is the only line that changes.

Variables: variables are like names for things and a place to store data. The keyword `var` says 'create a variable', `sleep1` is the name of the data, and after the `=` is the data itself.

Arrays: Arrays are lists, and store data that occurs in order – such as days of the week.

Errors: The most common syntactical errors are in the array. The form has to be `[` then a list (of numbers in this case), separated by commas, and then the `]`. Also, sometimes the `,` and the `.` get confused.

Step: 1.8

Correct Answer: 3

Wrong Answers:

The first example has no commas, and the second example is closed with a brace `}`.

Step: 1.9-1.11

```
// this displays the webcam  
movie = webcam();  
tint("#4FF3F3", 30);
```

Notes:

Functions: `tint()` is a function. Functions perform actions, like verbs. `tint()` changes the color of the video. But, to what color?

Arguments: arguments provide a general function with specificity. `tint("red", 50)` tints the video a red color with a 50% density.

Hex Value: when you click on a color, you get the color picker and can simply choose the color you want. The text that is created – a series of 6 numbers, a to f, and letters, 0 to 9 – is called a hex value and it describes the exact color. It should be in quotes.

`rect()`: is a new function that creates rectangles. It will be explored more thoroughly in the next section.

Step: 2.1-2.2

```
var sleep2 = [8, 6, 5, 5, 8, 7.4, 9];
```

Notes:

Nothing will change just yet. There is new data, but it isn't displayed. Just make sure that the data is correct.

Step: 2.3

Correct Answer: 3

Step: 2.4-2.7

```
// this section displays the histogram
var mon1 = rect(0, movie.height, 30, unit * sleep1[0], "#0CDF0C");
var tues1 = rect(30, movie.height, 30, unit * sleep1[1], "#4ECF4E");
var wed1 = rect(60, movie.height, 30, unit * sleep1[2], "#35E035");
var thurs1 = rect(90, movie.height, 30, unit * sleep1[3], "#3AA83A");
var fri1 = rect(120, movie.height, 30, unit * sleep1[4], "#1F961F");
var sat1 = rect(150, movie.height, 30, unit * sleep1[5], "#34C234");
var sun1 = rect(180, movie.height, 30, unit * sleep1[6], "#19DF19");
var mon2 = rect(350, movie.height, 30, unit * sleep2[0], "#CCF04B");
```

Notes:

Concept: For these steps, the variable name gets changed - this means that there is a new rectangle. At first it is in the same location as the first one - so it won't be visible. Then the first argument gets changed. That argument specifies the x position.

Syntax: take care with commas, and that the color is still in quotes after it is changed.

Step: 2.8

```
var mon2 = rect(350, movie.height, 30, unit * sleep2[0], "#CCF04B");
```

Notes: sleep1 changed to sleep2. Make sure that the [0] is still intact. That's important! It says to refer to use the first number of the sleep2 array. Each number in an array is called an element.

Step: 2.9

```
var tue2 = rect(380, movie.height, 30, unit * sleep2[0], "#CCF04B");
```

Notes:

At this point, the new line will be the same height as the last one. We'll change that soon.

Step: 2.10

```
var tue2 = rect(380, movie.height, 30, unit * sleep2[1], "#CCF04B");
```

Notes:

Index: an index is a position in an array. Arrays have their elements in order: first, second third, etc. You refer to the first element with this syntax: sleep2[0]. Notice that the 'first' element is actually 0! The second is 1, etc. The only code change in this step is the index of sleep2, [1].

Step: 2.11

Correct Answer: 3

Wrong Answers:

The first answer uses the parentheses – these are used in functions, not arrays. The second answer is the wrong index. `exercise[5]` refers to 1.

Step: 2.12

```
// this section displays the histogram
var mon1 = rect(0, movie.height, 30, unit * sleep1[0], "#0CDF0C");
var tues1 = rect(30, movie.height, 30, unit * sleep1[1], "#4ECF4E");
var wed1 = rect(60, movie.height, 30, unit * sleep1[2], "#35E035");
var thurs1 = rect(90, movie.height, 30, unit * sleep1[3], "#3AA83A");
var fri1 = rect(120, movie.height, 30, unit * sleep1[4], "#1F961F");
var sat1 = rect(150, movie.height, 30, unit * sleep1[5], "#34C234");
var sun1 = rect(180, movie.height, 30, unit * sleep1[6], "#19DF19");

var mon2 = rect(350, movie.height, 30, unit * sleep2[0], "#F04BEC");
var tues2 = rect(380, movie.height, 30, unit * sleep2[1], "#EB6FE8");
var wed2 = rect(410, movie.height, 30, unit * sleep2[2], "#9B2098");
var thurs2 = rect(440, movie.height, 30, unit * sleep2[3], "#EE6FEB");
var fri2 = rect(470, movie.height, 30, unit * sleep2[4], "#F04BEC");
var sat2 = rect(500, movie.height, 30, unit * sleep2[5], "#D342CF");
var sun2 = rect(530, movie.height, 30, unit * sleep2[6], "#C311BF");
```

Notes:

Make sure that each step is done carefully, or else errors can creep in. Be careful with commas, and variable names, too. Although there is a lot of code in this step, there are no new concepts.

Step: 3.1-3.2

```
var my_text1 = text("Our sleep data!", 100, 10);
```

Notes:

Make sure that the cursor is at the bottom of your code before the text block is used. Otherwise, the text can be placed somewhere harder to find.

Make sure that the message in the text() is in quotes like this: text("this is the message")

Adding the arguments for the position is a matter of adding two numbers after the message. Make sure to separate each argument with a comma.

Step: 3.3-3.4

```
var my_text1 = text("Our sleep data!", 100, 10);  
my_text1.color = "#EE2B4D"  
my_text1.size = 70;
```

Notes:

Property: color and size are properties of the text that you can change. Notice that the syntax is variable_name DOT property.

Step: 3.5

Notes:

Changing the position of the charts: the first two arguments of the rect() describe the x and y position of the shape. movie.height is a special 'number' – it is the pixel height of the movie window. It can be replaced with any number, or, you can add a number to it, or minus a number from it. E.g. movie.height - 100.

Changing the size of the charts: the third argument changes the width of the bar. Remember that the height is set by the data.

Code Blocks: experiment with the code blocks. You can click on the function to find out how it works, e.g. click on the word triangle in the line var my_triangle = triangle(0, 0, 100, 0, 50, 100);.

Final Code

```
// this section stores the sleep data  
var sleep1 = [5, 7.5, 9, 8.5, 5, 7, 9];  
var sleep2 = [8, 6, 5, 5, 8, 7.4, 9];  
var unit = movie.height / Math.max(...sleep1, ...sleep2) / -1.5;
```

```
// this displays the webcam
movie = webcam();
tint("#4FF3F3", 30);

// this section displays the histogram
var mon1 = rect(0, movie.height, 30, unit * sleep1[0], "#0CDF0C");
var tues1 = rect(30, movie.height, 30, unit * sleep1[1], "#4ECF4E");
var wed1 = rect(60, movie.height, 30, unit * sleep1[2], "#35E035");
var thurs1 = rect(90, movie.height, 30, unit * sleep1[3], "#3AA83A");
var fri1 = rect(120, movie.height, 30, unit * sleep1[4], "#1F961F");
var sat1 = rect(150, movie.height, 30, unit * sleep1[5], "#34C234");
var sun1 = rect(180, movie.height, 30, unit * sleep1[6], "#19DF19");

var mon2 = rect(350, movie.height, 30, unit * sleep2[0], "#F04BEC");
var tues2 = rect(380, movie.height, 30, unit * sleep2[1], "#EB6FE8");
var wed2 = rect(410, movie.height, 30, unit * sleep2[2], "#9B2098");
var thurs2 = rect(440, movie.height, 30, unit * sleep2[3], "#EE6FEB");
var fri2 = rect(470, movie.height, 30, unit * sleep2[4], "#F04BEC");
var sat2 = rect(500, movie.height, 30, unit * sleep2[5], "#D342CF");
var sun2 = rect(530, movie.height, 30, unit * sleep2[6], "#C311BF");

var my_text1 = text("Our sleep data!", 100, 10);
my_text1.color = "#EE2B4D"
my_text1.size = 70;
```