



**THE  
EDUCATION  
FUND**

FOR EXCELLENCE IN MIAMI-DADE PUBLIC SCHOOLS

**2018-2019**

# Ideas with **IMPACT**



**idea packet**

sponsored by:

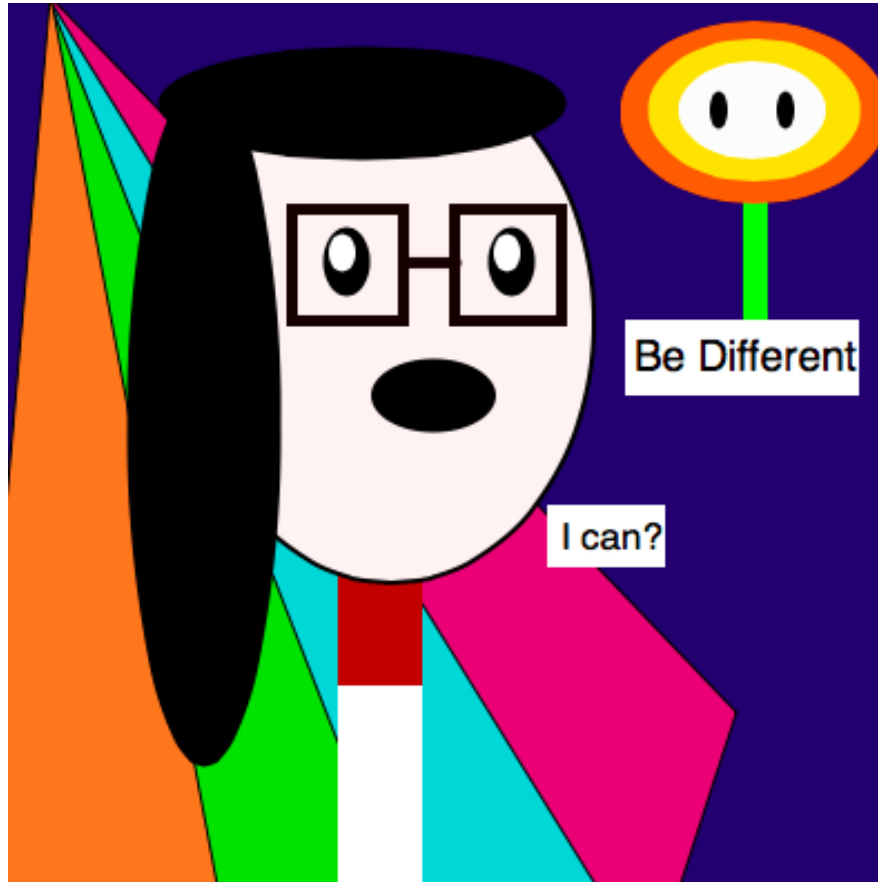


Ford Motor Company Fund

iCodeMe

# iCodeMe

Lisa Hauser  
lhauser@dadeschools.net  
cmshauser4math@gmail.com  
iPreparatory Academy  
Work Location #7581



For information concerning Ideas with IMPACT opportunities including  
Adapter and Disseminator grants, please contact:  
Debra Alamo, Interim Program Manager  
Ideas with Impact  
The Education Fund  
305-558-4544, Ext. 105  
Email: [dalamo@educationfund.org](mailto:dalamo@educationfund.org)  
[www.educationfund.org](http://www.educationfund.org)

# Table of Contents

Goals and Objectives.....	1
Sample Florida Standards.....	3
Course Outline.....	4
Lesson 1: iCodeMe.....	6
Lesson 2: Gallery Walk.....	13
Sample Self-Portrait and Code.....	14
Resources.....	18

## Goals and Objectives

The STEM problem in the United States is, truly, a computer science problem. According to the U.S. Bureau of Labor Statistics, by 2024 73% of STEM jobs created will be in computer occupations compared to 3% in physical sciences and 3% in life sciences. Moreover, in the United States, there are currently over 500,000 open computer related jobs nationwide, yet only about 50,000 students graduated into the workforce as computer scientists. The statistics are even more staggering when women and minorities are considered. Women represent less than 18% of computer scientist nationwide, while 4% of computer scientists are black and 5% are Hispanic.

Florida, one of 13 states with K-12 computer science standards, does not fare much better than the rest of the nation. In 2017, of 7,233 AP Computer Science tests given in the state, 25% of the tests were taken by female students, while 30% were taken by Hispanic students and 8% by black students. In the state of Florida, where the median income for computer related jobs hovers around \$120,000, fewer than 250 schools offer AP Computer Science. This is a problem, because high school students who take AP Computer Science are 7-8 times more likely to major in computer science.

This project aims to inspire students, especially in elementary and middle school, to consider computer science by leveraging the creative aspect of coding and allowing students to literally SEE THEMSELVES as coders. Furthermore, this highly adaptable project does not require the teacher to know how to code, making the project highly viable in any 3rd-12th grade classroom.

In this project, students create still or animated self-portraits using Java Script. All students take a picture of themselves using their device or a teacher provided camera. Then, students use Khan Academy's tutorial and compiler to recreate their portrait. Beginner coders will learn how to code basic shapes, fills, strokes, and will be introduced to variables. Advanced students will learn to add animation to their program including using loops. Spinoffs of the project include adding an image of a pet, using sounds, and coding an iconic historical figure. Final projects are printed and displayed so that students can complete a silent "gallery walk" where they provide constructive feedback to one another on the code and the artistry.

### Academic Objectives

At the end of this project, all students will be able to:

- Code a self-portrait in JavaScript

Beginner Coders will be able to:

- Code basic shapes, fills, strokes
- Understand basic variables in JavaScript

Intermediate to Advanced Coders will also be able to:

- Understand basic animation
- Understand basic interactive programming, including loops

*Behavioral Objectives*

At the end of this project, student will:

- Improve team work and collaboration capacity
- Develop creative problem solving and perseverance
- Improve confidence and self-esteem

# Sample Florida Standards

## Grades 3 - 5 Computer Science Standards

SC.35.CS-CP.2.2 Create, test, and modify a program in a graphical environment (e.g., block-based visual programming language), individually and collaboratively.

SC.35.CS-CP.2.3 Create a program using arithmetic operators, conditionals, and repetition in programs.

## Grades 6-8 Computer Science Standards:

SC.68.CS-CP.2.4 Develop problem solutions using a programming language, including all of the following: looping behavior, conditional statements, expressions, variables, and functions.

## Grades 9-12 Computer Science Standards:

SC.912.CS-CP.3.1 Create a computational artifact, individually and collaboratively, followed by reflection, analysis, and iteration.

## Course Outline

Inspired by Khan Academy's Drawing with Code lesson for Hour of Code, "iCodeMe" asks students to create a self-portrait using JavaScript. After working through the talk throughs and challenges in Khan Academy's Drawing with Code mission, students work on creating a self-portrait. The teacher can present additional challenges to her students by adding elements or requiring an animated or interactive component. Teachers may also require students to write a description of their self-portrait and/or code or to reflect on the process of coding the self-portrait.

Students would require a minimum of one hour to work through the Drawing with Code lesson and create a basic self-portrait, yet most students can create a more thorough self-portrait within three hours. Each student should be able to complete their own self-portrait, though allowing students to partner to solve problems in the code is highly recommended.

Once the final self-portraits are completed, they are submitted directly through Khan Academy for the teacher to grade using the included rubric. The teacher will compile the images and the codes for printing or will compile them in a slide presentation for students to view, give feedback and receive comments in a Gallery Walk, using a Gallery Walk protocol.

This project has been completed with students from 3<sup>rd</sup>-12<sup>th</sup> grade in a variety of settings and using varied timeframes. Some students have completed the project in a workshop over the course of several days, while others have done the project in an hour-long period and then worked on fine-tuning the portrait on their own time. Students with some coding experience have coded their self-portraits in a competition setting within 1.5 hours. The duration of the lesson is determined by the teacher considering the time limitations of their classroom.

Requirements for successful completion of the self-portraits include:

- Access to computers with keyboards
- Access to good internet connection to run Khan Academy videos
- Google Chrome or Safari are recommended web browsers, though the programs will run on Internet Explorer and Mozilla
- Students should have some typing skills
- Digital camera or computer with camera capability
- Optional: Color printer to print digital photos of students
- Optional: Class setup on Khan Academy

Requirements for a successful Gallery Walk include either:

- Color printer to print code and self-portraits
- Color display/projector or computer
- Color sticky note or color makers/pens

This project involves little instruction from the teacher, but extensive coaching. Though the teacher needs minimum coding experience to conduct this project, it is helpful if the teacher attempts to code their own self-portrait prior to assigning the project in order to become acquainted with the tutorial, documentation, and student experience.

Taking the Lesson Outside of the Classroom:

- Female students in grades 3-12 can submit their final self-portraits to a coding competition run by CodeArt Miami.
- Middle and high school female student can enter the iCodeMe competition at the Youth Fair. For more information, see the resource page at the end of the packet.



# Lesson 1: iCodeMe

## **Objectives:**

- Students will complete “Drawing with Code” lesson on Khan Academy
- Students will create a self-portrait using JavaScript.

## **Materials:**

- Computers with keyboards, preferably one for every student
- Google Chrome or Safari installed on each computer (Internet Explorer and Mozilla Firefox are suitable but not recommended)
- Khan Academy accounts for every student
- Headphones or earbuds for every student
- Digital camera or computers with cameras
- Color printer (optional)
- iCodeMe graph paper worksheet (optional)
- iCodeMe Instructions (optional)
- Class setup on Khan Academy (optional)

## **Classroom Setup:**

Every student should have access to one computer, yet students should be able to interact with one another by “budding coding” (side-by-side coding).

## **Prior Knowledge:**

Students should have some keyboarding experience.

## **Before the Lesson:**

- Teacher should log on to Khan Academy and create a class, unless teacher already has a class for her students set up in Khan Academy. For instructions on setting up a class on Khan Academy, follow this link:  
<https://www.khanacademy.org/resources/k-12-teachers-1/getting-started-on-khan-academy/a/how-to-create-a-teacher-account-and-set-up-classes>
- Teacher should assign the Hour of Code mission on Drawing with Code to the class. This will allow the teacher to monitor progress without students submitting work.
- Teacher should work through Hour of Code mission on Drawing with Code:  
<https://www.khanacademy.org/computing/hour-of-code/hour-of-drawing-code/v/welcome-hour-of-code>

- Teacher should determine challenges to setup for students (e.g. draw your self-portrait and include your pet, an animation, or an interactive element depending on coding ability and experience).
- Teacher should decide if the iCodeMe Gallery Walk will be used to help students improve their designs or if it will be used as an exhibition/culminating activity.

### **During the Lesson:**

- Take a photograph of each student to use as a reference when coding their self-portrait.
- Provide instructions to students (e.g. via Edmodo, teacher website, projection or handout). See iCodeMe Student Instruction page.
- Ensure every student has headphones or earbuds to listen to the Khan Academy tutorials.
- Allow students approximately one hour to complete Drawing with Code tutorial on Khan Academy.
- While students are working on the tutorial print the digital images of the students or email the digital images to each student.
- Monitor students to ensure they are moving through the challenges.
- Allow students to work for an additional hour to complete their self-portraits.
  - It might be helpful to distribute the iCodeMe Planning Worksheets to help students determine variable values.
- If assigning animation, students may want to work through the Animation Basics tutorial on Khan Academy:  
<https://www.khanacademy.org/computing/computer-programming/programming/animation-basics/a/what-are-animations>
- If assigning interactive elements, students may want to work through the Interactive Programs tutorial on Khan Academy:  
<https://www.khanacademy.org/computing/computer-programming/programming/interactive-programs/p/mouse-interaction>
- If the teacher wants students to complete a write up about their self-portrait, provide prompt and instructions for submission.
  - Sample Prompts:
    - Describe your image and why you chose the elements in the image.
    - Describe your code. What was the easiest element to code? What was the most challenging element to code? How did you overcome the challenges you faced while coding your self-portrait? What would you do differently?

- *Safeguard*: Students should copy and paste their code into a Word or other text editing tool and save the code. Students might also want to save a snapshot of their self-portrait.
- Students should share their code with teacher via Khan Academy by using the share button.

### **Assessments:**

- Observation of student work and collaboration.
- Work can be assessed using the iCodeMe Grading Rubric.
- Consider assigning two grades.
  - The first grade is simply for completion. Did the student complete the tutorial? Did the student submit a self-portrait? This grade can be assigned at the end of the period, or at the end of the “lesson.”
  - The second grade is for the quality of the self-portrait. For this grade, follow the iCodeMe Rubric or your own version of the rubric. This grade can be assigned after a certain time period agreed upon with students for final product submission.

### **Tips:**

- For ELL students who need translations or for student who need closed captioning, visit <https://khanacademy.zendesk.com/hc/en-us/categories/200186020-Translators>
- Students are likely to become frustrated. Rely on the documentation page to help students find the information they need. If the teacher does not know the answer, ask another student to help or review the tutorial with the student.
- Students can create numerous versions by using NEW PROJECT instead of a spinoff. Search “New Projects” to create a new project.
- Encourage students to use Pseudo-Code (`//Text`) to organize their code.
- Remember that code works like layers. Objects coded later in the program will appear over objects coded earlier.

# iCodeMe Grading Rubric

Student Name: \_\_\_\_\_ Period: \_\_\_\_\_

Total Points: \_\_\_\_\_ Grade: \_\_\_\_\_

Characteristic	5	4	3	2	1
<b>Creativity</b>	The creator thought outside the box and <b>went above and beyond the instructions</b> in the task. The student reimagined themselves and the task to create a face completely different from that of other students. The self portrait does not necessarily need to look like the student in order for the task to be completed successfully.	The self portrait had <b>some sort of creative twist</b> that made the piece stand out from other pieces. This could be from blending colors, using odd shapes, or adding other details besides regular facial features.	The student did a <b>good job in recreating their own face</b> as part of the assignment. They were able to successfully recreate their face using a variety of two-dimensional shapes.	The student showed <b>little creativity</b> in creating their self portrait. The piece is plain. This could mean little to no color or variety in shapes.	The student showed <b>no creativity</b> and stuck to the generic template given to them.
<b>Complexity</b>	The amount of detail <b>shows maximum effort</b> given by the student in the time provided. The creator <b>used complex shapes</b> to create their masterpiece. Other details besides the facial features are added such as background/background color and accessories (i.e. earrings, necklaces, shirt). Possibly their self portrait is animated.	The work shows <b>significant effort</b> to create a masterpiece. Other shapes such as ovals and triangles are used instead of limiting to squares and circles.	The piece showed <b>good effort</b> made by the creator, but the piece could have more details.	The creator showed <b>minimal effort</b> in creating the piece. The creator had the ability to do more.	There was <b>no effort</b> made when creating the piece.

<b>Self-Portrait Details</b>	Included at least <b>5 different facial features</b> such as hair, eyebrows, mouth, nose, eyes, etc.	Included at least <b>4 different facial features</b> such as hair, eyebrows, mouth, nose, eyes, etc.	Included at least <b>3 different facial features</b> such as hair, eyebrows, mouth, nose, eyes, etc.	Included at least <b>2 different facial features</b> such as hair, eyebrows, mouth, nose, eyes, etc.	Included at least <b>1 facial feature</b> such as hair, eyebrows, mouth, nose, eyes, etc.
<b>Pet Details</b>	Included at least <b>5 different facial features</b> such as hair, eyebrows, mouth, nose, eyes, etc.	Included at least <b>4 different facial features</b> such as hair, eyebrows, mouth, nose, eyes, etc.	Included at least <b>3 different facial features</b> such as hair, eyebrows, mouth, nose, eyes, etc.	Included at least <b>2 different facial features</b> such as hair, eyebrows, mouth, nose, eyes, etc.	Included at least <b>1 different facial features</b> such as hair, eyebrows, mouth, nose, eyes, etc.
<b>Efficiency of Code</b>	Student wrote the code in a <b>neat and organized</b> manner. There are <b>no redundant lines</b> of code because the creator <b>used special codes</b> such as loops and variables. It is <b>easy to understand</b> what each line of code is doing. Efficient code would have <b>comments</b> labeled, but it is not necessary.	Student wrote the code in a <b>neat and organized</b> manner. It is easy to understand what each line of code is doing.	The code is <b>relatively organized</b> . Others can understand the code after a few minutes examining.	The code is <b>not clearly organized</b> . The creator has to explain the code in order for others to understand.	The code has <b>no kind of organization</b> , which makes it hard for the creator to explain and for another person to understand.

Grading Scale:

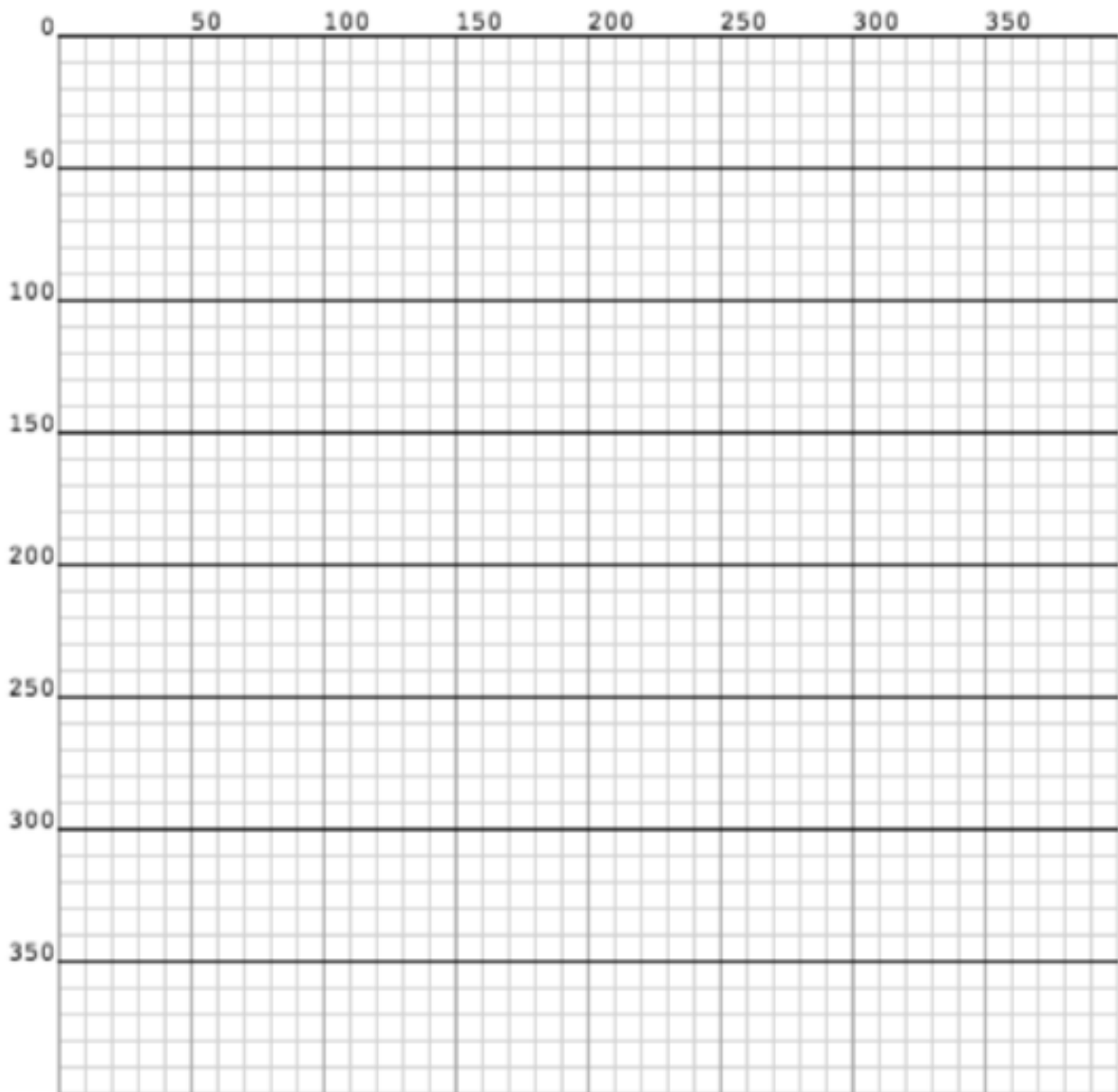
## iCodeMe Instructions for Students

- Take a picture of yourself.
- Launch Google Chrome or Safari.
- Log on to Khan Academy.
- Go to Assignments.
- Find “Drawing with Code” mission or navigate to
  - o <https://www.khanacademy.org/computing/hour-of-code/hour-of-drawing-code/p/project-self-portrait>
- You will have approximately 1 hour to complete “Drawing with Code” mission.
- At the end of the mission, you will create a self-portrait based on the picture you took at the beginning of class.
- You will have 1 hour to complete your self-portrait to resemble the picture taken of you at the beginning of class. You will also need to include an image of your pet.
- You will be judged on the attached rubric that includes points for creativity, use of complex shapes, number of details included, animation, and the efficiency of your code.
- You may use the iCodeMe Graph Paper Worksheet to help you plan your self-portrait.
- Once you have completed your code or once time has elapsed, please save your project. You should also create a backup of your code by:
  - o Copying and pasting your code into a Word document.
  - o Taking a screenshot of your self-portrait and inserting it into your Word document.
- Your program will be automatically shared with your teacher.

# iCodeMe Planning Worksheet

Student Name: \_\_\_\_\_

Period: \_\_\_\_\_ Date: \_\_\_\_\_



## Lesson 3: iCodeMe Gallery Walk

### Objectives:

- Students will be able to give and receive constructive feedback on the self-portraits.
- Students will be able to give and receive constructive feedback on the code generating the self-portraits.

### Materials:

- Code and Self-portraits
  - Computers or large displays with code and self-portrait side by side
  - Color printouts of code and self-portraits
- Sticky note pads in three colors for each student. Assign one color for “I like,” one color for “I wonder,” and one color for “Next steps.” If color sticky pads are unavailable, color pens or markers are an option.

### Activities:

- Teacher posts student self-portraits and code around the classroom.
- Students stand in front of their own work.
- Teacher asks all students to rotate around the room providing feedback to each creator. Students are not allowed to talk during the Gallery Walk.
- Students must write one thing they like about the work, one thing they wonder about the work, and one thing the coder could do next or could improve. Students should put their initials on the sticky notes they leave for coders.
- At the end, students return to their work to read feedback.
- Students can then come together as a class to debrief the experience.

### Assessments:

- Observe students providing feedback, read comments.

### Tips:

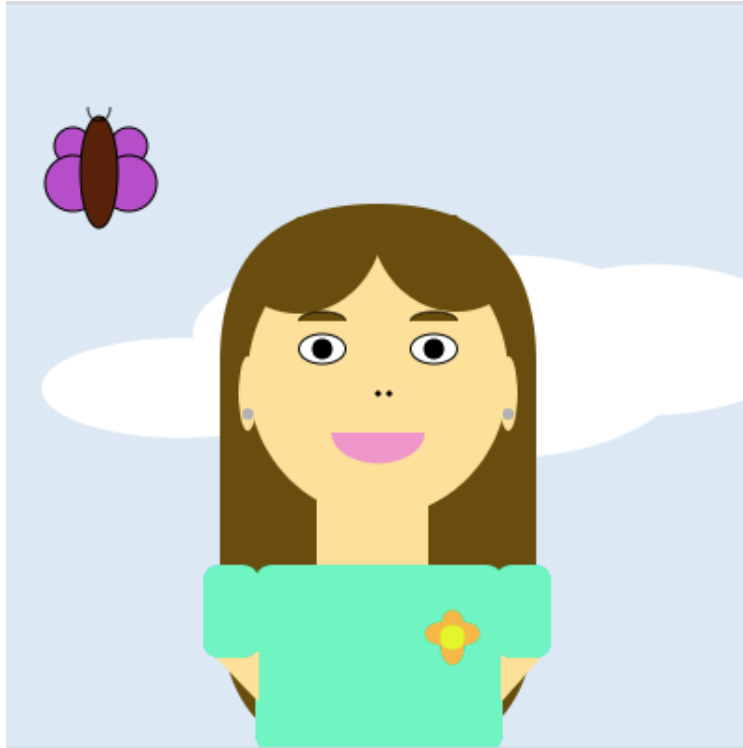
- Teachers want to actively monitor the Gallery Walk to keep students rotating and ensuring the feedback is constructive.
- The Gallery Walk can be done as a final activity or it can be done before work is submitted for final grading.
- Gallery Walks are most effective when norms are established for giving and receiving feedback. It is important for the teacher and the students to establish the norms in a previous discussion.



## Sample Self Portrait with Animation

By Hanna S., 3<sup>rd</sup> Grade Student at iPreparatory Academy

Note: Student has provided permission for use.



## Sample Code

By Hanna S., 3<sup>rd</sup> Grade Student at iPreparatory Academy

```
//HANNA
var c=0;
var speed = 2;
//background and clouds
draw = function() {
  background(220, 233, 245);
  if(c>16){
    speed=-0.1;
  }
  if(c<10){
    speed=0.1;
  }
  //move the clouds
  c = c+speed;

  //clouds
  fill(255, 255, 255);
  noStroke();
  ellipse(c+334,180,162,81);
  ellipse(c+254,189,202,109);
  ellipse(c+168,177,163,76);
  ellipse(c+80,206,150,54);
  //hair
  fill(105, 77, 10);
  rect(115,107,170,300,201);
  //head
  fill(255, 224, 153);
  ellipse(200,200,140,150);
  //bangs
  noStroke();
  fill(105, 77, 10);
  arc(242, 113, 93, 104,45, 180);
  arc(157, 114, 93, 104,0, 137);
  //neck
  noStroke();
  fill(255, 224, 153);
  rect(167,254,60,200);
  //right eyebrow
  stroke(0, 0, 0);
  fill(105, 77, 10);
```

```

arc(170,170,25,10,180,360);
//left eyebrow
fill(105, 77, 10);
arc(230,170,25,10,180,360);
//right white eye
fill(255, 255, 255);
ellipse(170,185,25,16);
//left white eye
fill(255, 255, 255);
ellipse(230,185,25,16);
//right black eye
fill(0, 0, 0);
ellipse(170,185,10,10);
//left black eye
fill(0, 0, 0);
ellipse(230,185,10,10);
//mouth
noStroke();
fill(240, 149, 202);
arc(200,230,50,33,0,180);
//nose
fill(0, 0, 0);
ellipse(206,209,3,3);
ellipse(200,209,3,3);
//ears
fill(255, 224, 153);
ellipse(130,209,10,40);
ellipse(270,209,10,40);
//earrings
fill(179, 177, 177);
ellipse(130, 220, 6, 6);
ellipse(270, 220, 6, 6);
//shirt
fill(31, 242, 193);
rect(134, 301, 133, 100, 10);
//arms
fill(255, 224, 153);
triangle(110, 348, 136, 348, 136, 375);
triangle(266, 348, 290, 348, 266, 375);
//sleeves
fill(31, 242, 193);
rect(106,301,30,50,10);
rect(263,301,30,50,10);
//flower

```

```
fill(245, 184, 71);
ellipse(240,335,13,20);
ellipse(245,338,20,13);
ellipse(240,345,13,20);
ellipse(235,338,20,13);
fill(227, 250, 20);
ellipse(240,340,13,13);

//BUTTERFLY
stroke(0, 0, 0);
fill(183, 78, 201);
ellipse(36, c+62, 20, 20);
ellipse(36, 96, 30, 30);
ellipse(66, c+62, 20, 20);
ellipse(66, 96, 30, 30);
fill(89, 35, 8);
ellipse(50, 90, 20, 60);
//antenna
noFill();
arc(51,55,10,15,0,90);
arc(49,55,10,15,44,180);

};
```

Note: This code can be copied and pasted into a Khan Academy project or another compiler to generate the self-portrait and the animations.

# Resource List

## Links to Competitions

### *Code/Art Competition Registration*

To submit self-portraits for Code/Art Miami Competition, go to:

<https://code-art.com/competition/>

### *Miami Dade Youth Fair Handbook*

Enter students to compete in iCodeMe at the Youth Fair

<http://www.thefair.me/thefair/exhibits.php#EntryForms>

or reach out to [lisa@code-art.com](mailto:lisa@code-art.com)



# Contributors with **IMPACT**

## Platinum Star

---



FORD MOTOR COMPANY FUND



School District  
Education Foundation  
Matching Grant Program



## Gold Star

---



America's Most Convenient Bank®



ASSURANT®

*P L Dodge*  
FOUNDATION



Iris Smith



## Silver Star

---



PEREZ TRADING COMPANY

**MBF** Miami  
Bayside  
Foundation

Rod and Lucy  
Petrey

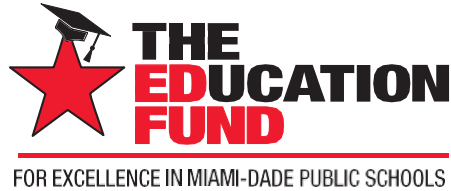
## Bronze Star

---

Raj Rawal and  
Anne Marie Miller

Robert Russell  
Memorial Foundation

Jack Chester  
Foundation



# Apply for an Ideas with **IMPACT** Adapter Grant!

All Miami-Dade County public school teachers, media specialists, counselors, or assistant principals may request funds to implement any project idea, teaching strategy, or project from the 2018 Idea EXPO workshops and/or curriculum ideas profiled annually in the **Ideas with IMPACT** catalogs from 1990 to the current year, 2018-19. Most catalogs can be viewed on The Education Fund's website at [educationfund.org](http://educationfund.org) under "Ideas with IMPACT Catalog Publications."

- Open to all K-12 M-DCPS teachers, counselors, media specialists
- Quick and easy reporting requirements
- Grants range from \$150 - \$400
- Grant recipients recognized at an Awards Reception

To apply, you must contact the teacher who developed the idea before submitting your application. Contact can be made by attending a workshop given by the Disseminator, communicating via email or telephone, by visiting the Disseminator in their classroom, or by having the Disseminator visit your classroom.

Project funds are to be spent within the current school year or an extension may be requested. An expense report with receipts is required by Monday, June 3, 2019.

**APPLICATION DEADLINE:**  
**December 13, 2018**

**Apply online at [educationfund.org](http://educationfund.org)**

**For more information, contact:**  
Audrey Onyeike, Program Director  
305.558.4544, ext. 113  
[audrey@educationfund.org](mailto:audrey@educationfund.org)