MYR: A Web-Based Platform for Teaching Coding Using VR

Presenters

Chris Berns
Grace Chin
Jason Kiesling
Vrinda Punj
Elena Izotova

Authors

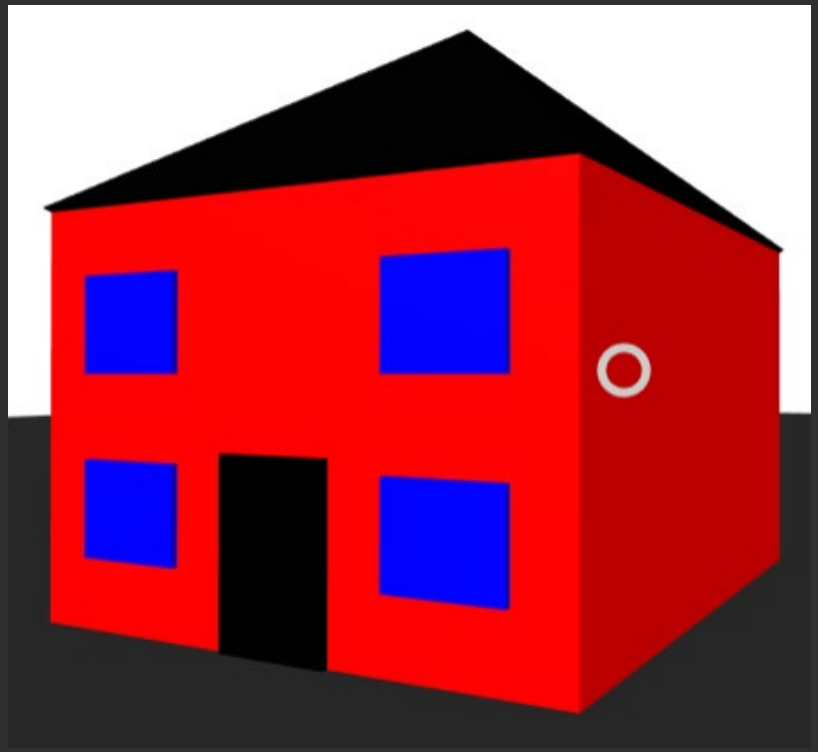
Chris Berns
Grace Chin
Joel Savitz
Jason Kiesling
Fred Martin

UMASS LOWELL

Introduction

What is MYR?

- “My Reality”
- Web-based tool for creating VR scenes
- Beginner-friendly IDE
- Engaging coding experience



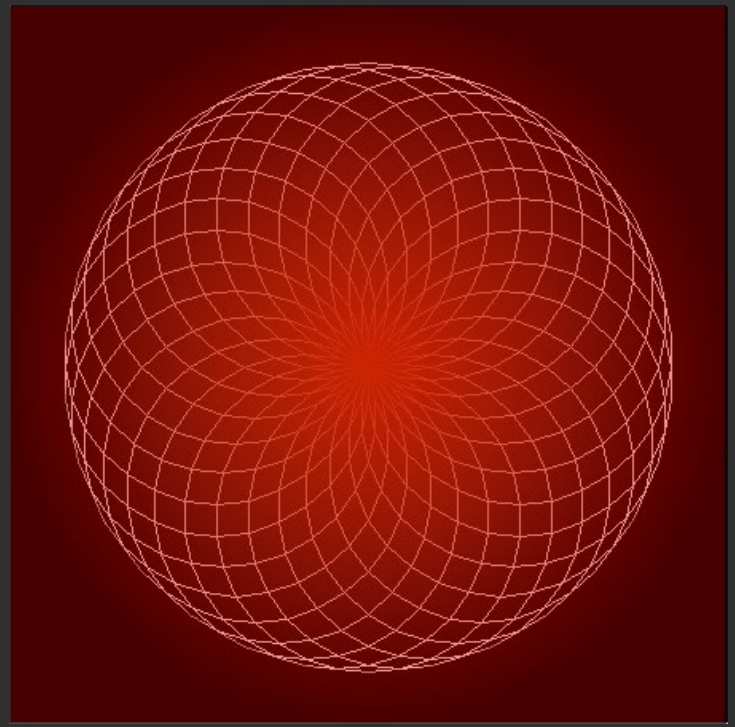
Goals

- Engaging environment for text-based coding
- Budget-friendly and easily integrated
- Aid research for CS education

Background

Prior work

- Syntactic simplicity →
  Better for learning
- Visual, interactive context →
  Computational thinking
- MYR influenced by
  - Logo (1967)



Prior work

- Syntactic simplicity → Better for learning
- Visual, interactive context → Computational thinking
- MYR influenced by
  - Logo (1967)
  - Processing (2001)

Prior work

- Syntactic simplicity →
  Better for learning

- Visual, interactive context →
  Computational thinking

- MYR influenced by
  - Logo (1967)
  - Processing (2001)
  - Scratch (2009)

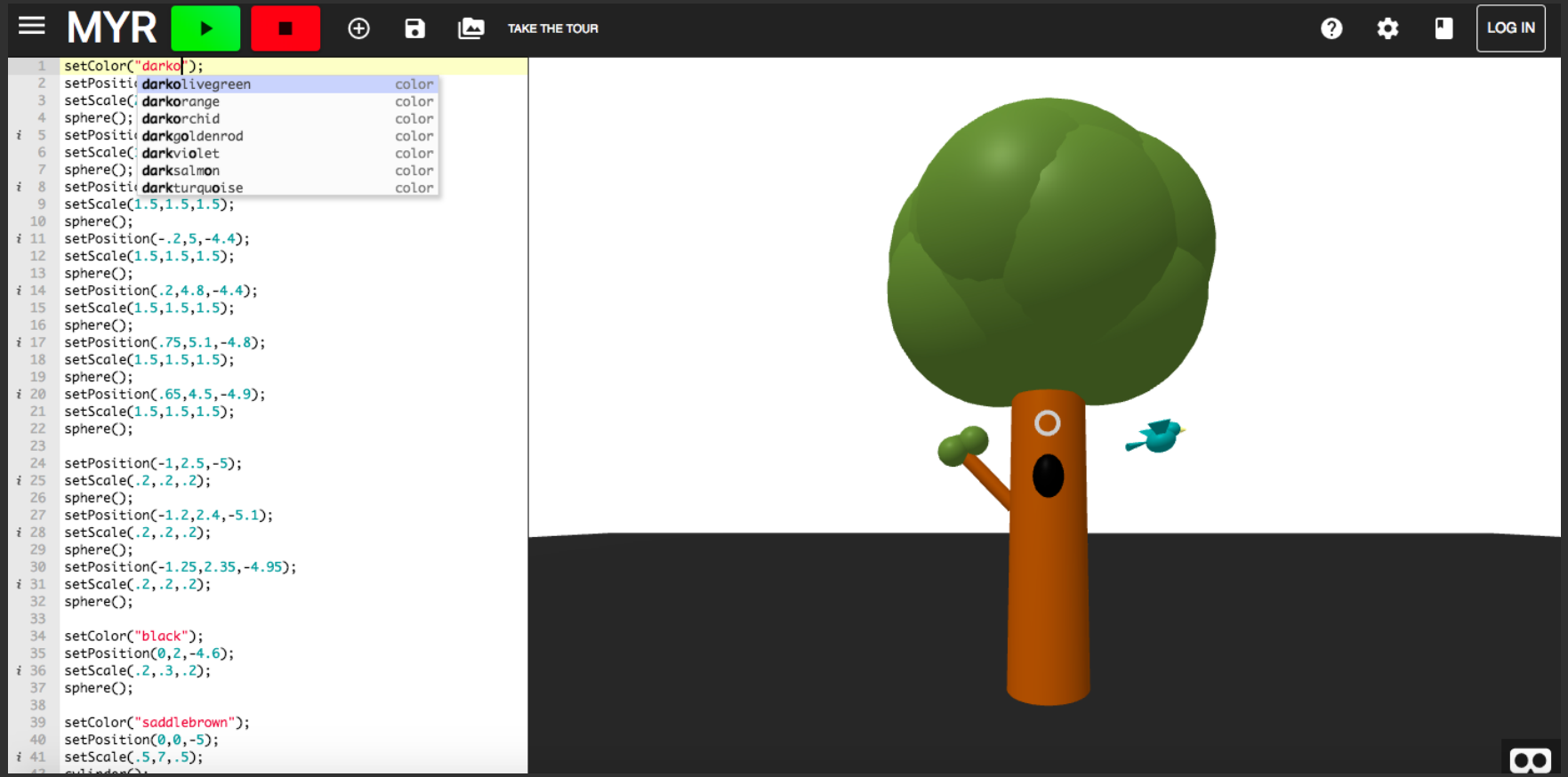
Virtual reality

- **Immersive** → Enhancement of learning experience
- **Web-based** → Affordability & ease of set-up
- **Beginner-friendly** → Usability as an educational tool

	Immersive	Web-Based	Beginner-Friendly
Unity			
A-Frame			
MYR			

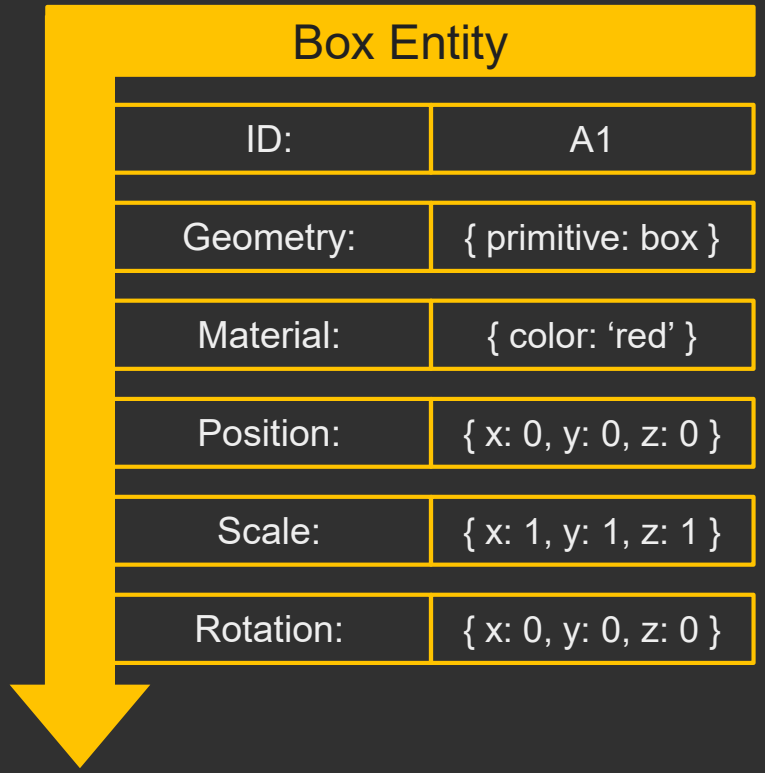
Overview of MYR

IDE


MYR
TAKE THE TOUR
LOG IN
setColor("darko");
darkolivegreen color
darkorange color
darkorchid color
darkgoldenrod color
darkviolet color
darksalmon color
darkturquoise color
setScale(1.5,1.5,1.5);
sphere();
setPosition(-.2,5,-4.4);
setScale(1.5,1.5,1.5);
sphere();
setPosition(.2,4.8,-4.4);
setScale(1.5,1.5,1.5);
sphere();
setPosition(.75,5.1,-4.8);
setScale(1.5,1.5,1.5);
sphere();
setPosition(.65,4.5,-4.9);
setScale(1.5,1.5,1.5);
sphere();
setPosition(-1,2.5,-5);
setScale(.2,.2,.2);
sphere();
setPosition(-1.2,2.4,-5.1);
setScale(.2,.2,.2);
sphere();
setPosition(-1.25,2.35,-4.95);
setScale(.2,.2,.2);
sphere();
setColor("black");
setPosition(0,2,-4.6);
setScale(.2,.3,.2);
sphere();
setColor("saddlebrown");
setPosition(0,0,-5);
setScale(.5,7,.5);

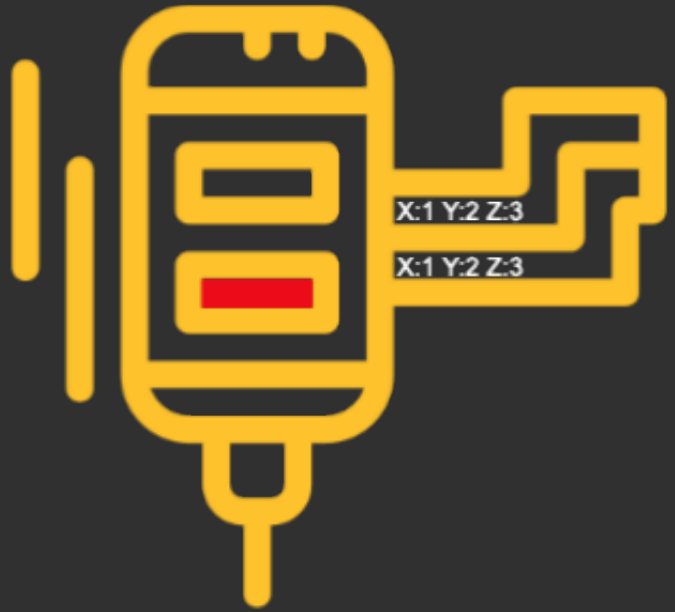
Entity-component-system

- Components are basic building blocks
- Entities are collections of components
- Separation of concerns
- Composition over inheritance
- Emerging way to deal with complexity

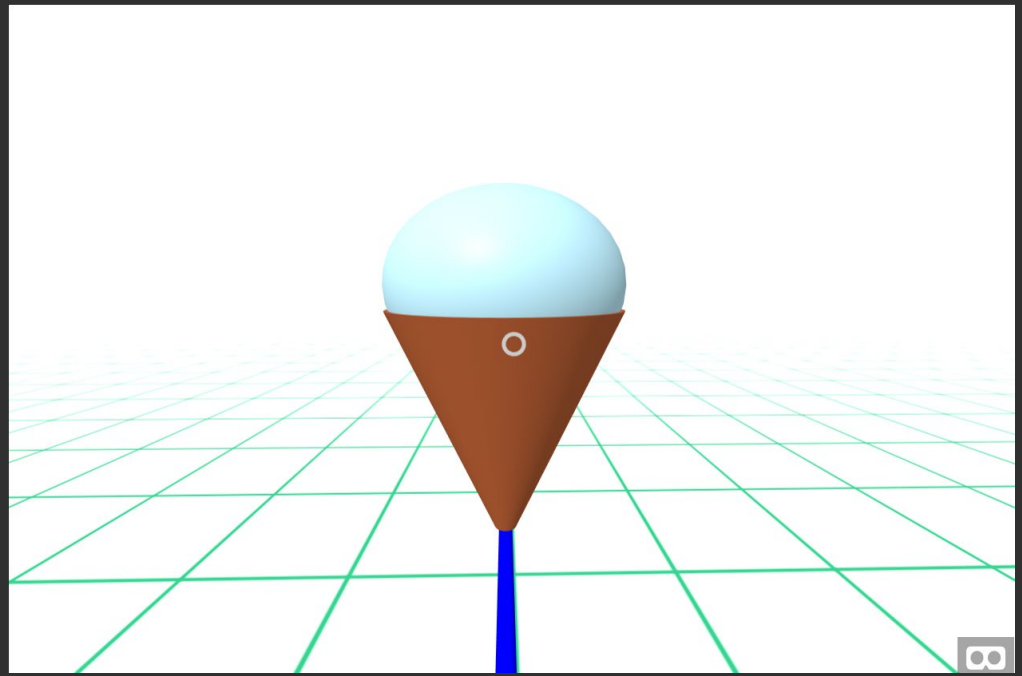


Cursor

- Produces components
- Creates entities
- Ensures defaults for all geometries
- Only change one component at a time
- Supports an iterative approach



An example of the MYR API



```
➔ setColor('#A0522D');
➔ setPosition(0, 1, -2.5);
➔ setScale(0.5, 1, 0.5);
➔ setRotation(180, 0, 0);
➔ cone();
➔ setColor('lightblue');
➔ setPosition(0, 1.6, -2.5);
➔ setScale(0.5, 0.4, 0.5);
➔ sphere();
```

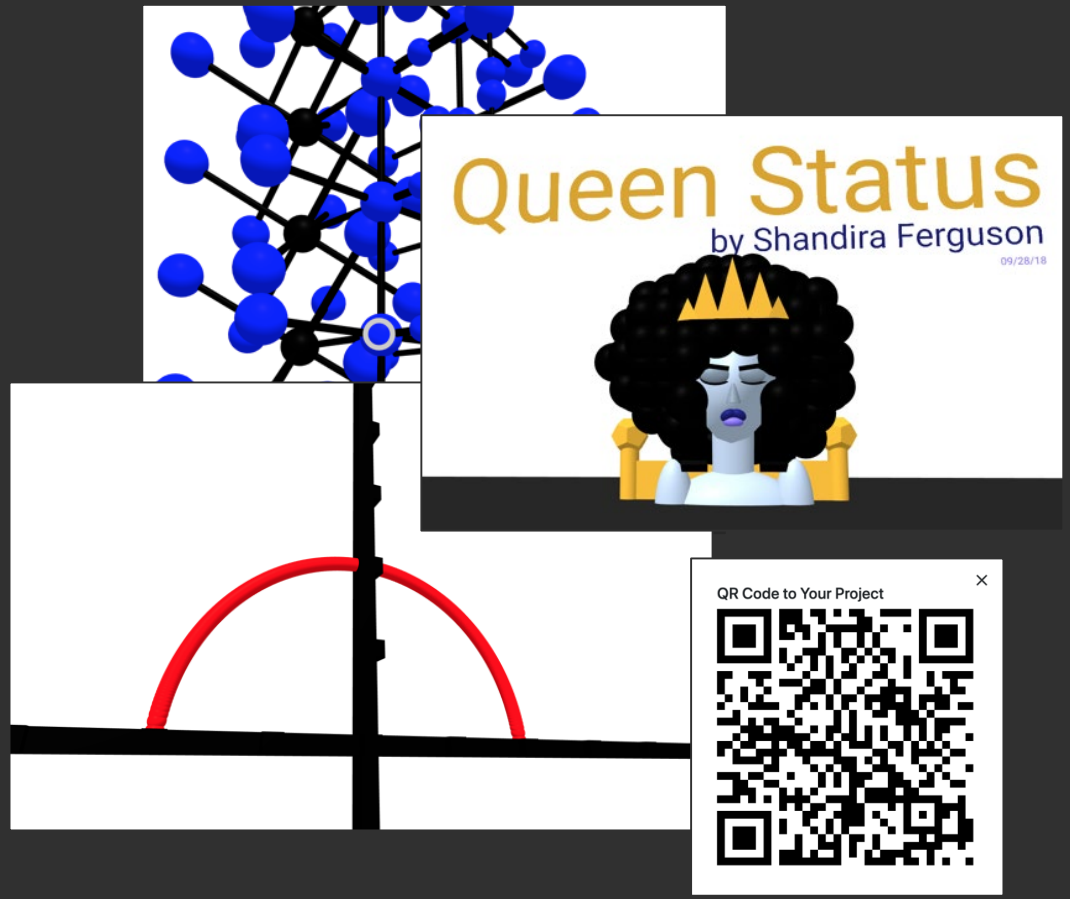
Application Features

Real-time sync

- Automatically sends the scene to all devices
- Key to rendering more often
- Collaboration and instructor-lead teaching
- All through the browser
  - No software to install
  - No cables
  - No special networks

Sharing

- Critical to learn from others
- Shareable via URL
- Easily modify and extend examples



Snapshot capture

- Creativity + Research
- Store application state
  - Creates a timeline
- Conduct research
  - Time to try
  - Error rates
  - User choices

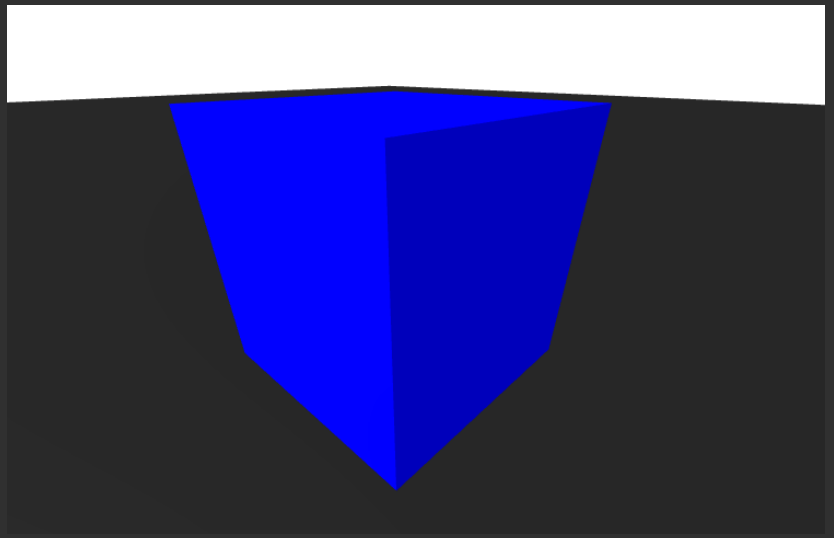
Pilot Testing

Context

- App Inventor summer camp
- 13 middle schoolers
  - Little to no experience with text-based programming
- Students asked to complete tasks by modifying MYR starter code
  - Observed students' performance
  - Gathered feedback from students

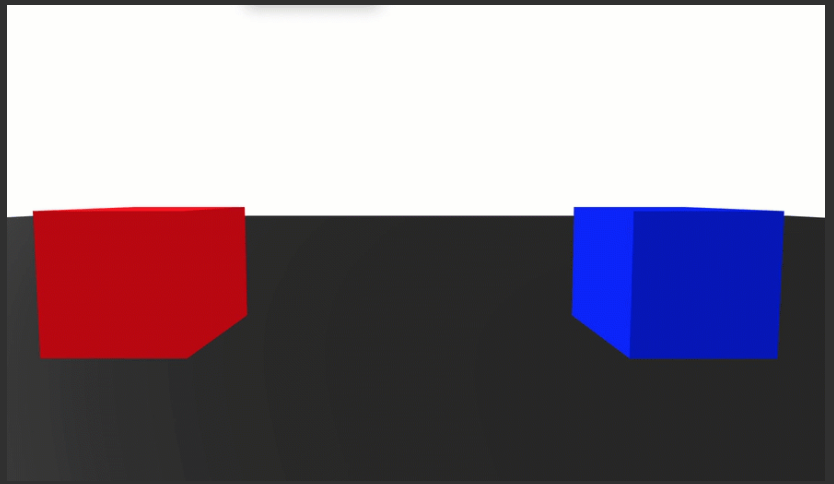
Identify the line that sets the color of a box

```
setColor('blue');
setPosition(0, 1, -2);
box();
```



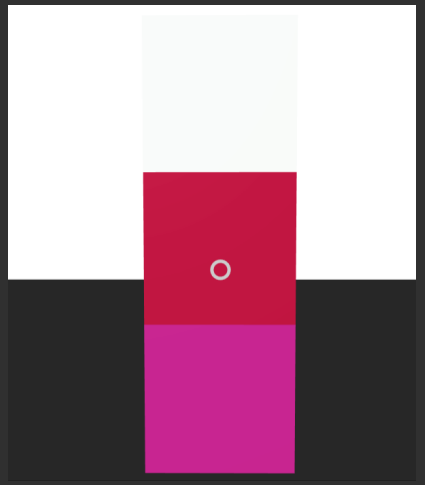
Given 2 boxes, draw another box in between

```
setColor('red');
setPosition(2, 1, 0);
box();

setColor('blue');
setPosition(-2, 1, 0);
box();
```



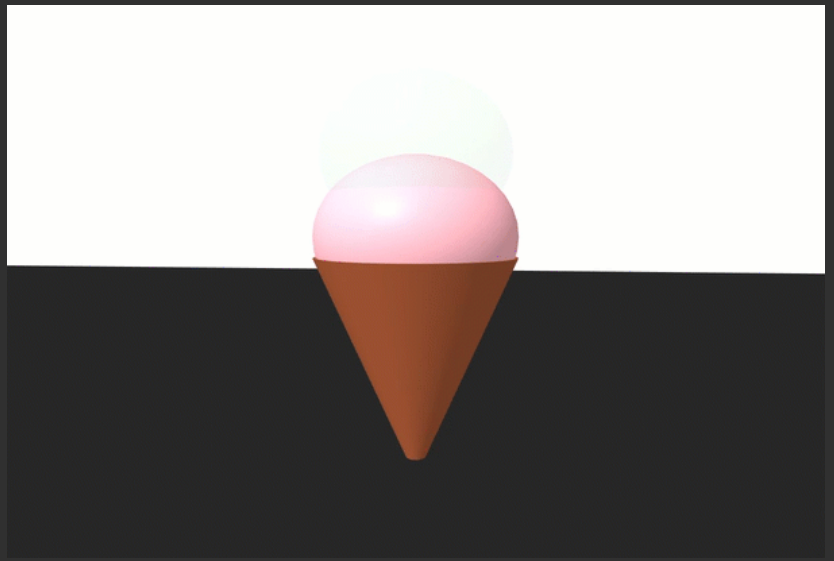
Add another box to the function

```
function stack(x, z) {
        setPosition(x, 1, z);
        box();
        setPosition(x, 2, z);
        box();
}

stack(0, -2);
```



Given an ice cream code, add another scoop

```
setColor('sienna');
setPosition(0, 2, -2.5);
setScale(.5,1,.5);
setRotation(180, 0, 0);
cone();

setColor('#98ff98');
setPosition(0, 3, -2.5);
setScale(.47,.4,.47);
sphere();
```



What do you like about using a VR educational tool?

Visually seeing the code output

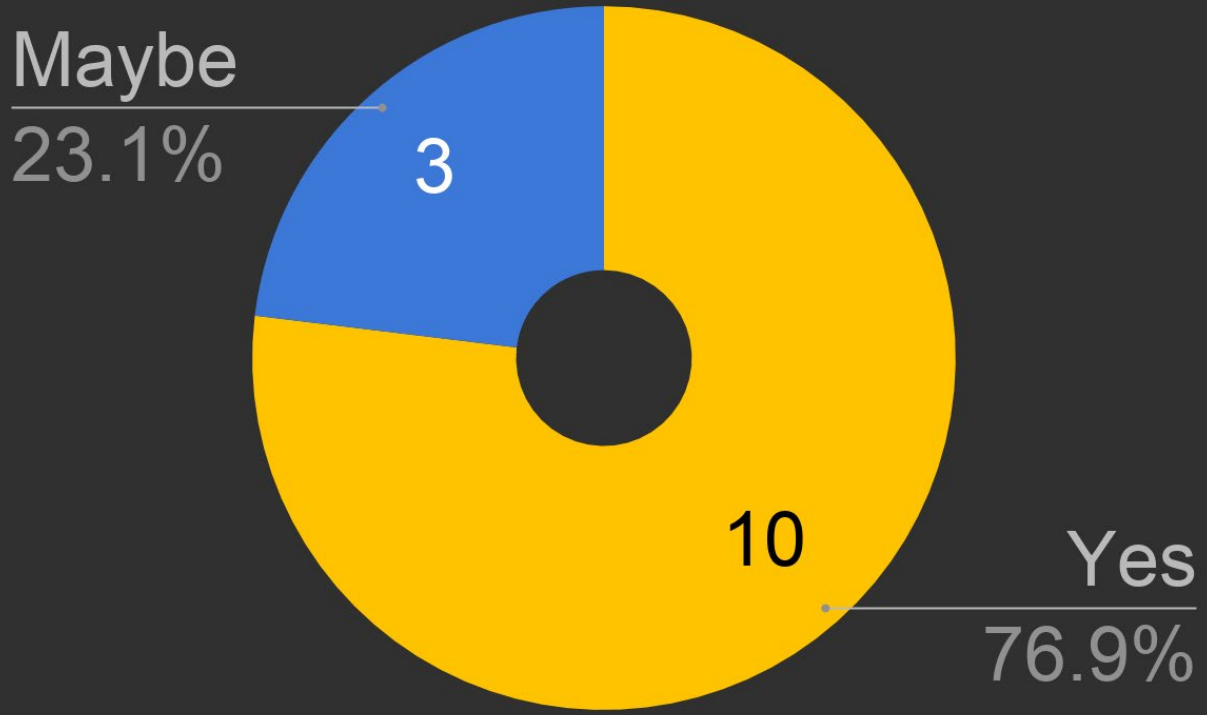
Easily viewing it via a smartphone

“Everything”

High level of control

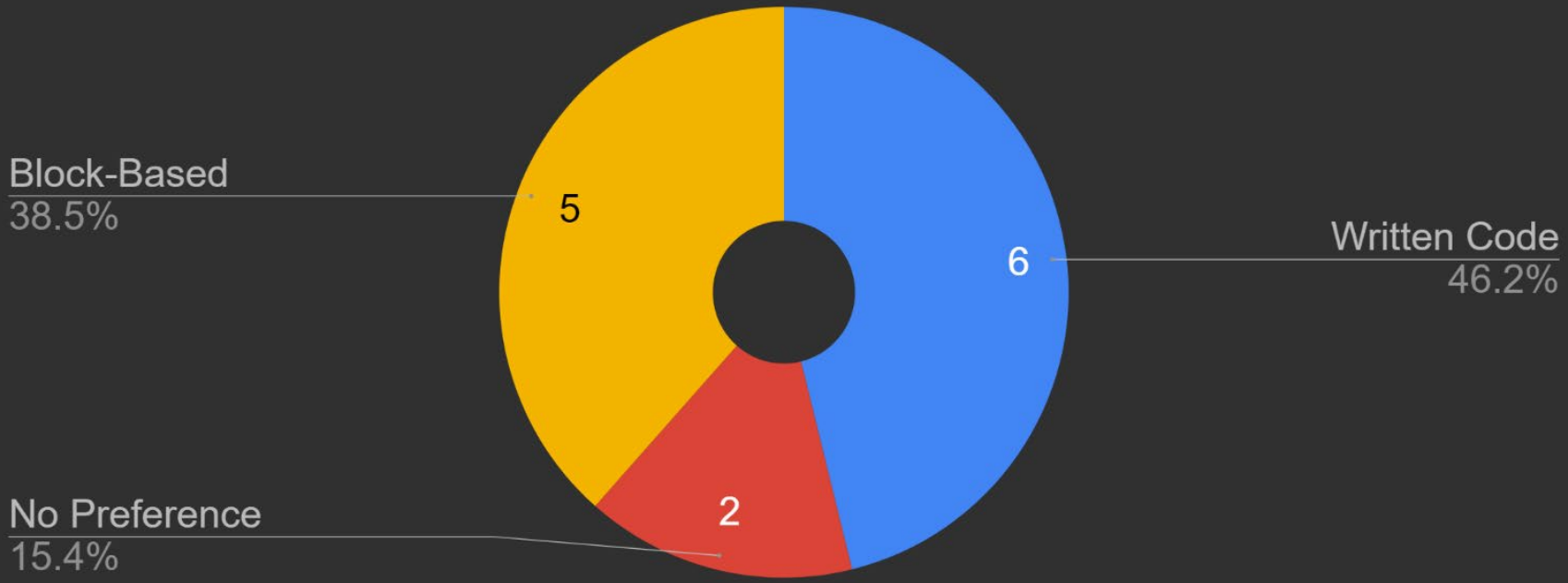
Creating new worlds

Are you interested in exploring MYR more?



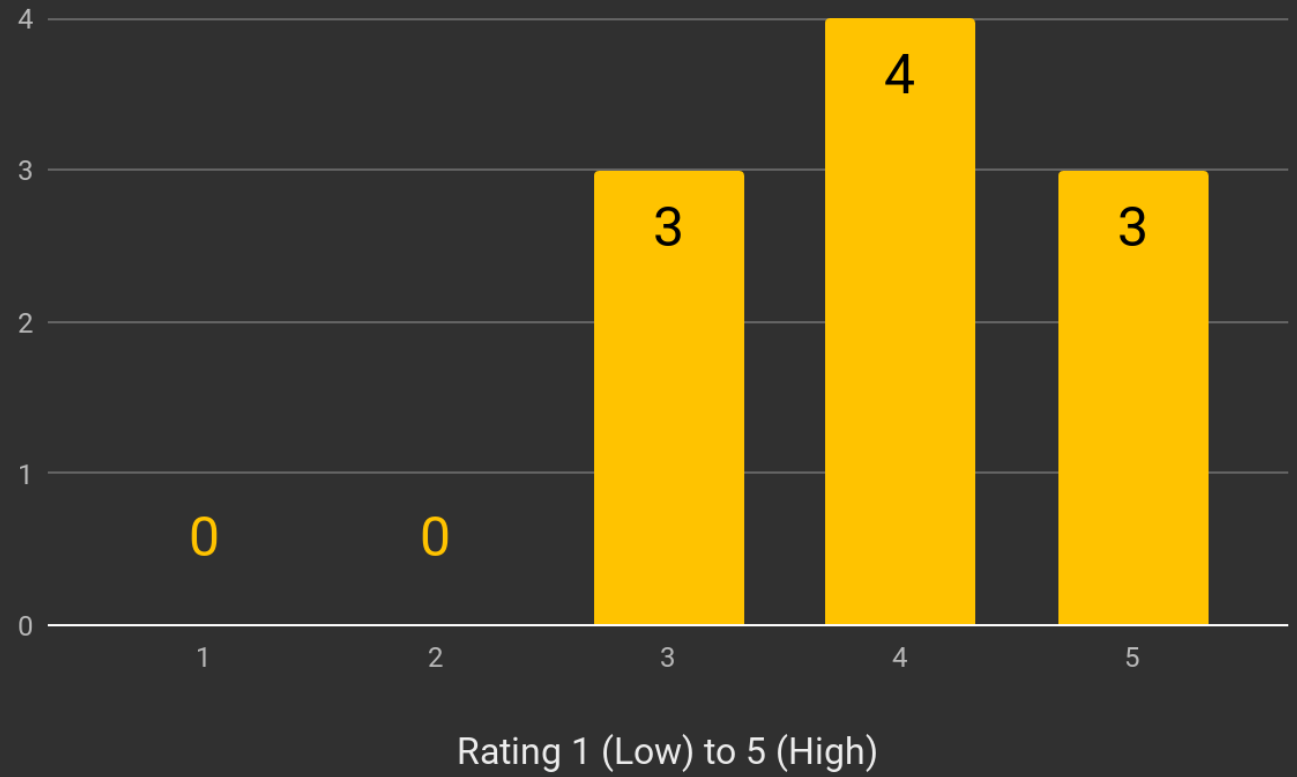
Above is a pie chart with “Yes” having 10 votes (76.9%), “Maybe” having 3 votes (23.1%), and “No” having 0 votes (0%).

Do you prefer using blocks or written code?



Above is a pie chart with “Block-Based” having 6 votes (38.5%), “Written Code” having 6 votes (46.2%), and “No Preference” having 2 votes (15.4%).

Rate how much VR enhances your coding experience

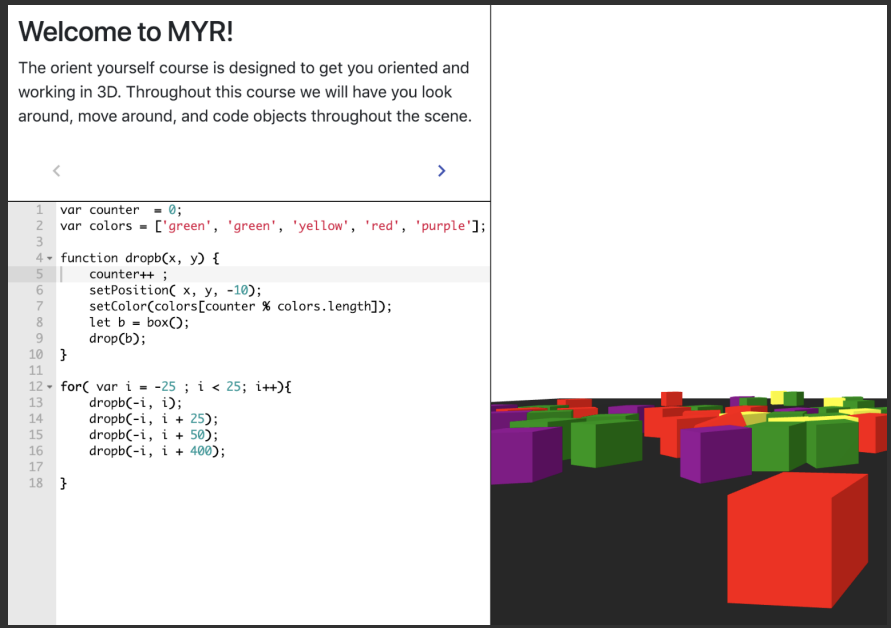


Above is a bar chart with 5 having 3 votes, 4 having 4 votes, 3 having 3 votes, 2 and 1 having 0 votes, and no response with 3 votes.

Future Work

Integrated curricula

- Takes the burden off of educators
- Helps teachers unfamiliar with CS
- Facilitates learning for all users
- Eliminates communication gaps - everything is in one place



Conclusions

Conclusions

- Can be used in the classroom + at home
  - Easily integrated with classroom technology
- Highly engaging
  - Students remain enthusiastic despite obstacles
- A new shareable artwork platform
- Seek to create partnerships with educators

The #MadeWithMYR Challenge

Step 1: CREATE your own MYR scene.

Step 2: SUBMIT your scene by March 7 via

- **LearnMYR.org/SIGCSE**
- *and/or* **any social media**

Step 3: WIN a free MYR t-shirt by

- **wowing the developer team**
- *or* **getting the most likes** on social media

Read the full contest details at LearnMYR.org/SIGCSE.

This material is based upon work supported by the National Science Foundation under Grant No. 1433592.