

The Impact of Creative Coding on Student Engagement and Learning

EXPLORING PLATFORMS, ACCESSIBILITY, AND CURRICULUM INTEGRATION

BY GERARD VELLA
NEWARK SCHOOL MALTA

What is Creative Coding?

Creative coding refers to writing code for expressive purposes, including art, music, and interactive applications. Differentiates from traditional programming focused on functionality, emphasizing creativity and problem-solving.

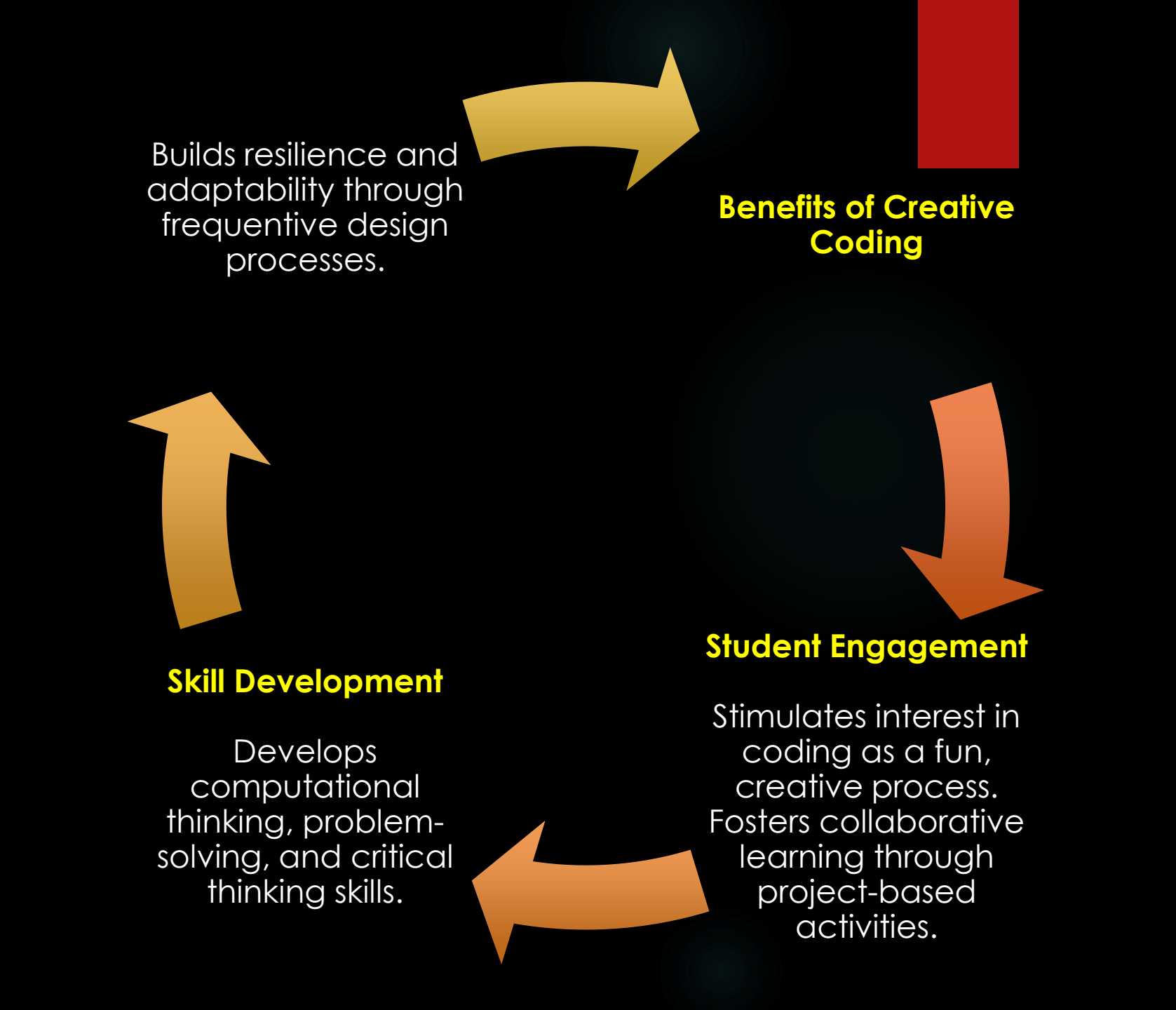
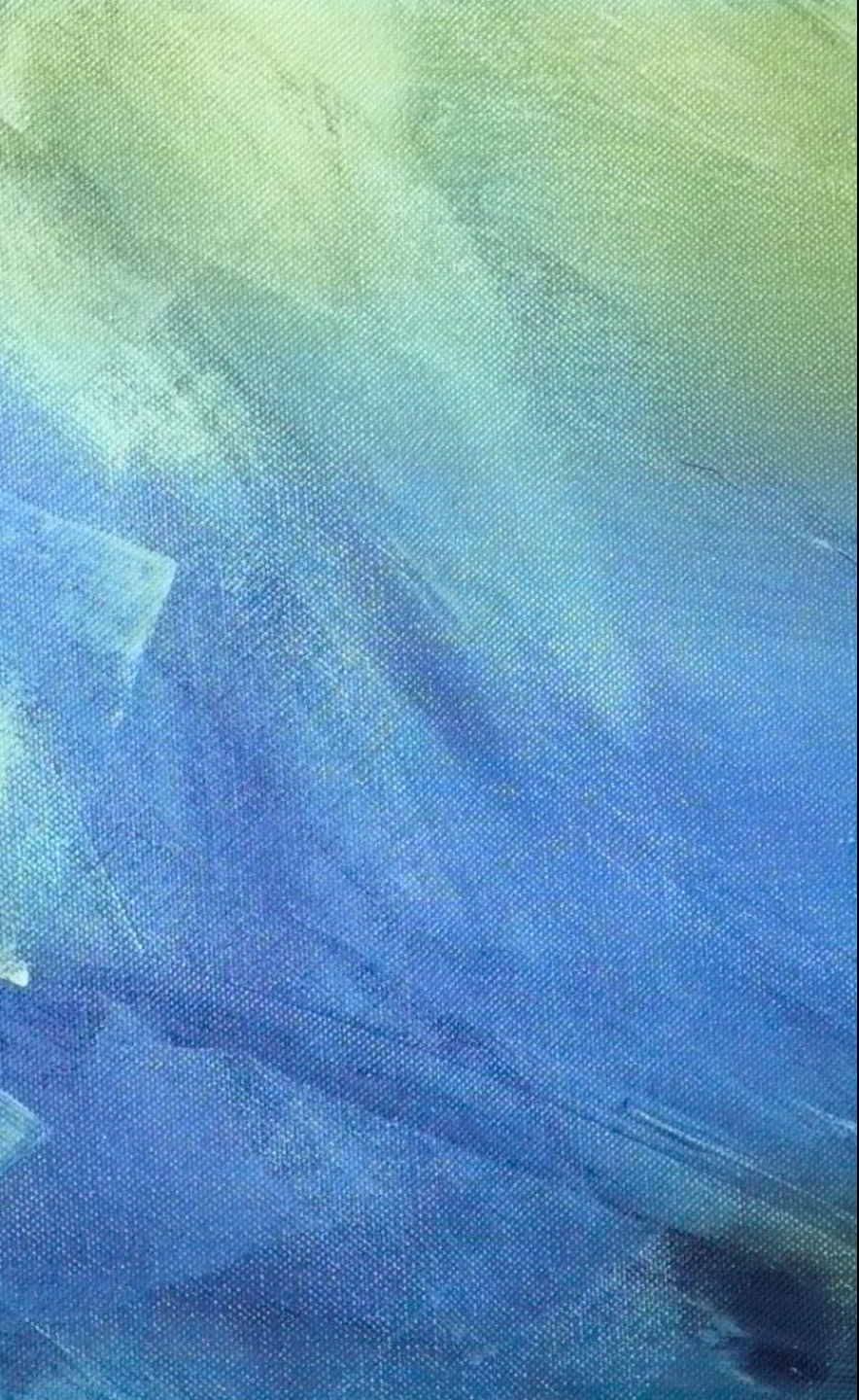
```
mirror_mod = modifier_ob.  
set mirror object to mirror  
mirror_mod.mirror_object =  
operation == "MIRROR_X":  
mirror_mod.use_x = True  
mirror_mod.use_y = False  
mirror_mod.use_z = False  
operation == "MIRROR_Y":  
mirror_mod.use_x = False  
mirror_mod.use_y = True  
mirror_mod.use_z = False  
operation == "MIRROR_Z":  
mirror_mod.use_x = False  
mirror_mod.use_y = False  
mirror_mod.use_z = True  
  
selection at the end -add  
mirror_ob.select= 1  
modifier_ob.select=1  
context.scene.objects.active  
("Selected" + str(modifier_ob.  
mirror_ob.select = 0  
= bpy.context.selected_object  
data.objects[one.name].select  
print("please select exactly  
-- OPERATOR CLASSES --  
  
types.Operator):  
X mirror to the selected  
object.mirror_mirror_x"  
mirror X"  
  
context):  
context.active_object is not
```



What is the Importance of Creative Coding?

- Engagement through Creativity
- Encourages students to express themselves.
- Bridges the gap between computing and arts.
- Enhances motivation and participation in STEM subjects.

(STEM subjects are Science, Technology, Engineering, and Mathematics. STEM covers a broad range of disciplines and are the foundation for innovation, technology, and problem-solving in today's world).



Builds resilience and adaptability through frequentive design processes.

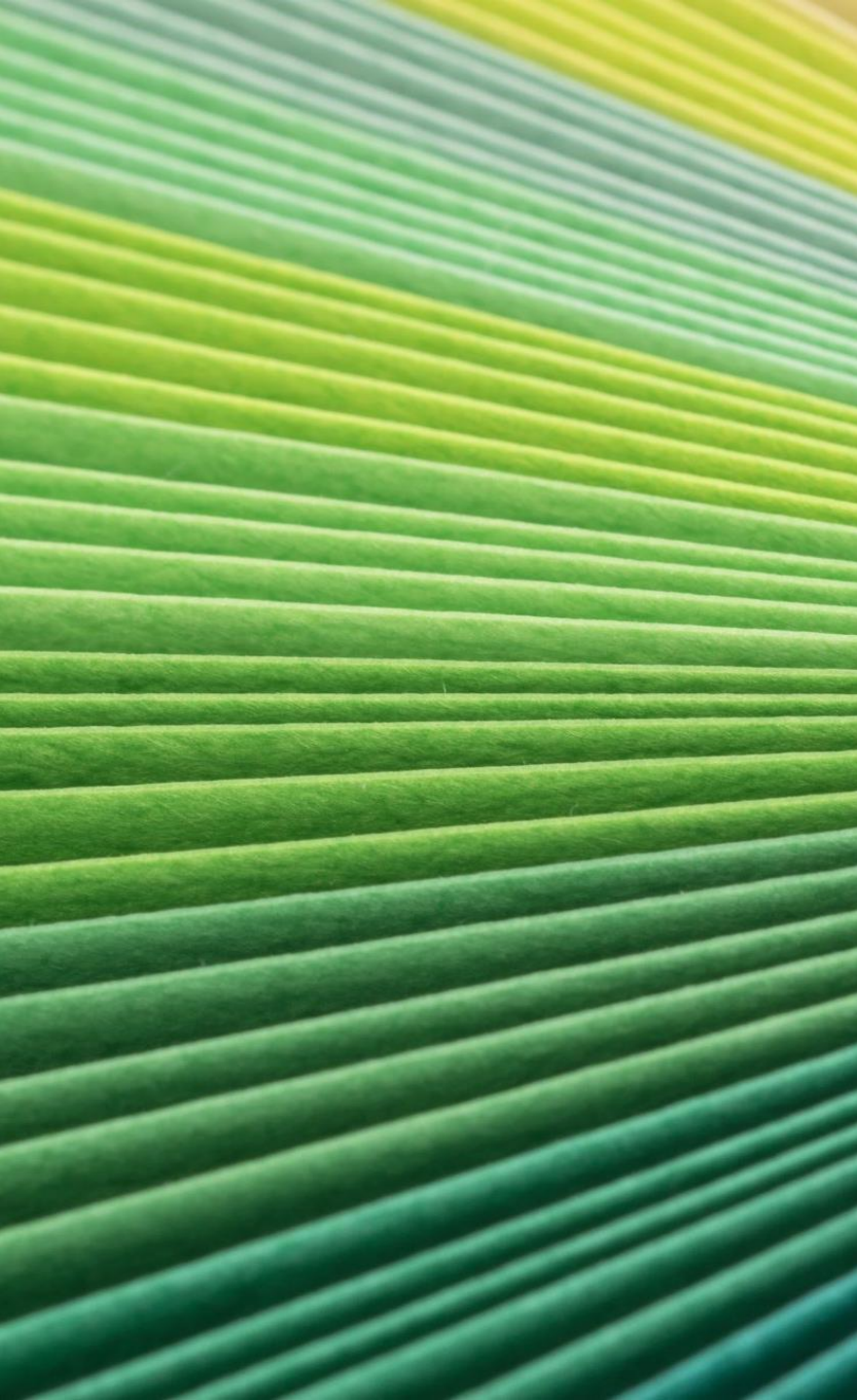
Benefits of Creative Coding

Skill Development

Develops computational thinking, problem-solving, and critical thinking skills.

Student Engagement

Stimulates interest in coding as a fun, creative process. Fosters collaborative learning through project-based activities.



Coding Platforms Overview.

Popular Platforms for Creative Coding are:

- **Scratch**

Visual programming language tailored for younger learners.
Drag-and-drop interface promotes accessibility and ease of use.

- **Processing**

A flexible software sketchbook and a language for learning how to code within the context of the visual arts.
Designed for artists, designers, and beginners.

- **Pico-8**

Fantasy console for making, sharing, and playing tiny games and other computer programs.
Encourages creativity within technical constraints.

- **Tynker**

Offers a gamified coding experience for children with various interactive coding puzzles.
Incorporates storytelling and game design elements.

Accessibility of Coding Platforms

Importance of Accessibility:

Wide range of tools ensures that coding is accessible regardless of background or experience.

Key Features to Enhance Accessibility:

User-Friendly Interfaces: Like that of Scratch, which reduces entry barriers for younger users.

Community Support: Platforms often have robust online communities for sharing resources and support.

Adaptability for Various Learning Needs: Many platforms can accommodate different learning styles.

Curriculum Integration

How Creative Coding Can Be Integrated into the Curriculum

Cross-Disciplinary Projects: Combining coding with subjects like art, science, and mathematics.

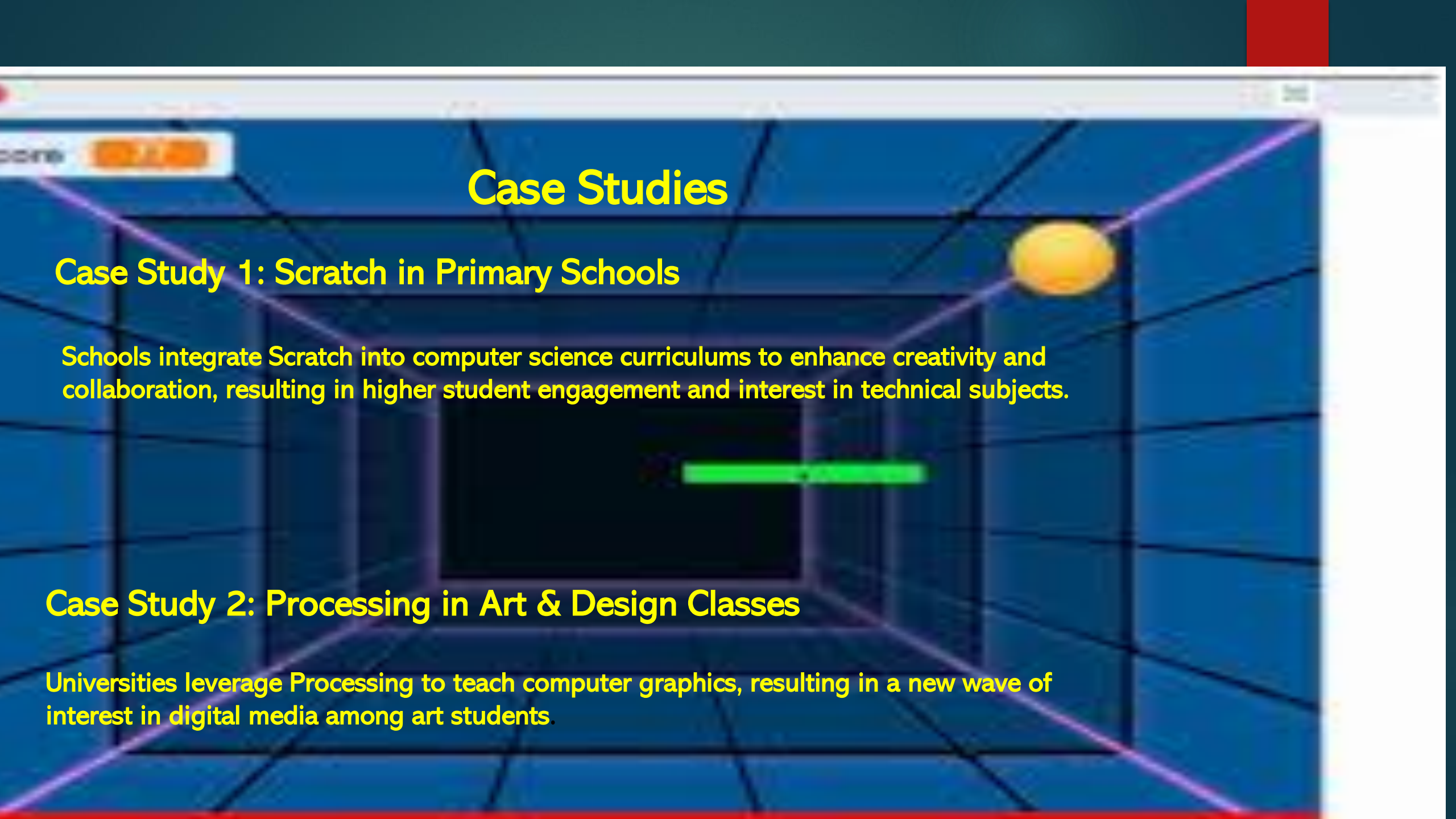
Real-World Applications: Using coding to solve problems in local communities or global issues.

STEAM Education: Integrating arts into STEM to promote holistic education.

Examples:

Using Scratch to create interactive stories in literacy classes.

Employing Processing for data visualization projects in math or statistics.

The background of the slide is a screenshot of a Scratch game. It features a blue sky with a grid pattern, a yellow sun in the top right, and a green horizontal bar in the center. The title 'Case Studies' is written in yellow text at the top center.

Case Studies

Case Study 1: Scratch in Primary Schools

Schools integrate Scratch into computer science curriculums to enhance creativity and collaboration, resulting in higher student engagement and interest in technical subjects.

Case Study 2: Processing in Art & Design Classes

Universities leverage Processing to teach computer graphics, resulting in a new wave of interest in digital media among art students.

Challenges to Adoption

Potential Barriers

1. Inadequate teacher training or professional development.
2. Limited access to technology in underfunded schools.
3. Resistance to change traditional teaching methods.

Solutions:

Provide professional development for educators.

Establish grants or funding sources to secure necessary technology.

Foster a culture of innovation within school systems to support **STEM** education.

What is STEM?

STEM is an approach to education that focuses on the hard sciences, develops critical thinking skills, and improves problem-solving abilities.

S

Science

Biochemistry or
Biophysics



Environmental
Sciences/
Hydrology/
Geoscience



Medicine

T

Technology

Computer Science



Industrial Design



Information
Technology

E

Engineering

Aerospace
Engineering



Biomedical
Engineering



Electrical
Engineering

M

Math

Applied
Mathematics



Data Science



Statistics

STEEM

Science

Technology

Engineering

Mathematics





TECH TRENDS

Conclusion

Summary of Key Points:

Creative coding significantly enhances student engagement and learning through accessibility and integration into existing curricula.

Offers a unique avenue for students to develop crucial skills while exercising creativity.

Call to Action:

Encourage educators and institutions to explore creative coding platforms as a part of their curriculum.

This Presentation offers a framework for highlighting the significance of creative coding in education, its potential platforms, accessibility for students, and practical curricular applications.

Thank you for
Listening 😊