Project-based Learning with Scratch

Jared O'Leary
BootUp PD

What's the plan?

- Project-based learning?
- Explore Scratch projects
- Discussion

How to reach the resources

- www.JaredOLeary.com
  - Presentations
    - Project-based Learning with Scratch (ISTE)

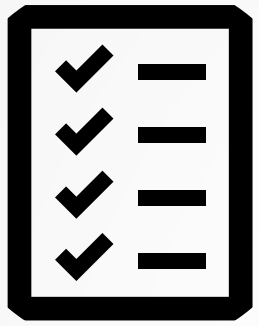
Project-based learning?

- “Project-based learning is built on the idea that real-life problems capture student interest and provoke critical thinking and develop skills as they engage in and complete complex tasks that typically result in a realistic product, event, or presentation to an audience." (p. 40)

Tobias, E. S., Campbell, M. R., & Greco, P. (2015). Bringing Curriculum to Life: Enacting Project-Based Learning in Music Programs. *Music Educators Journal*, 102(2), 39–47

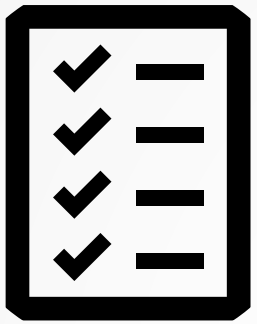
1. Central to the curriculum
2. Organized around driving questions
3. Focused on a constructive investigation
4. Student-driven
5. Authentic

Tobias, E. S., Campbell, M. R., & Greco, P. (2015). Bringing Curriculum to Life: Enacting Project-Based Learning in Music Programs. *Music Educators Journal*, 102(2), 39–47



Fixed

Project continuum



Fixed

Open

Project continuum

Fixed

Open

Project continuum

Example: Fixed project criteria

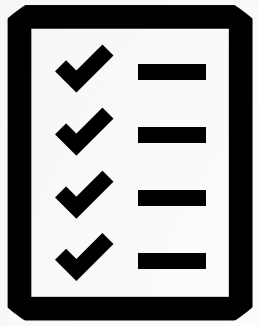
- Game
- One player sprite
- Three enemy sprites
- At least two “if _ then” blocks
- At least one variable

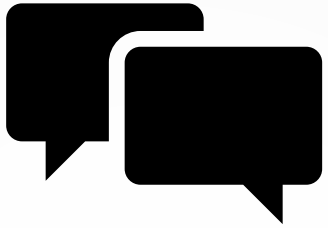
Example: Open project questions

- Can you create a school appropriate project that…
  - …helps someone?
  - … is scary, funny, exciting, boring, musical, silly, relaxing, or colorful?
  - … solves a problem you see in the world?
  - … reminds you of a special event, story, or place?
  - … you can give as a gift to someone else?
  - … you can use for another class?

Example: Open project questions

- Can you create a **school appropriate** project that…
  - …helps someone?
  - … is scary, funny, exciting, boring, musical, silly, relaxing, or colorful?
  - … solves a problem you see in the world?
  - … reminds you of a special event, story, or place?
  - … you can give as a gift to someone else?
  - … you can use for another class?





Fixed Flexible Open

Project continuum

Example: Flexible prompts with embedded criteria

- What type of project can you create that includes at least two “if _ then” blocks and at least one variable?
- How might you create a game that keeps track of a score?
- Storyboard and create a superhero(ine) project that uses several different “Events” blocks.

Example: Storyboard questions

- What sprite(s) will you use as superhero(ines)?
  - What kind of superpowers or technology will they have?
  - Will they transform into their superhero(ine) costume or always be a superhero(ine)?
    - If they are transforming, what will they look like normally? What will they look like when they are a superhero(ine)?
- Who will the superhero(ines) try and save?
  - What kind of danger are they in?
  - If it's another sprite, what kind of powers or technology will they use?
- How might your superhero(ine) save the day?
  - What algorithms can you create to do that?
- Will users be able to interact with your superhero(ine) project?
  - If so, what kind of code will you use to create that interaction?

Example: Storyboard questions

- What sprite(s) will you use as superhero(ines)?
  - What kind of superpowers or technology will they have?
  - Will they transform into their superhero(ine) costume or always be a superhero(ine)?
    - If they are transforming, what will they look like normally? What will they look like when they are a superhero(ine)?
- Who will the superhero(ines) try and save?
  - What kind of danger are they in?
  - If it's another sprite, what kind of powers or technology will they use?
- How might your superhero(ine) save the day?
  - What algorithms can you create to do that?
- Will users be able to interact with your superhero(ine) project?
  - If so, what kind of code will you use to create that interaction?

Example: Storyboard questions

- What sprite(s) will you use as superhero(ines)?
  - What kind of superpowers or technology will they have?
  - Will they transform into their superhero(ine) costume or always be a superhero(ine)?
    - If they are transforming, what will they look like normally? What will they look like when they are a superhero(ine)?
- Who will the superhero(ines) try and save?
  - What kind of danger are they in?
  - If it's another sprite, what kind of powers or technology will they use?
- How might your superhero(ine) save the day?
  - What algorithms can you create to do that?
- Will users be able to interact with your superhero(ine) project?
  - If so, what kind of code will you use to create that interaction?

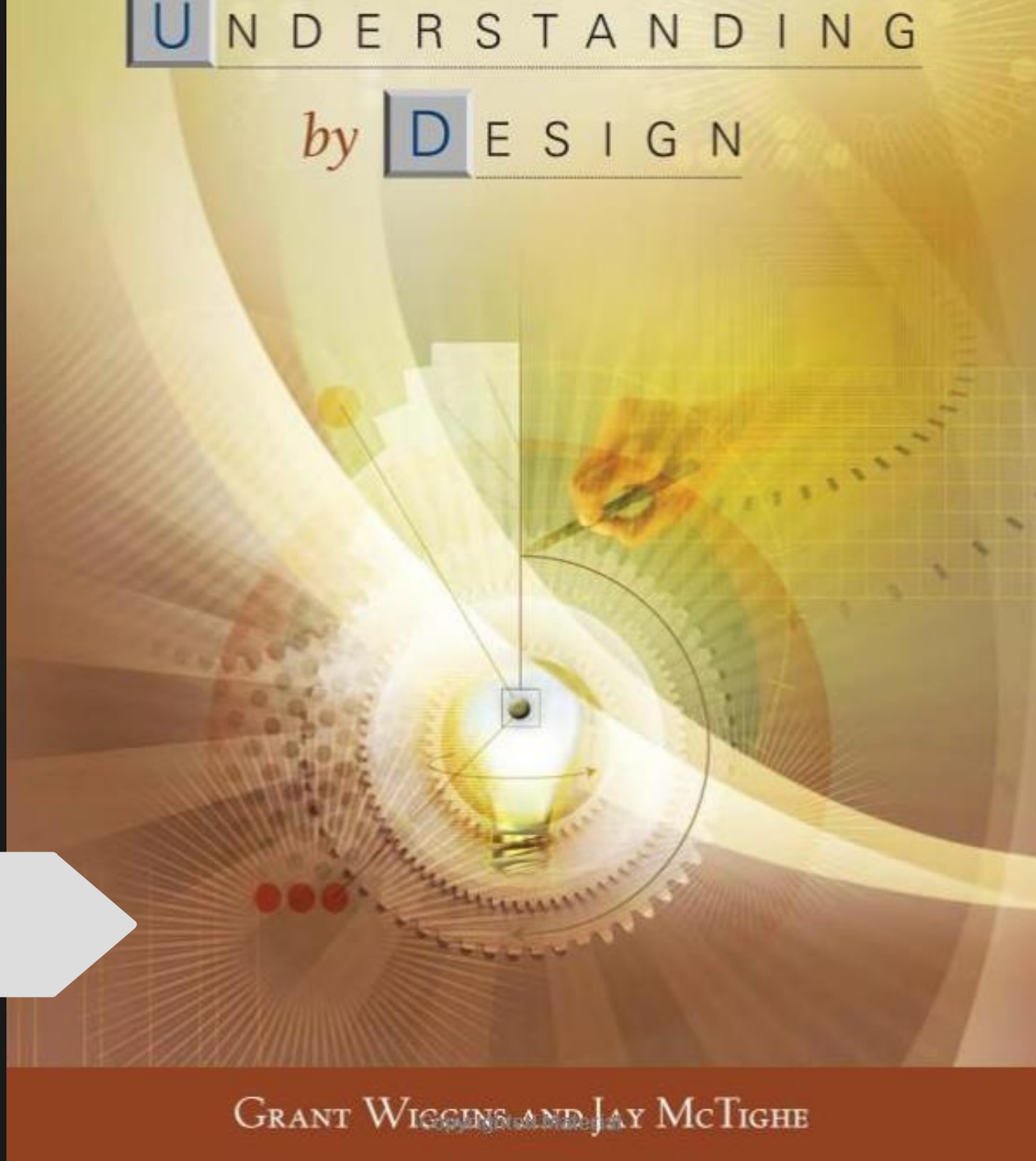
Example: Storyboard questions

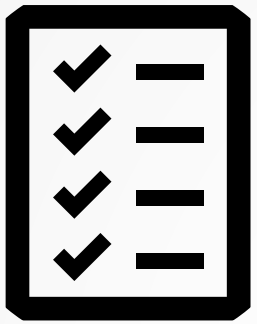
- What sprite(s) will you use as superhero(ines)?
  - What kind of superpowers or technology will they have?
  - Will they transform into their superhero(ine) costume or always be a superhero(ine)?
    - If they are transforming, what will they look like normally? What will they look like when they are a superhero(ine)?
- Who will the superhero(ines) try and save?
  - What kind of danger are they in?
  - If it's another sprite, what kind of powers or technology will they use?
- How might your superhero(ine) save the day?
  - What algorithms can you create to do that?
- Will users be able to interact with your superhero(ine) project?
  - If so, what kind of code will you use to create that interaction?

1. Choose a worthy topic
2. Find a real-life context
3. Create generative questions
4. Develop critical thinking and cultivate dispositions
5. Decide the scope
6. Design the experience

Tobias, E. S., Campbell, M. R., & Greco, P. (2015). Bringing Curriculum to Life: Enacting Project-Based Learning in Music Programs. *Music Educators Journal*, 102(2), 39–47

Backward design projects





Fixed Flexible Open

Backward design

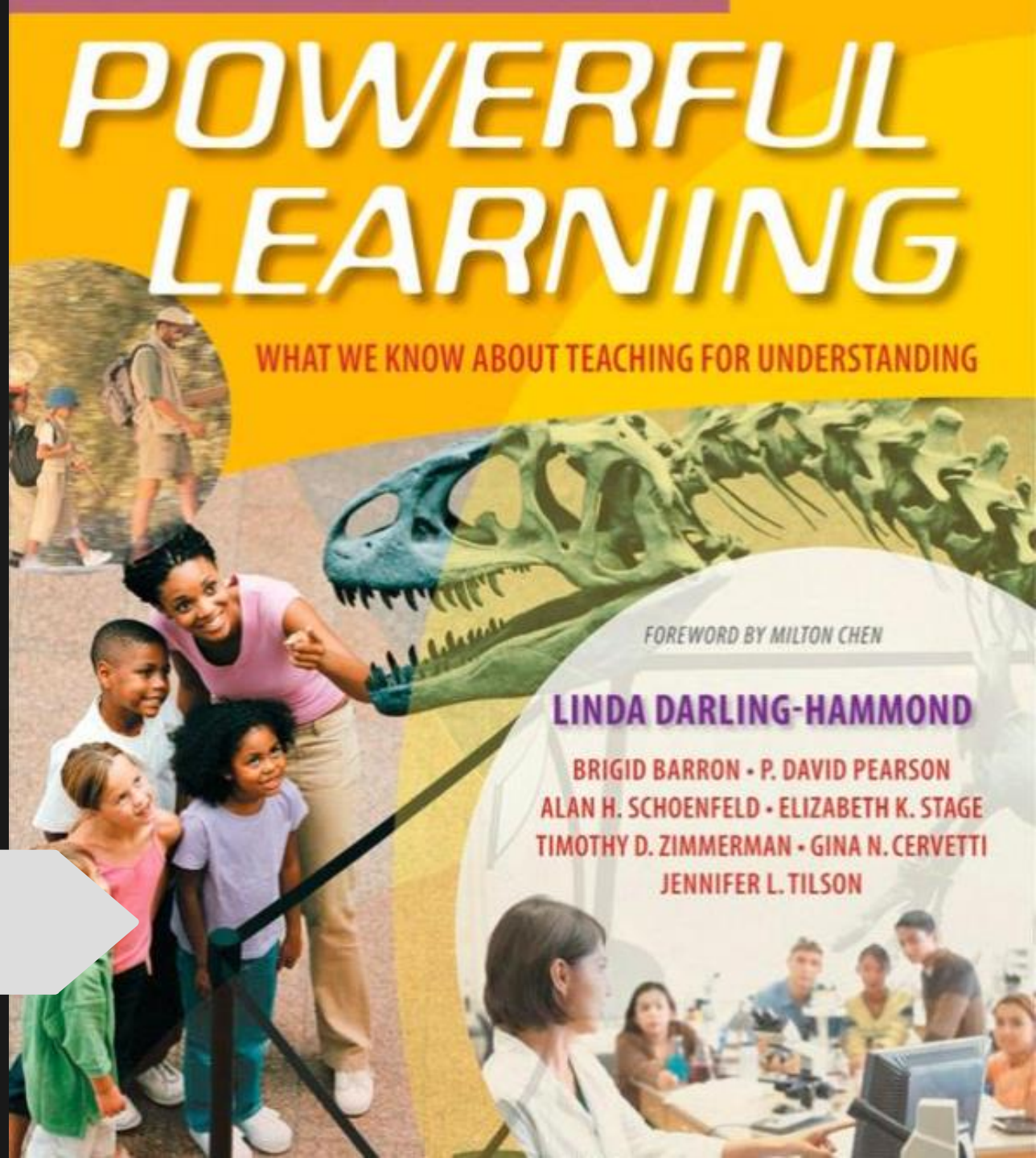
1. Identify the desired results
    a. Big ideas
    b. Enduring understandings
    c. Essential questions
2. Determine evidence
3. Plan learning experiences

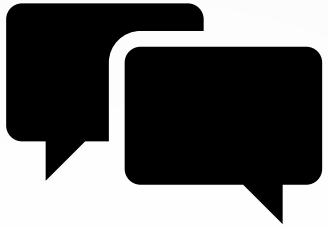
National Core Arts Standards

tworks for various contexts.
forming choices for media artworks? How can presenting or sharing media artworks in a public format help a media artist learn and grow?

3rd (MA:Pr6.1.3)	4th (MA:Pr6.1.4)	5th (MA:Pr6.1.5)	6th (MA:Pr6.1.6)	7th (MA:Pr6.1.7)	8th (MA:Pr6.1.8)	HS Proficient (MA:Pr6.1.I)
a. Identify and describe the presentation conditions, and take on roles and processes in presenting or distributing media artworks.	a. Explain the presentation conditions, and fulfill a role and processes in presenting or distributing media artworks.	a. Compare qualities and purposes of presentation formats, and fulfill a role and associated processes in presentation and/or distribution of media artworks.	a. Analyze various presentation formats and fulfill various tasks and defined processes in the presentation and/or distribution of media artworks.	a. Evaluate various presentation formats in order to fulfill various tasks and defined processes in the presentation and/or distribution of media artworks.	a. Design the presentation and distribution of media artworks through multiple formats and/or contexts.	a. Design the presentation and distribution of collections of media artworks, considering combinations of artworks, formats, and audiences.
experience, and share results of and improvements for presenting media artworks.	b. Explain results of and improvements for presenting media artworks.	b. Compare results of and improvements for presenting media artworks.	b. Analyze results of and improvements for presenting media artworks.	b. Evaluate the results of and improvements for presenting media artworks, considering impacts on personal growth.	b. Evaluate the results of and implement improvements for presenting media artworks, considering impacts on personal growth and external effects.	b. Evaluate and implement improvements in presenting media artworks, considering personal and local impacts, such as the benefits for self and others.

Inquiry-based projects



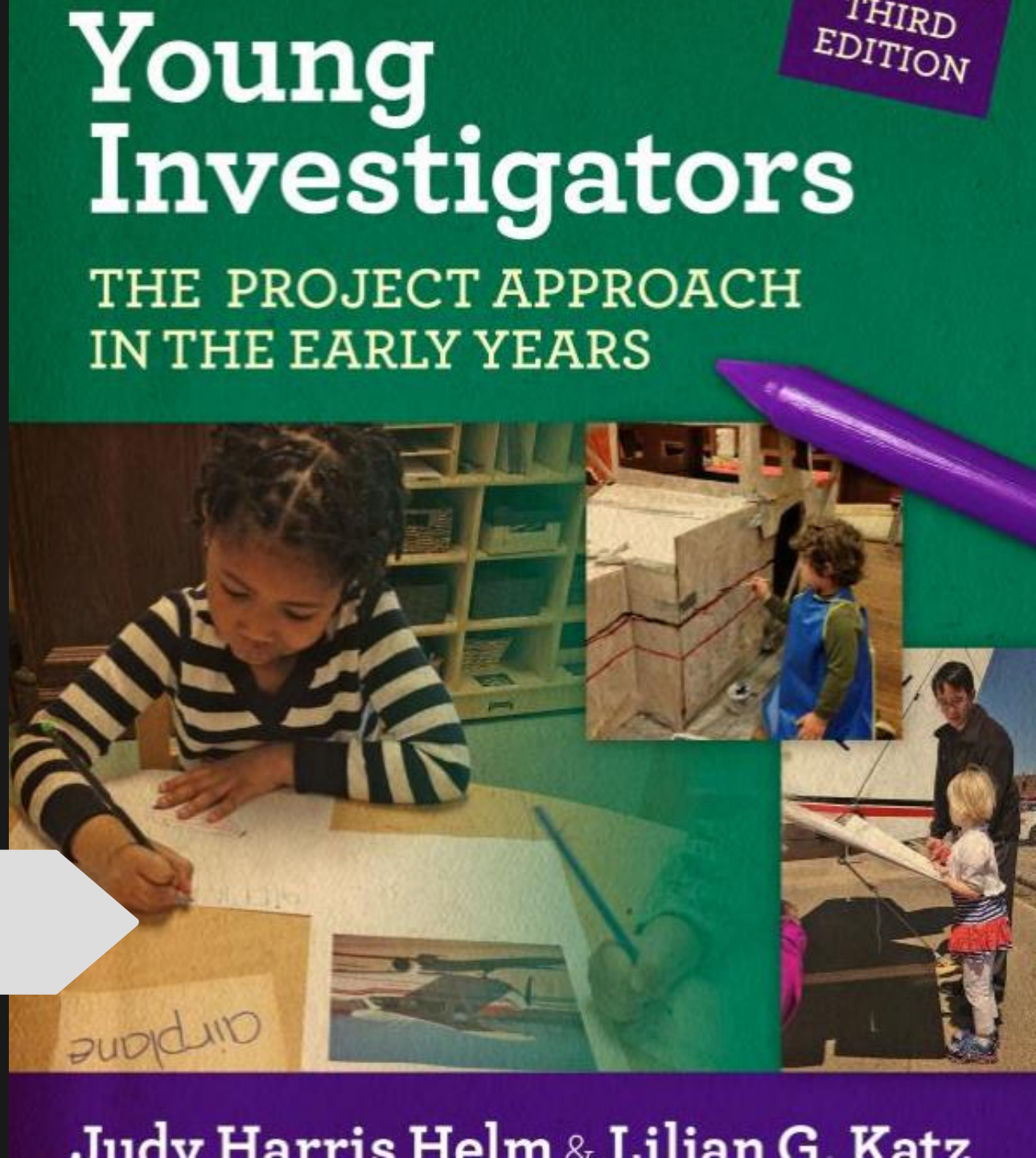


Fixed Flexible Open

Inquiry-based project stages

1. Vision
2. Inquiry
3. Build
4. Showtime
5. Transition

Emergent projects



Fixed

Flexible

Open

The project approach phases

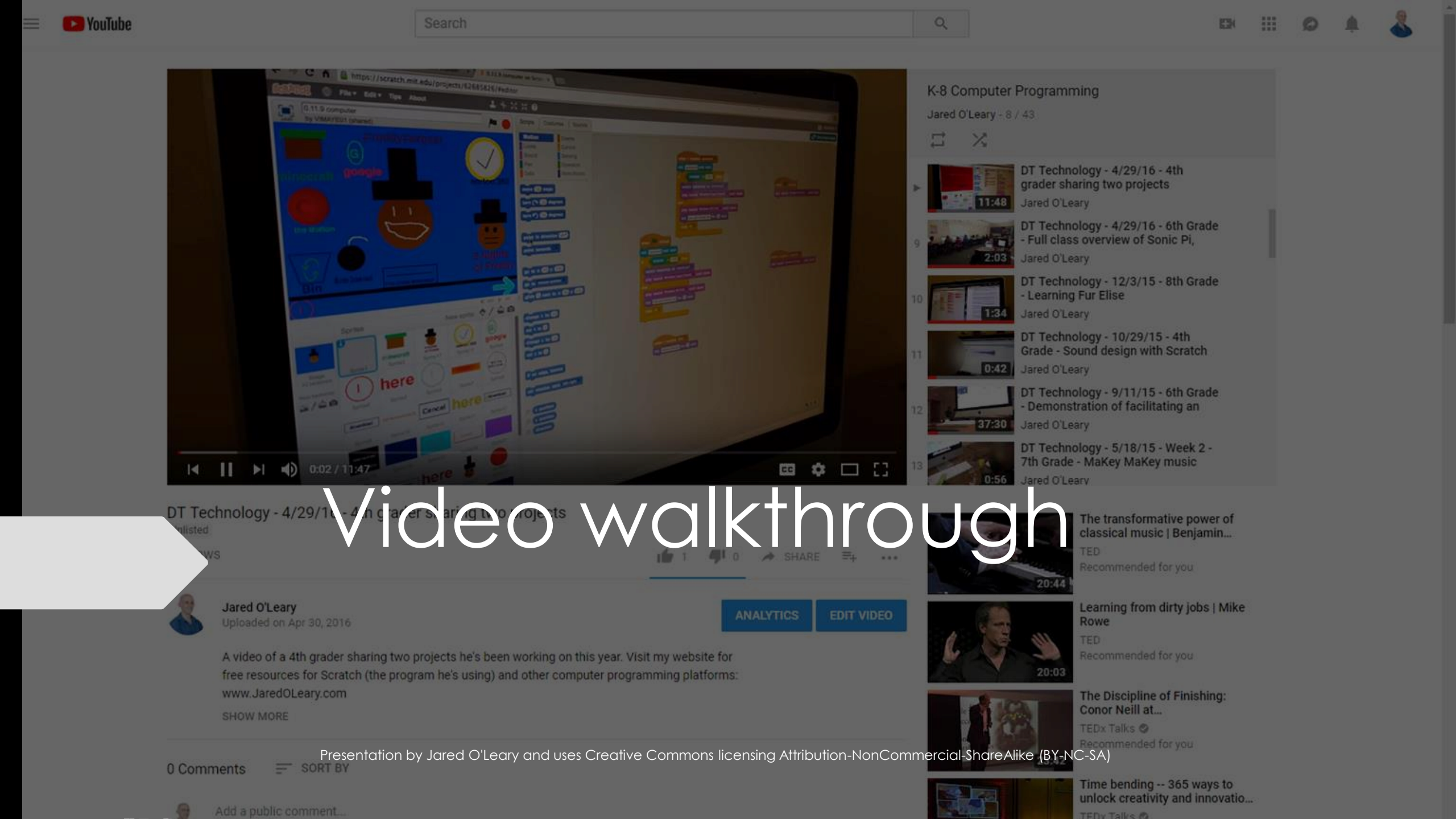
1. Determine a topic
2. Plan and investigate the topic
3. Culminating event/activities and assessment

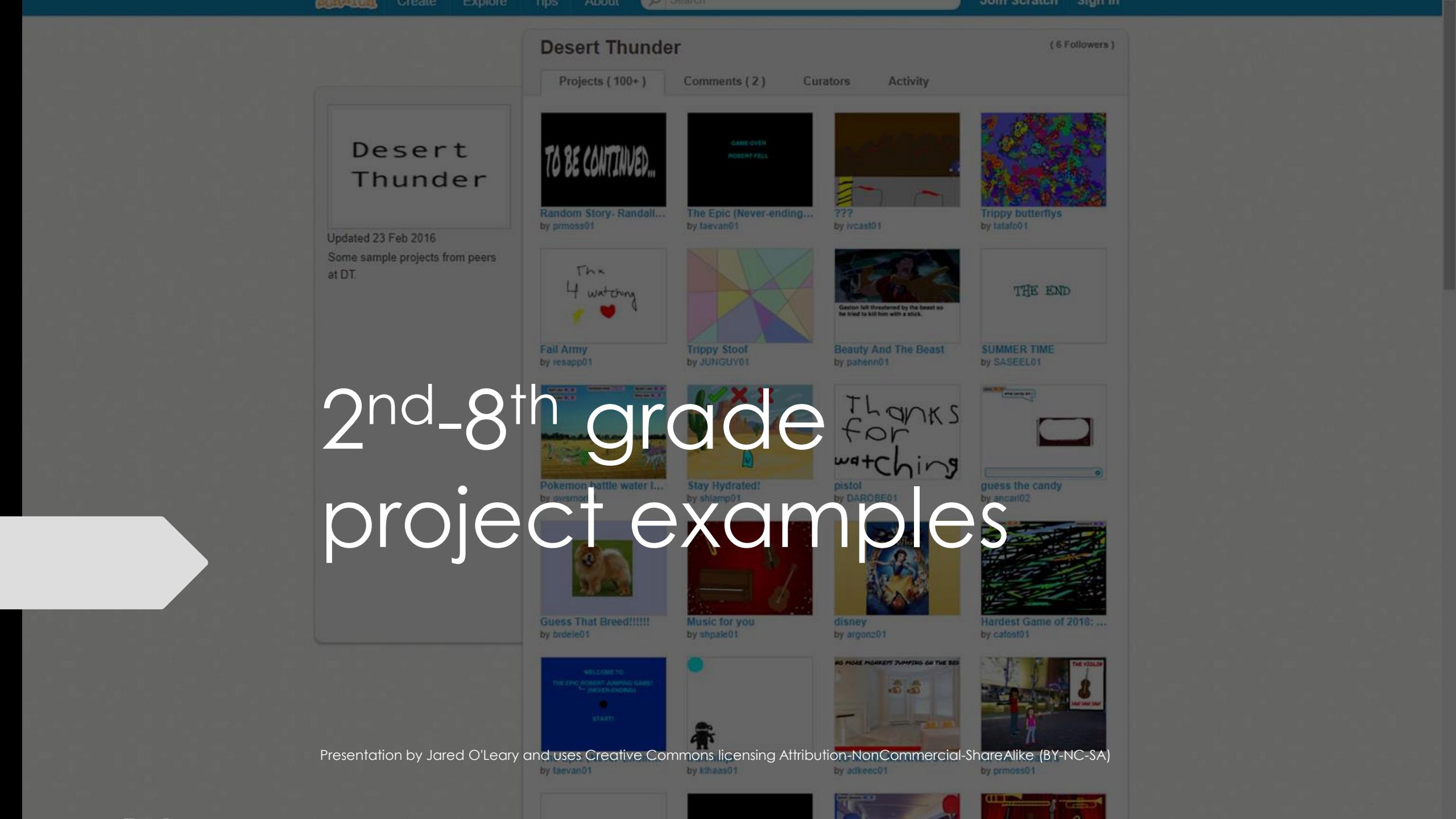
If using a sequential curriculum . . .

- Create a base project idea or theme
- Layer in new concepts and understandings
- Revisit throughout the year

0.11.9 computer

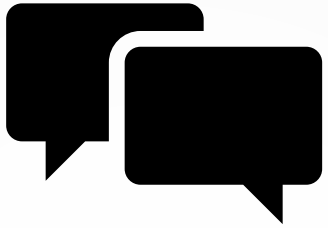
Video walkthrough





2nd-8th grade project examples

Explore Scratch projects



Fixed Flexible Open

Coder Resources

An Amazing Maze Game

Coder Resources

Project Sequence

(complete each step before moving to the next)

1. Sign in and create a new project
2. Create levels
    a. **Additional resources:**
        i. **Video:** Image editor: Bitmap mode (5:16)
        ii. **Video:** Image editor: Vector mode (5:00)
3. Create player controls
4. Create a restart function
5. Detect the walls
6. Create a goooooaaaaalllllllll
7. **Have some friends play test your game and give you feedback**
    a. Make some adjustments based on the feedback
8. Add in comments

Project Extensions

(pick and choose extensions that sound interesting)

1. Create a roguelike challenge
2. Add variables (Advanced)
3. Clean up your code with functions
4. Share your project
5. Create a thumbnail
6. Learn even more Scratch tips
7. Learn how to use a micro:bit with Scratch

Debugging Exercises

(practice your debugging skills by solving these bugs)

1. Why don't we switch to the next level when we touch the goal (the green rectangle)?
2. Why does Scratch Cat move to the right instead of the left when we press the left arrow?
3. Why do we stay on level 1 even when we reach the goal?
4. ***micro:bit required*** Why doesn't the Player sprite move when I tilt the micro:bit?
5. Even more debugging exercises

Scratch Projects

BootUp Professional Development™

WHO WE ARE | WHAT WE DO | CURRICULUM | NEWS & BLOG

BootUp Curriculum

Scratch (Grades 3+)

In this introductory sequence of projects for Scratch, we gradually introduce a variety of practices and concepts while simultaneously introd
a variety of blocks and tools in Scratch. Each of the projects is aligned with the algorithms and programming standards developed by the Co
Teachers Association (CSTA). Each project may take several classes to complete. Scratch (Grades 3+) Overview Video for projects #1-#10 (1:4
#20 (1:48), and projects #21-#30 (1:44).

#1 Animate Your Name

Minimum Experience:
Grades 3+, 1st year using Scratch, 1st quarter or

#2 Interactive Collage

Minimum Experience:
Grades 3+, 1st year using Scratch, 1st quarter or

#3 Jump Scare Slideshow

Minimum Experience:
Grades 3+, 1st year using Scratch, 1st qu

Coder Resources

Pumpkin Carver

Minimum Experience:
3+, 1st year using Scratch, 3rd quarter or

Overview & Purpose:
create a pumpkin carver simulator that users to "carve" a pumpkin with their . The purpose of this project is to introduce a drawing application using pen blocks bining them with previous understandings.

SON PLAN | CODER RESOURCES

#23 What Can You Create? Drawing

Minimum Experience:
Grades 3+, 1st year using Scratch, 3rd quarter or later

Overview & Purpose:
This challenge asks coders to use a limited selection of block types within an unlimited number of sprites to create art. The purpose of this challenge is to encourage coders to think creatively about block combinations to better understand algorithmic sequences.

LESSON PLAN | CODER RESOURCES

#24 Carve a Pumpkin with Code

Minimum Experience:
Grades 3+, 1st year using Scratch, 3rd quarter or later

Overview & Purpose:
Coders continue to develop their understanding pen blocks by creating algorithms to carve pumpkins. This purpose of this project is to reinforce understandings of how to draw shapes with code.

LESSON PLAN | CODER RESOURCES

Music Player

Minimum Experience:
3+, 1st year using Scratch, 3rd quarter or

se:
previous understandings of buttons to create a music tiple buttons. The purpose of this is to reinforce understandings of arity by combining previous understandings a new context.

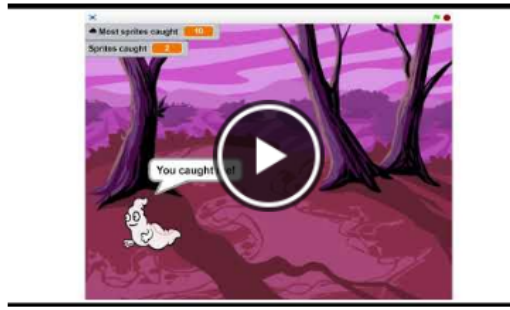
SON PLAN | CODER RESOURCES

#26 Blinking Maze Game

Minimum Experience:
Grades 3+, 1st year using Scratch, 3rd quarter or later

Overview & Purpose:
Coders create a player controlled blinking maze game with multiple, custom levels. The purpose of this project is to reinforce understandings of the previous maze game, while introducing new mechanics.

LESSON PLAN | CODER RESOURCES



#27 Sprite Catcher

Minimum Experience:
Grades 3+, 1st year using Scratch, 3rd quarter or later

Overview & Purpose:
Coders combine their understandings from previous projects to create a sprite catcher game The purpose of this project is to reinforce understandings of modularity in a new context.

LESSON PLAN | CODER RESOURCES

Follow the steps

Click here to learn about Scratch 3.0

BootUp Professional Development™

Beatbox Machine

Coder Resources

Project Sequence

(complete each step before moving to the next)

1. Sign in and create a new project
2. Create funny backdrops
3. Trigger sounds
4. Add in comments

Project Extensions

(pick and choose extensions that sound interestin

1. Fix a bug

1 – Preview projects

2 - Click “Coder Resources”

3 – Follow the steps

goo.gl/MKn7Uz

(case sensitive)

Lesson Plans

#22 Pumpkin Carver

Minimum Experience:
Grades 3+, 1st year using Scratch, 3rd quarter or later

Overview & Purpose:
Coders create a pumpkin carver simulator that allows users to "carve" a pumpkin with their mouse. The purpose of this project is to introduce creating a drawing application using pen blocks by combining them with previous understandings.

LESSON PLAN | CODER RESOURCES

#23 What Can You Create? Drawing

Minimum Experience:
Grades 3+, 1st year using Scratch, 3rd quarter or later

Overview & Purpose:
This challenge asks coders to use a limited selection of block types within an unlimited number of sprites to create art. The purpose of this challenge is to encourage coders to think creatively about block combinations to better understand algorithmic sequences.

LESSON PLAN | CODER RESOURCES

#24 Carve a Pumpkin with Co

Minimum Experience:
Grades 3+, 1st year using Scratch, 3rd qu later

Overview & Purpose:
Coders continue to develop their unders pen blocks by creating algorithms to car pumpkins. This purpose of this project is reinforce understandings of how to draw with code.

LESSON PLAN | CODER RESOURCE

#25 Music Player

Minimum Experience:
Grades 3+, 1st year using Scratch, 3rd quarter or

& Purpose:
ne their previous understandings of teractive buttons to create a music ith multiple buttons. The purpose of this project is to reinforce understandings of modularity by combining previous understandings within a new context.

LESSON PLAN | CODER RESOURCES

#26 Blinking Maze Game

Minimum Experience:
Grades 3+, 1st year using Scratch, 3rd quarter or later

Overview & Purpose:
Coders create a player controlled blinking maze game with multiple, custom levels. The purpose of this project is to reinforce understandings of the previous maze game, while introducing new mechanics.

LESSON PLAN | CODER RESOURCES

#27 Sprite Catcher

Minimum Experience:
Grades 3+, 1st year using Scratch, 3rd qu later

Overview & Purpose:
Coders combine their understandings fro previous projects to create a sprite catch The purpose of this project is to reinforc understandings of modularity in a new c

LESSON PLAN | CODER RESOURCE

ScratchJr

BootUp Professional Development™

WHO WE ARE | WHAT WE DO | CURRICULUM | NEWS & BLOG

BootUp Curriculum

ScratchJr (Grades K-2)

In this introductory sequence of projects for ScratchJr, we gradually introduce a variety of practices and concepts while simultaneously intro coders to a variety of blocks and tools in ScratchJr. Each of the projects is aligned with the algorithms and programming standards develope the Computer Science Teachers Association (CSTA). Each project may take several classes to complete. ScratchJr (Grades K-2) Overview Video #1-#10 (1:18), projects #11-#20 (1:32), and projects #21-#30 (1:35).

#1 Dancing Alone

Minimum Experience:
Grades K+, 1st year using Scratch Jr., 1st quarter or

#2 Can't Stop Dancing

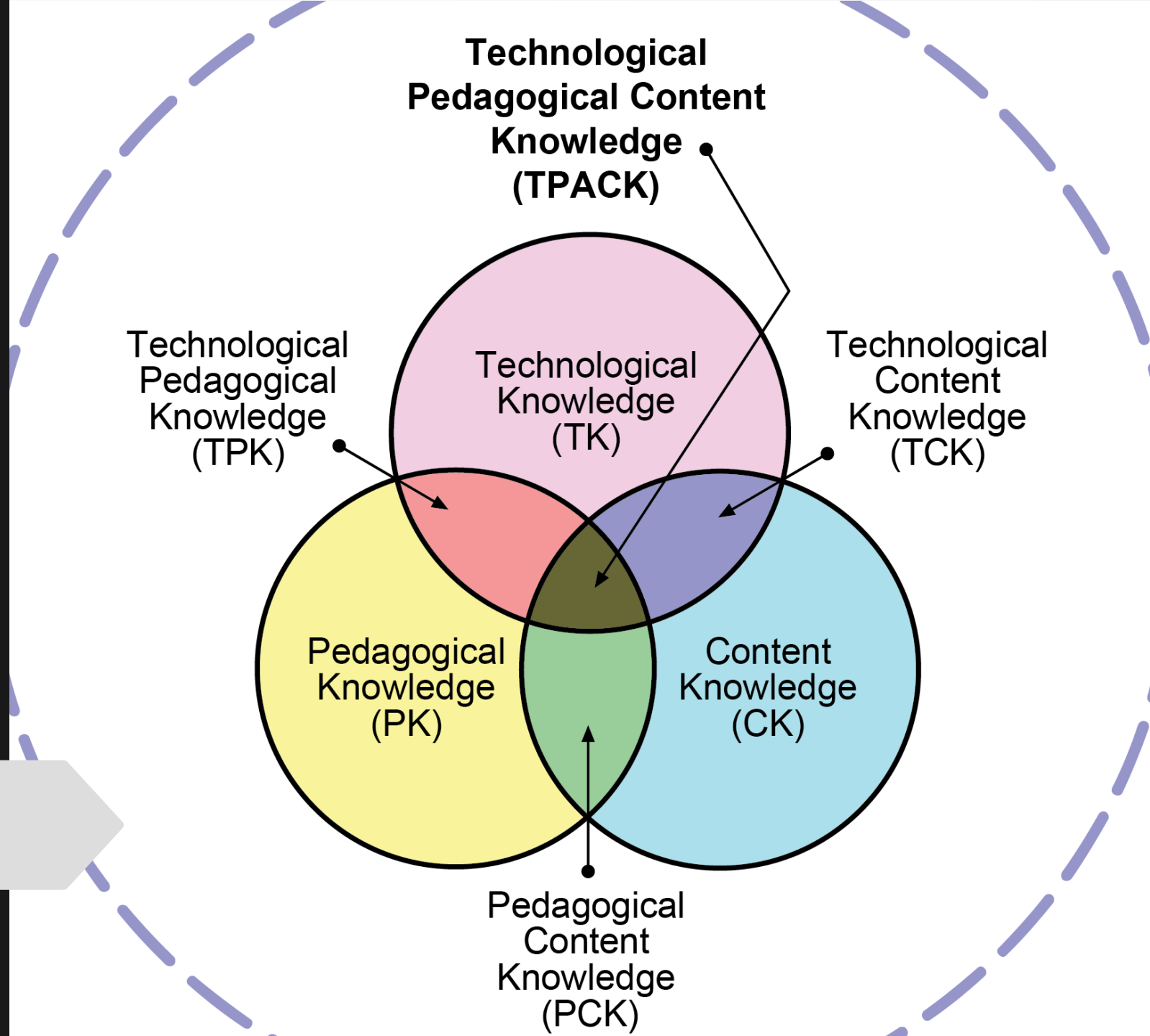
Minimum Experience:
Grades K+, 1st year using Scratch Jr., 1st quarter or

#3 Dance Party

Minimum Experience:
Grades K+, 1st year using Scratch Jr., 1st

Discussion

TPACK



ISTE Standards

ISTE STANDARDS FOR COMPUTER SCIENCE

In an increasingly digital world, computer science plays a star role. The ISTE Standards·CSE describe what computer science teachers must kno effectively integrate these essential concepts.

EXPLORE COMPUTER SCIENCE EDUCATOR STAN

1 Knowledge of content	Computer science educators demonstrate knowledge of computer science co important principles and concepts.
2 Effective teaching and learning strategies	Computer science educators demonstrate effective content pedagogical strat the discipline comprehensible to students.
3 Effective learning environments	Computer science educators apply their knowledge of learning environments maintaining safe, ethical, supportive, fair and effective learning environments
Effective professional knowledge and skills	Computer science educators demonstrate professional knowledge and skills i readiness to apply them.

ISTE Standards for

Computer Science

DOWNLOAD

EXPLORE COMPUTER SCIENCE EDUCATOR STANDARDS

1	Knowledge of content	Computer science educators demonstrate knowledge of computer science content and model important principles and concepts.
2	Effective teaching and learning strategies	Computer science educators demonstrate effective content pedagogical strategies that make the discipline comprehensible to students.
3	Effective learning environments	Computer science educators apply their knowledge of learning environments by creating and maintaining safe, ethical, supportive, fair and effective learning environments for all students.
4	Effective professional knowledge and skills	Computer science educators demonstrate professional knowledge and skills in their field and readiness to apply them.

EXPLORE COMPUTER SCIENCE EDUCATOR STANDARDS

1	Knowledge of content	Computer science educators demonstrate knowledge of computer science content and model important principles and concepts. +
2	Effective teaching and learning strategies	Computer science educators demonstrate effective content pedagogical strategies that make the discipline comprehensible to students. +
3	Effective learning environments	Computer science educators apply their knowledge of learning environments by creating and maintaining safe, ethical, supportive, fair and effective learning environments for all students. +
4	Effective professional knowledge and skills	Computer science educators demonstrate professional knowledge and skills in their field and readiness to apply them. +

EXPLORE COMPUTER SCIENCE EDUCATOR STANDARDS

1	Knowledge of content	Computer science educators demonstrate knowledge of computer science content and model important principles and concepts.
2	Effective teaching and learning strategies	Computer science educators demonstrate effective content pedagogical strategies that make the discipline comprehensible to students.
3	Effective learning environments	Computer science educators apply their knowledge of learning environments by creating and maintaining safe, ethical, supportive, fair and effective learning environments for all students.
4	Effective professional knowledge and skills	Computer science educators demonstrate professional knowledge and skills in their field and readiness to apply them.

EXPLORE COMPUTER SCIENCE EDUCATOR STANDARDS

1	Knowledge of content	Computer science educators demonstrate knowledge of computer science content and model important principles and concepts.
2	Effective teaching and learning strategies	Computer science educators demonstrate effective content pedagogical strategies that make the discipline comprehensible to students.
3	Effective learning environments	Computer science educators apply their knowledge of learning environments by creating and maintaining safe, ethical, supportive, fair and effective learning environments for all students.
4	Effective professional knowledge and skills	Computer science educators demonstrate professional knowledge and skills in their field and readiness to apply them.

CT COMPETENCIES

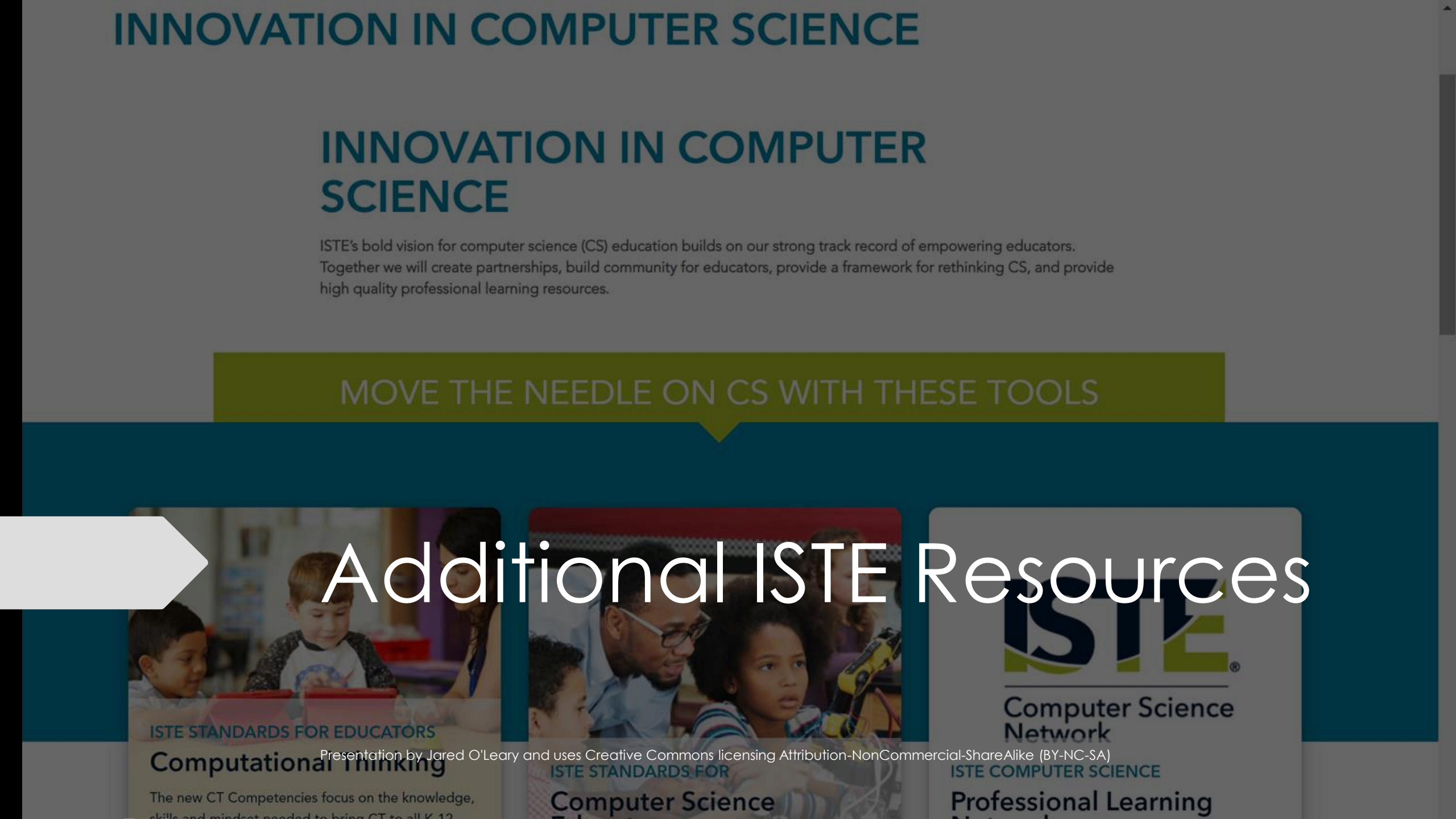
Read how this body of work complements the existing CSTA K-12 Computer Science Standards for Students and the K-12 Computer Science Framework, and why ISTE created the CT Competencies.

1. Computational Thinking (Learner) +
2. Equity Leader (Leader) +
3. Collaborating Around Computing (Collaborator) +
4. Creativity & Design (Designer) +
5. Integrating Computational Thinking (Facilitator) +

EXPLORE THE STUDENT STANDARDS

1	Empowered Learner	Students leverage technology to take an active role in choosing, achieving and demonstrating competency in their learning goals, informed by the learning sciences.
2	Digital Citizen	Students recognize the rights, responsibilities and opportunities of living, learning and working in an interconnected digital world, and they act and model in ways that are safe, legal and ethical.
3	Knowledge Constructor	Students critically curate a variety of resources using digital tools to construct knowledge, produce creative artifacts and make meaningful learning experiences for themselves and others.
4	Innovative Designer	Students use a variety of technologies within a design process to identify and solve problems by creating new, useful or imaginative solutions.
5	Computational Thinker	Students develop and employ strategies for understanding and solving problems in ways that leverage the power of technological methods to develop and test solutions.
6	Creative Communicator	Students communicate clearly and express themselves creatively for a variety of purposes using the platforms, tools, styles, formats and digital media appropriate to their goals.
7	Global Collaborator	Students use digital tools to broaden their perspectives and enrich their learning by collaborating with others and working effectively in teams locally and globally.

Additional ISTE Resources



Session Evaluation

Please take a moment to evaluate this session. Your valuable feedback helps make the overall program stronger and ensures we're meeting your learning needs. Evaluations are also used by the conference program committee to provide feedback to presenters and inform future presentations.

To provide feedback and rate the quality of this session, please use the ISTE19 app or locate the session online using the program search at isteconference.org.

Thank you!

Q&A

- www.JaredOLeary.com
  - Presentations
    - Project-based Learning with Scratch (ISTE)