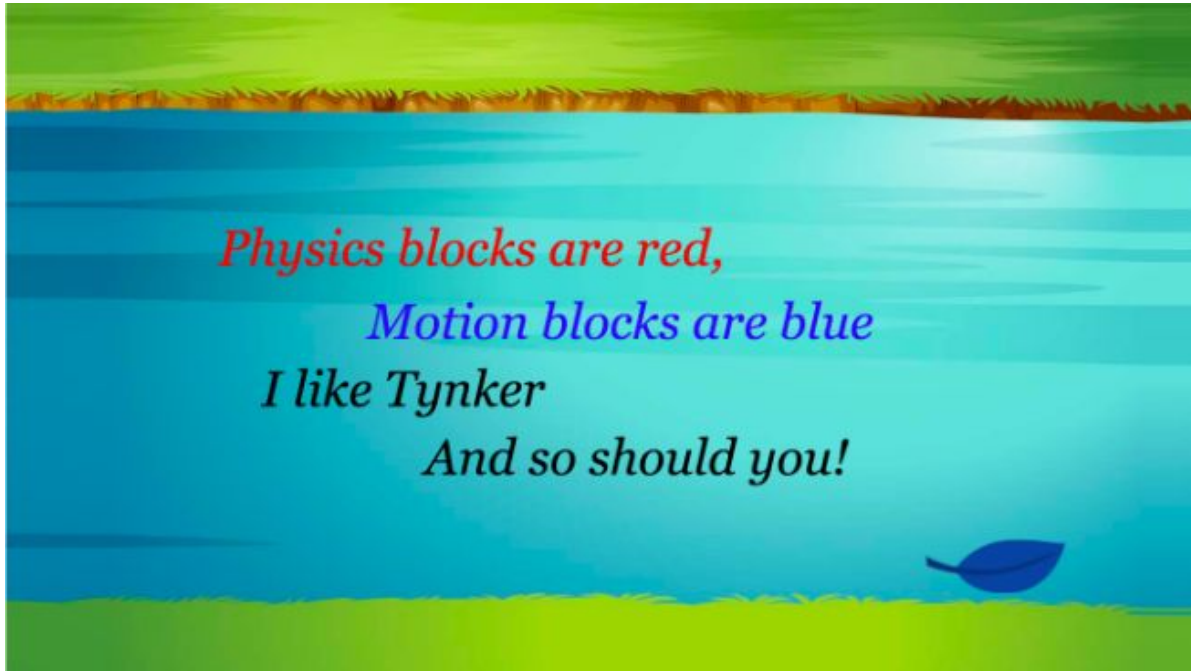


Code a Poem Teacher Guide



Summary

- | | |
|----------------------------|---|
| ● Coding skill level: | Intermediate |
| ● Recommended grade level: | Grades 3-8 (U.S.), Years 4-9 (U.K) |
| ● Time required: | 40 minutes |
| ● Number of modules: | 1 module |
| ● Coding Language: | Block-based |

Teacher Guide Outline

Welcome!

- How to Prepare

Activity

- Overview
- Getting Started (10 minutes)
- DIY Module (30 minutes)
- Extended Activities

Going Beyond Code a Poem

- Do More With Tynker
- Tynker for Schools

Help

Welcome!

Let's celebrate World Poetry Day (March 21) by reading, creating, and sharing poetry! In this lesson, students will tap into their creative writing skills as they create their own poem. After students finish writing their poem, they will follow a step-by-step tutorial to bring their poem to life using Tynker. Activities include selecting a background that matches the theme of their poem; using the text tool to write their poem; programming the starting position of the first line of their poem; programming the lines of their poem to glide in or drop from above; and programming an Actor to gently float down the screen and change colors. For an additional challenge, students can complete the bonus activities in "Step 8" of the tutorial.

By the end of this lesson, students will have combined creativity with problem-solving skills while reinforcing coding concepts such as simple events, delays, screen bounds, advanced costume handling, advanced motion, and advanced animation.

How to Prepare

This activity is designed for self-directed learning. Your role will be to help students individually and facilitate as students complete the coding activities on their own. The best way to prepare is to:

1. **Familiarize yourself with the material.** After selecting your Tynker lesson (e.g., Code a Poem), read through this teacher guide and complete the activity before assigning it to students. This will allow you to troubleshoot anything in advance and plan for potential questions from your students.
2. **OPTIONAL: Sign up for a teacher account.** Although an account is NOT required, creating a free teacher account will allow you to access teacher guides, answer keys, and tons of additional resources. You'll also be able to create free accounts for your students, monitor their progress, and see their projects.
3. **OPTIONAL: Create student accounts.** From your teacher account, you can easily create free student accounts for all your students. This will allow them to save their projects and progress, so they can continue coding when they get home! Again, this is not necessary to complete the Code a Poem lesson.

Activity

Overview

Objectives

Students will...

- Write an original poem
- Apply coding concepts such as simple events, delays, screen bounds, advanced costume handling, advanced motion, and advanced animation to create an interactive poem

Materials

- **For web:** Computers, laptops, or Chromebooks (1 per student)
- **For mobile:** iPads or Android tablets (1 per student)

Vocabulary

- **Code:** The language that tells a computer what to do
- **Actor:** A Tynker character or object that can talk and interact with others
- **Stage:** The background of the project where the Actors are placed
- **Sequence:** The order in which steps or events happen
- **Command:** A specific action or instruction that tells the computer to do something

U.S. Standards

- **CCSS-ELA:** RI.3.7, RF.3.4, RF.3.4.A, SL.3.1, RF.4.4.A, RF.1.4.A, RF.4.4, SL.4.1, RF.5.4.A, RF.5.4, SL.5.1, RI.6.4, RI.6.7, SL.6.1, SL.7.1, SL.8.1
- **CCSS-Math:** MP.1
- **CSTA:** 1B-AP-11, 1B-AP-12, 1B-AP-14, 1B-AP-15, 2-AP-13, 2-AP-16, 2-AP-17
- **CS CA:** 3-5.AP.10, 3-5.AP.13, 3-5.AP.14, 3-5.AP.16, 3-5.AP.17, 6-8.AP.13, 6-8.AP.16
- **ISTE:** 1.c, 1.d, 4.d, 5.c, 5.d, 6.b

U.K. Standards

National Curriculum in England (computing):

- **Key Stage 2 (Years 4-6)**
 - Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts
 - Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs
 - Understand computer networks, including the internet; how they can provide multiple services, such as the World Wide Web, and the opportunities they offer for communication and collaboration
 - Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact
- **Key Stage 3 (Years 7-9)**
 - Create, reuse, revise and repurpose digital artefacts for a given audience, with attention to trustworthiness, design and usability
 - Understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct, and know how to report concerns

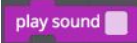

Getting Started (10 minutes)

1. Tell students that they are going to create their own animated poem using Tynker! *Optional:* Inspire students by reading different poems that are age-appropriate. For example, younger students will likely enjoy poems by Shel Silverstein, while older students may like Lewis Carroll’s “Jabberwocky.”
2. As a class, create a poem and write it on the board. It can be a poem about school spirit, coding, spring, or any other topic that inspires you!
3. Next, ask students to create their own poem and write it on a piece of paper. If they’re struggling to get started, ask them to create a specific type of poem (e.g., acrostic, limerick, haiku) and provide them category options about what to write about (e.g., weather, their favorite sport, animals, pets, nature).

DIY Module (30 minutes)

This lesson has one DIY (do-it-yourself) module. Facilitate as students complete the Code a Poem module on their own:

Code a Poem (DIY)

- In this DIY project, students will create an interactive poem.
- Tell students to follow the step-by-step instructions and drag code blocks from the tutorial tab to the center coding area.
- Point out to students that they’ll need to create one Actor for each line of their poem, as explained in “Step 3” of the tutorial.
- Do students want to modify their project? Direct their attention to “Step 8” of the tutorial, which include bonus activities. Here are some hints to help your students get started:
 - **Add more Actors-** Tell students to draw their own Actors or select Actors from Tynker’s Media Library.
 - **Add sound effects or background music-** Tell students to use the “play sound” code block. Here’s what it looks like:
 
 - **Add a second verse-** Tell students to use the “hide” code block to make the first verse disappear from the Stage. Here’s what it looks like:
 
- If students finish early, encourage them to create another poem that uses a different form structure (e.g., acrostic, limerick, haiku).

Extended Activities

Show and Tell

Encourage students to share their projects with the class:

- Use your projector to display their unique animated poems. Did students complete the bonus activities? How did they modify their code?

Discussion

Ask your students...

- What is something you learned today about poems?
 - Do you have a favorite poem? What is it about?
 - Who can describe how they experimented with their code? Discuss obstacles and successes.
-

Going Beyond Code a Poem

If your students enjoyed Code a Poem, they're sure to enjoy the rest of what Tynker has to offer! Tynker offers a complete premium solution for schools to teach computer science. Over 400 hours of lessons are available to take K-8 students from block coding to advanced text coding. We offer tons of resources for teachers, including comprehensive guides, free webinars, and a forum to connect with other educators.

Do More with Tynker

With Tynker, kids don't just acquire programming skills—they explore the world of possibilities that coding opens up. Tynker has several interest-driven learning paths that make coding fun, both inside and outside the classroom:

- **Coding and Game Design:** Your students can use Tynker Workshop, a powerful tool for crafting original programs to make games, stories, animations, and other projects. They can even share their work with other kids in the Tynker Community.
- **Drones and Robotics:** Tynker integrates with connected toys, including Parrot drones and Lego WeDo robotics kits, so kids can see their code come to life.
- **Minecraft:** Tynker integrates with Minecraft so your students can learn coding through a game they love. Tynker offers skin and texture editing, as well as a custom Mod Workshop that lets kids try their original code in Minecraft.

Tynker for Schools

Used in over 80,000 schools, our award-winning platform has flexible plans to meet your classroom, school, or district needs. All solutions include:

- Grade-specific courses that teach visual coding, JavaScript, Python, robotics and drones
- A library of NGSS and Common Core compliant STEM courses that are great for project-based learning
- Automatic assessment and mastery charts for whole schools and individual classes and students
- Easy classroom management with Google Classroom and Clever integration
- Professional training, free webinars and other teacher training resources

Need help getting Tynker started at your school? [Contact us](#) to learn more about teaching programming at your school with Tynker!

Help

Need help? Below you'll find answers to frequently asked questions about using Code a Poem.

How do I prepare for Code a Poem?

1. **Familiarize yourself with the material.** After selecting your Tynker lesson (e.g., Code a Poem), read through this teacher guide and complete the activity before assigning it to students. This will allow you to troubleshoot anything in advance and plan for potential questions from your students.
2. **OPTIONAL: Sign up for a teacher account.** Although an account is NOT required, creating a free teacher account will allow you to access teacher guides, answer keys, and tons of additional resources. You'll also be able to create free accounts for your students, monitor their progress, and see their projects.
3. **OPTIONAL: Create student accounts.** From your teacher account, you can easily create free student accounts for all your students. This will allow them to save their projects and progress, so they can continue coding when they get home! Again, this is not necessary to complete the Code a Poem lesson.

Who is this activity for?

Code a Poem is intended for students in grades 3-8 with some coding experience.

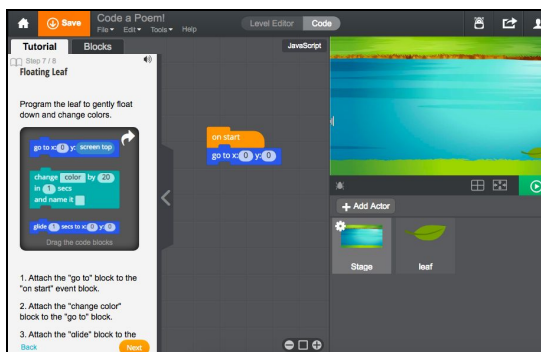
Do I need to create Tynker accounts for my students?

No, you do not need to create Tynker accounts for your students.

What devices do I need?

- **For web:** Computers, laptops, or Chromebooks (1 per student) with an internet connection
- **For mobile:** iPads or Android tablets (1 per student) with an internet connection

How do my students code their projects?



The Code a Poem DIY module includes a workspace for students to code their projects. The section on the left is a tutorial tab that gives step-by-step directions, describes what is happening in each step, and provides the necessary code blocks. Tell students to follow the step-by-step instructions and drag blocks from the tutorial tab to the center coding area.

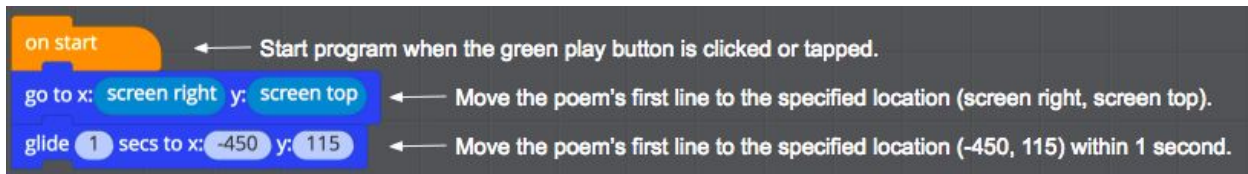
What will my students learn?

Students will combine creative writing, imagination, and coding concepts (i.e., simple events, delays, screen bounds, advanced costume handling, advanced motion, and advanced animation) to create an interactive poem! “Step 8” of the DIY tutorial includes a bonus section that encourages students to expand on their project while experimenting with new code blocks. In this process, students will develop debugging and logical reasoning skills.

What do the code blocks do?

This is an open-ended project, and below is an example (with noted pseudocode) of what your students’ code might look like:

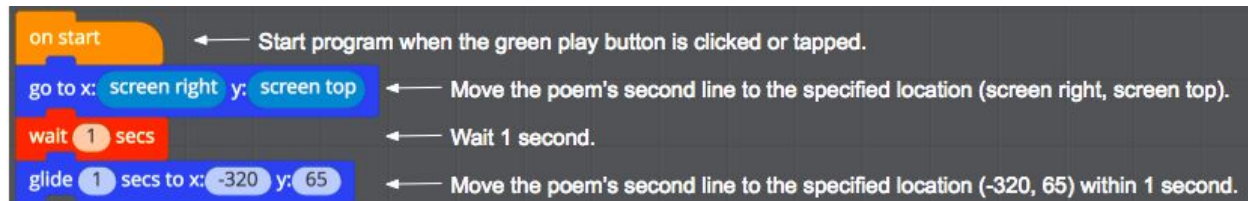
Poem line 1-



```

on start ← Start program when the green play button is clicked or tapped.
go to x: screen right y: screen top ← Move the poem's first line to the specified location (screen right, screen top).
glide 1 secs to x: -450 y: 115 ← Move the poem's first line to the specified location (-450, 115) within 1 second.
  
```

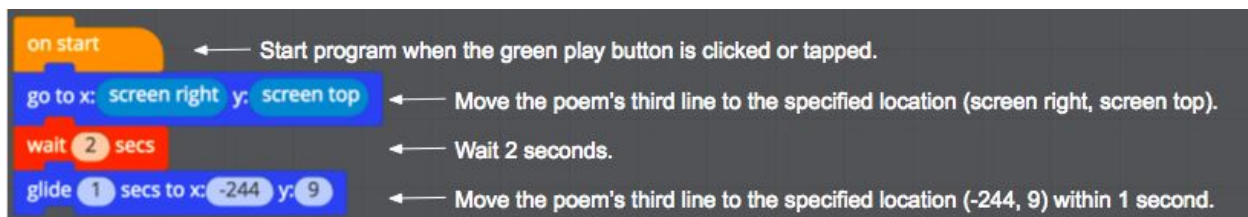
Poem line 2-



```

on start ← Start program when the green play button is clicked or tapped.
go to x: screen right y: screen top ← Move the poem's second line to the specified location (screen right, screen top).
wait 1 secs ← Wait 1 second.
glide 1 secs to x: -320 y: 65 ← Move the poem's second line to the specified location (-320, 65) within 1 second.
  
```

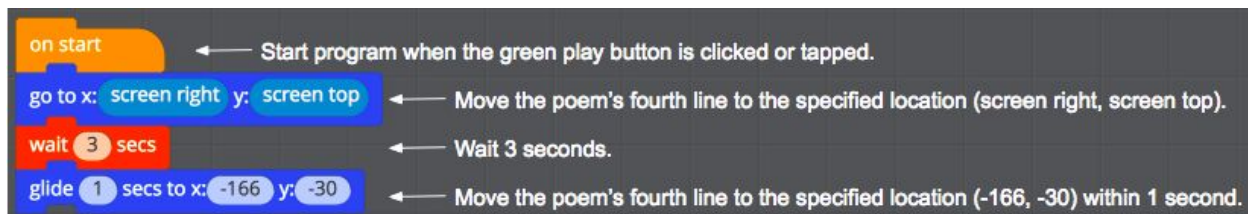
Poem line 3-



```

on start ← Start program when the green play button is clicked or tapped.
go to x: screen right y: screen top ← Move the poem's third line to the specified location (screen right, screen top).
wait 2 secs ← Wait 2 seconds.
glide 1 secs to x: -244 y: 9 ← Move the poem's third line to the specified location (-244, 9) within 1 second.
  
```

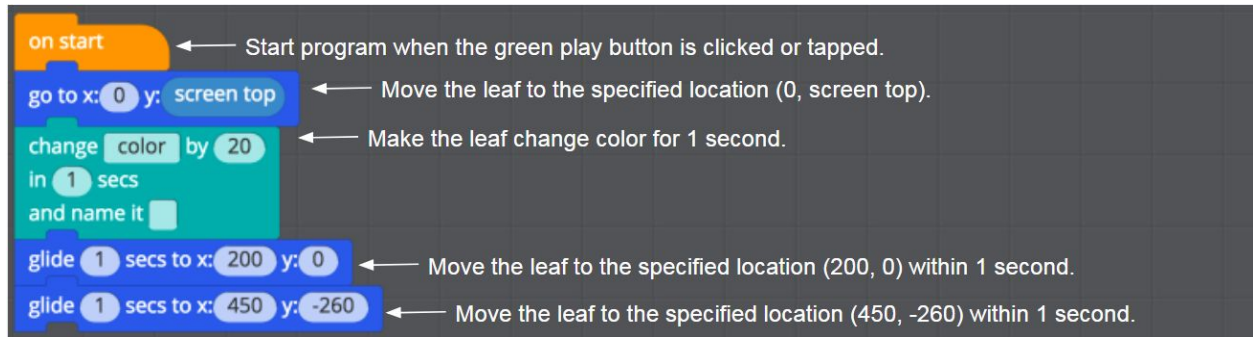
Poem line 4-



```

on start ← Start program when the green play button is clicked or tapped.
go to x: screen right y: screen top ← Move the poem's fourth line to the specified location (screen right, screen top).
wait 3 secs ← Wait 3 seconds.
glide 1 secs to x: -166 y: -30 ← Move the poem's fourth line to the specified location (-166, -30) within 1 second.
  
```

Leaf-



The image shows a sequence of Scratch code blocks on a dark grey grid background. Each block is annotated with a white arrow pointing to it and a corresponding text explanation. The blocks are: 1. An orange 'on start' block. 2. A blue 'go to x: 0 y: screen top' block. 3. A teal 'change color by 20 in 1 secs and name it' block. 4. A blue 'glide 1 secs to x: 200 y: 0' block. 5. A blue 'glide 1 secs to x: 450 y: -260' block.

```
on start  
go to x: 0 y: screen top  
change color by 20  
in 1 secs  
and name it  
glide 1 secs to x: 200 y: 0  
glide 1 secs to x: 450 y: -260
```

← Start program when the green play button is clicked or tapped.

← Move the leaf to the specified location (0, screen top).

← Make the leaf change color for 1 second.

← Move the leaf to the specified location (200, 0) within 1 second.

← Move the leaf to the specified location (450, -260) within 1 second.

How can I contact the Tynker support team?

If you have any issues or questions, send us an email at support@tynker.com.