Balancing Act
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: **Tynker blocks**

Teacher Guide Outline

Welcome!
- How to Prepare

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

Going Beyond Balancing Act
- Do More With Tynker
- Tynker for Schools

Help

Welcome!

In this lesson, students will use Tynker's AI (artificial intelligence) body tracking model to create an interactive game where the player balances a ball on a platform using their face. Let's see how long your students will last before the ball falls to the floor!

Students will begin with a teacher-led discussion to learn how the physics code blocks work. Next, students will move on to the DIY module where they'll reinforce coding concepts to program a game called "Balancing Act." The DIY module includes a sample project, step-by-step tutorial, code blocks to help students get started, and ideas on how to expand on their project. Additionally, students will review what they learned in today's coding adventure as they complete the "Extended Activities" section of this teacher guide. **Important:** Students will need a working webcam in order to complete the project.

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 (Balancing Act), read through the 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. **Sign up for a teacher account.** 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. **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!

Activity

Overview

Objectives

Students will...
- Use the Physics Engine to create a game
- Apply coding concepts such as video control, simple pose detection, physics, and simple loops
- Use code blocks to create an interactive game that uses Tynker's body tracking AI

Materials
- **For web:** Computers, laptops, or Chromebooks (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
- **Loop:** An action that repeats one or more commands over and over
- **Artificial Intelligence:** The idea of machines (such as computers) demonstrating intelligence, similar to how our brain works

U.S. Standards
- **CCSS-ELA:** RF.3.4.A, RF.4.4.A, RF.5.4.A, 6-8.RST.3, 6-8.RST.4, 6-8.RST.7
- **CCSS-Math:** MP.1
- **CSTA:** 1B-AP-11, 1B-AP-12, 1B-AP-15, 2-AP-13, 2-AP-17
- **CS CA:** 3-5.AP.13, 3-5.AP.17, 6-8.AP.13, 6-8.AP.15
- **ISTE**: 1.1.c, 1.1.d, 1.4.d, 1.5.c, 1.5.d, 1.6.b

U.K. Standards
National Curriculum in England (computing):
- **Key Stage 2 (Years 4-6)**
  - Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs
  - 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)
- Show students the completed sample project by clicking this image in the tutorial:
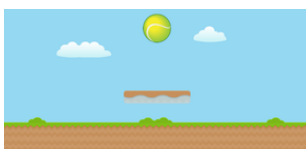
- Time permitting, explain how the "set active" and "set static" physics blocks work. For example, tell students that…
  - An active, static Actor will collide with other active Actors but will not fall to the ground. Check for understanding by asking students which Actor behaved this way. (Answer: the Platform Actor.)
  - An active, non-static Actor will collide with other active Actors but will fall to the ground. Check for understanding by asking students which Actor behaved this way. (Answer: the Ball Actor.)
  - If an Actor is active, the Actor is affected by physics.
  - An inactive Actor will not be affected by physics at all.

DIY Module (30 minutes)

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

Balancing Act (DIY)
- In this DIY, students will use Tynker's AI pose detection blocks and the Physics Engine to create an exciting game where the player balances a ball using their face.
- Note that this project utilizes an AI model. This model has already been "trained" to recognize the human body by being shown many photos of people and data about what's in the photos. Now the model is ready to be used in the project, where it can "find" your students' face using their webcam.
- Coding activities include setting up the Stage by enabling the camera, face detection, and the Physics Engine; adding a key-press event to start the game; choosing or drawing a platform Actor; programming the platform Actor to follow the player's head as it moves across the screen; choosing or drawing a ball Actor; programming the Ball Actor to respond to gravity.
- Make sure students follow the step-by-step instructions and drag blocks from the tutorial tab to the center coding area.
- Do students want to learn more about the physics code blocks and what amazing games/projects they can create with it? Direct them to this link:

  - [The Physics Blocks](https://www.tynker.com/blog/tynker-toolbox-the-physics-blocks/) - Explore what's possible with Tynker's physics blocks Link address:
    https://www.tynker.com/blog/tynker-toolbox-the-physics-blocks/

- Remind students that the Platform Actor and Ball Actor have multiple costumes ready for them to use! For example, here is a sample of the various Platform Actor costumes they can choose from:

	Golf Club
	Grass
	Ground
	Hockey Stick
	Log
	Pipe
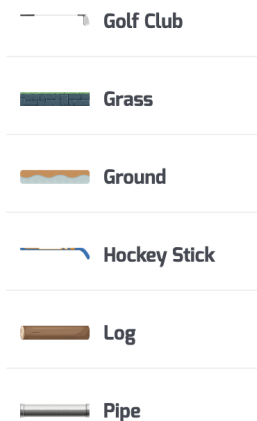
- Encourage students to have fun and get silly with their game. For example, ask them to try bobbing their head up, down, left, and right!
- Time permitting, encourage students to add background music and/or sound effects. Give a hint and ask them to experiment with the "play sound" block:


- Do students want to add a timer to their game? Make sure students add a Timer Actor then experiment with their code, making one change at a time and noting changes to their game. Note that there are different ways to add a timer to the game, but here's one example to help them get started:

  Notice how the timer keeps going once the ball hits the ground? Time permitting, see if students can add onto the code and make the timer stop when the game ends (the ball hits the ground).
- Time permitting, encourage students to further enhance their game by duplicating the Ball Actor! Try inviting students to pick a different costume for each ball. Here is a sample of the various Ball Actor costumes they can choose from:

	Apple
	Baseball
	Basketball
	Beach Ball
	Bowling Ball
	Cinnamon Roll
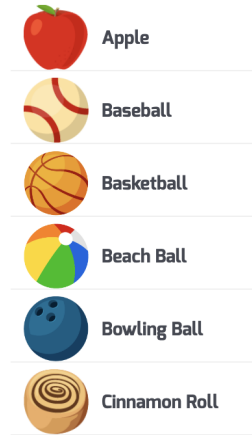
Extended Activities

Discussion

Ask your students…

- What ideas do you have for a physics project/game using Tynker? (Answer example: Simulate gravity in space.)
- Who can explain the behavior of an active, static Actor? (Answer: An active, static Actor will collide with other active Actors but will not fall to the ground.)
- True or False: If an Actor is static, the Actor is affected by gravity. (Answer: False)
- True or False: If an Actor is active, the Actor is affected by physics. (Answer: True)
- Is an inactive Actor affected by physics? (Answer: No, an inactive Actor will not be affected by physics.)

Going Beyond Balancing Act

If your students enjoyed Balancing Act, 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:

- **Even More Free AI Tutorials** — https://www.tynker.com/blog/tynker-toolbox-the-ai-blocks/

- **Artificial Intelligence 101** — This AI course uses Tynker's fun and friendly code blocks. Webcam projects powered by AI put your students at the center of the action! Create interactive Snapchat-style costumes, hand-tracking apps, activity games, and so much more. Along the way, students understand how AI models are driving their amazing inventions behind the scenes. Students even explore natural language processing (NLP), the technology used by modern chatbots like ChatGPT. Find out more here:
  https://www.tynker.com/k8-school/coding-curriculum/artificial-intelligence-101
- **Artificial Intelligence 401** — This JavaScript course is perfect for experienced students looking to implement AI in their programs. Learn how to use MediaPipe, an AI solution from Google, and explore topics like neural networks, supervised learning, and much more. More than that, students dive into AI ethics and consider how AI might impact our future. Find out more here:
  https://www.tynker.com/k8-school/coding-curriculum/artificial-intelligence-javascript-401
- **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.
- **Robotics** — Tynker integrates with connected toys, including 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 90,000 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, Java, Python, and robotics
- 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 Balancing Act.

How do I prepare for Balancing Act?

Familiarize yourself with the material. After selecting your Tynker lesson (Balancing Act), read through the 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.

How does this project use AI?

This project utilizes a pre-trained AI body tracking model. The AI model has been trained on a large dataset of images featuring the human body. When your students interact with the project, the model analyzes the incoming video feed to identify and follow the movements of the user's body landmarks, specifically of the face, providing accurate and instantaneous tracking. Using Tynker's AI blocks, students can apply the AI technology to build interactive, sophisticated projects. Let's see what your students will create!

Who is this activity for?

Balancing Act is intended for students in grades 3-5 (U.S.) or years 4-6 (U.K.) with some coding experience.

How do my students code their project?

The Balancing Act DIY module includes a workspace for students to code their project. The section on the left is a tutorial tab that gives step-by-step directions and describes what is happening in each step. Students can code their project by dragging code blocks to the middle coding panel.

What will my students learn?

Students will combine creativity and coding concepts to create an interactive project called "Balancing Act" that uses Tynker's AI model. Additionally, students can also expand on their project while experimenting with new code blocks. In this process, students will develop debugging and logical reasoning skills.
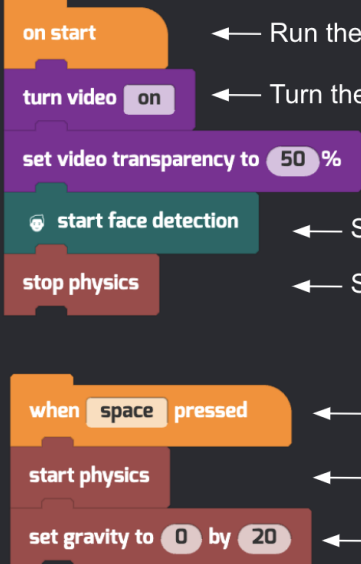
What do the code blocks do?

Below is sample code for the Actors.

 Stage:

on start ←— Run the code attached to this block when the Play button is clicked.

turn video `on` ←— Turn the video feed on.

set video transparency to `50` % ←— Set the transparency of the video feed to 50%.

start face detection ←— Start the AI model that is trained to detect face landmarks.
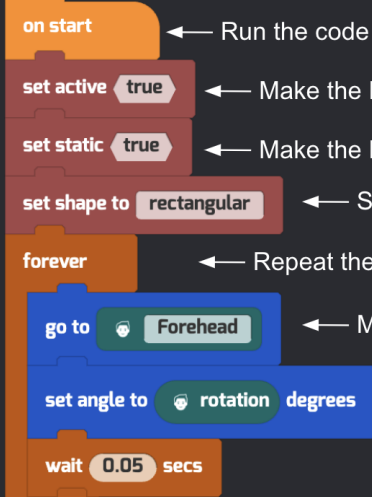
stop physics ←— Stop the physics engine.

when `space` **pressed** ←— Run the code attached to this block when the spacebar is pressed.

start physics ←— Start the physics engine.

set gravity to `0` **by** `20` ←— Set the horizontal and vertical forces of gravity.

Platform:

on start ←— Run the code attached to this block when the Play button is clicked.

set active `true` ←— Make the Platform Actor react to gravity and collisions.

set static `true` ←— Make the Platform Actor not fall to the ground.

set shape to `rectangular` ←— Set the Platform Actor's bounding box to "rectangular".

forever ←— Repeat the code inside this loop forever.

go to `Forehead` ←— Make the Platform Actor go to the player's forehead.
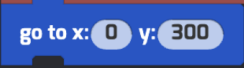
set angle to `rotation` **degrees** ←— Point the Platform Actor in the direction of the player's forehead.

wait `0.05` **secs** ←— Wait a specified amount of seconds.

Ball:

on start ←—— Run the code attached to this block when the Play button is clicked.

set active `true` ←—— Make the Ball Actor react to gravity and collisions.

set static `false` ←—— Make the Ball Actor fall to the ground.

set shape to `circular` ←—— Set the Ball Actor's bounding box to "circular".

go to x: `0` y: `300` ←—— Make the Ball Actor go to the specified x/y-location on the Stage.

Note: This is an open-ended DIY project. Above is an example that uses the provided code blocks to create a Balancing Act project.

How can I contact the Tynker support team?

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