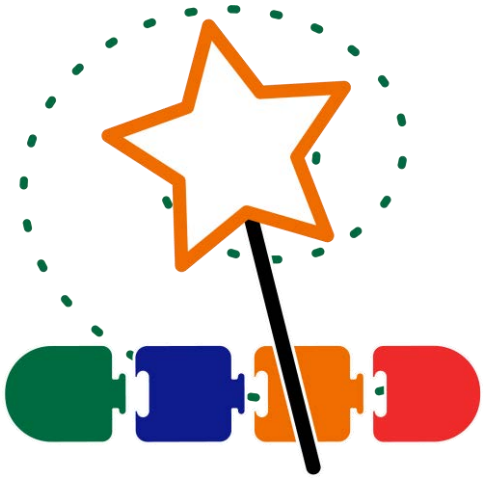
TECHNO Whiz

Teacher Guide

Lessons for Primary and Elementary School Students



Technology Course
using

ScratchJr

Become a programming whiz kid. Build animated scenes, collages, and games.

In this course, students jump into the world of coding. They learn how to sequence blocks in ScratchJr to build simple scripts and loops. Have fun! Design silly scenes, feed a pet monster, explore a magical land, race to the finish line, and more! This fun-filled introduction to programming will spark students' imaginations. Invite curious young minds to become whiz kids!

TECHNOKids®

Table of Contents

Introduction

This section provides valuable information about teaching TechnoWhiz. It includes a description of the Teacher Guide, as well as an overview of the course. In addition, there are ideas for implementation and technology integration.

For additional guidance, open the course in TechnoHub and select Get Started to access preparatory steps, resource list, and scheduling timetable.

How to Use This Guide

This Teacher Guide contains the following:

Getting Started – this section contains a course description, as well as ideas for implementation.

Course Instructions – The course is comprised of six sessions, each focused on a problem-solving task that aligns with the project theme. Each session includes assignments that break down the task into manageable steps. The components of each session are as follows:

- Overview – An explanation of the session activities and their purpose.
- Materials – A list of handouts, sample files, templates, and teacher resource materials needed to teach the session.
- Teaching Strategies – Instructional methods recommended for teaching the activities.
- Lesson Plan – A detailed list of each step in the session.
- Learning Objectives – A summary of the content knowledge and technical skills taught throughout the session.
- Assignments – A session consists of assignments completed by students. Actions to be performed on the computer by the student are indicated with a triangle (▷). Background information is indicated with a dash (–).
- Review – A session review contains a list of fill-in-the-blank, multiple choice, or short-answer questions intended to review ScratchJr tools and programming knowledge (answers included).
- Skill Review – An additional assignment intended to review coding skills (includes completed sample).
- Extension Activity – An additional activity that relates to the problem-solving task presented in the session.

Appendices – this section contains additional information or materials including the following resources.

- Assessment Tools – Skill summary and marking sheet to evaluate coding skills and the final ScratchJr culminating project.
- Glossary – A definition of each term.
- Contact Information – How to contact TechnoKids Inc. for curriculum support.

TechnoWhiz Overview

Introduction to TechnoWhiz

In this course, students jump into the world of coding. They learn how to sequence blocks in ScratchJr to build simple scripts and loops. Have fun! Design silly scenes, feed a pet monster, explore a magical land, race to the finish line, and more! This fun-filled introduction to programming will spark students' imaginations. Invite curious young minds to become whiz kids!

Students complete the following tasks:

- In session 1, students become programming whiz kids. Their first assignment is to sequence a series of tasks to learn about the job of a programmer. Next, they open ScratchJr to investigate the function of many of the tools. By adding Motion blocks to the programming area, they learn how to code a character's movement. Next, they connect blocks together to form a script that sequences actions. Finally, students cause the script to loop. At the end of the session, they combine their coding skills to create a silly scene. It comes to life by combining a colorful background with moving characters.
- In session 2, the programming whiz kids take the Counting Challenge. They learn to direct how many steps a character moves to reach a goal. To start, they solve puzzles to help the cat eat the cake, the dog catch the ball, and the horse reach the barn. Next, they explore how to edit the Motion coding blocks to set the number of steps. Once they have mastered this skill, students build scripts to help characters get home. Can they do it?
- In session 3, the programming whiz kids design a game. They build code that has a player feed a pet monster. To start, they create an opening sequence to attract attention. It has the character talk, grow, shrink, and blink. The action is controlled using the Wait and Repeat blocks. Afterwards, students apply their knowledge to add food to the stage that when clicked moves to the pet monster and then disappears. What does the pet monster like to eat?
- In session 4, the programming whiz kids design an interactive magical land. The game invites players to explore the place to discover surprises. By clicking on objects, they can cause flowers to grow, a sun to spin, or dragons to hop. The speed of the action is controlled to make it happen slow or fast. What amazing things are hidden in the scene?
- In session 5, the programming whiz kids build a racing game. Players watch the action and then must select the winner. If they pick correctly, the racer will do a celebration dance. Who will reach the finish line first?
- In session 6, the programming whiz kids invite others to play their racing game. Players rate what they liked the most about the activity. Afterwards, the young game designers answer questions about coding the ScratchJr project.

Implementation and Technology Integration Ideas

TechnoWhiz develops computational thinking in primary and elementary school students. The programming activities are ideal for Grades 2 and up. Step by step instructions explain how to build animated scenes, interactive collages, and games using ScratchJr. Have fun! Animate silly scenes, feed a pet monster, design a magical land, or create a race.

Ideas for Implementation

TechnoWhiz introduces basic programming concepts to beginners. The activities are suitable for any teaching situation. Select the option that works best for you and your students:

- *Coding Unit with Primary Students*: Assignments in Sessions 1-3 in TechnoWhiz are ideal for students new to ScratchJr. They learn how to design animated scenes and a simple game. The emphasis is on coding basics including how to build scripts, sequence commands, and create simple loops. The activities focus upon intentionally coding characters to move.
- *Coding Unit with Elementary Students:* Once students understand the fundamentals of coding in Sessions 1-3, they extend their learning in Sessions 4-6. The activities are ideal for young programmers that are ready for a challenge. They learn how to direct the timing of events to produce an interactive collage and original racing game.
- *Hour of Code*: If you only have one class to teach coding there are many assignments in TechnoWhiz that can be used for this purpose. If your students are beginners, they can develop simple animations. Assignment 3 targets how to build a looping script, Assignment 7 explores controlling movement, and Assignment 8 focuses upon changing the appearance of a character. If your students have existing knowledge of ScratchJr, the skill reviews in Sessions 1-5 are excellent challenges.
- *Makerspace Workshop Series*: If you are running a workshop series as part of an after-school program or community event, then you will need to select assignments that fit the number of classes offered. Consider the age range and coding abilities of students. Pick activities they will enjoy. For example, design a treasure hunt, far away planet, or hop to the top race.

Technology Integration Suggestions

The TechnoWhiz course is primarily a STEM project that teaches coding. However, the activities also integrate into other areas of curriculum including language arts, mathematics, social studies, and visual arts.

- *Language Arts*: Include TechnoWhiz as a language arts unit. The coding activities are a form of visual storytelling. Apply story writing elements such as characters, setting, and plot to produce animated scenes and games. In addition, strengthen communication skills by having students reflect upon their programming decisions.

- *Mathematics*: Integrate TechnoWhiz into an existing problem-solving unit in Math class. The assignments are an ideal fit because coding requires mathematical and logical thinking. As well, the activities develop critical thinking skills by having students systematically plan, design, code, and test their projects.

- *Social Studies*: Transfer map making skills to a new task. The assignments in Session 2 have students apply knowledge of directions (left, right, up, and down) to move a character from one location to another.
- *Visual Arts*: Introduce digital paint tools. The Session 4 and 5 extension activities explain how to create original characters and backgrounds. This is a great way to express creativity.

Session 1
Become a Whiz Kid

In this session, students become programming whiz kids. Their first assignment is to sequence a series of tasks to learn about the job of a programmer. Next, they open ScratchJr. to investigate the function of many of the tools. By adding Motion blocks to the programming area, they learn how to code a character's movement. Next, they connect blocks together to form a script that sequences actions. Finally, students cause the script to loop. At the end of the session, they combine their coding skills to create a silly scene. It comes to life by combining a colorful background with moving characters.

Session 1 Getting Started

Overview

In this session, students become programming whiz kids. Their first assignment is to sequence a series of tasks to learn about the job of a programmer. Next, they open ScratchJr to investigate the function of many of the tools. By adding Motion blocks to the programming area, they learn how to code a character's movement. Next, they connect blocks together to form a script that sequences actions. Finally, students cause the script to loop. At the end of the session, they combine their coding skills to create a silly scene. It comes to life by combining a colorful background with moving characters.

Materials

- ScratchJr
- Become a Whiz Kid template (optional resource for Assignment 1)
- Sample files:
    - *get moving* video
- ScratchJr Flashcards: (optional)
    - Interface, Block Categories, Motion Blocks, Triggering Blocks, End Blocks
- Session 1 Review: About Coding
- Session 1 Skill Review: Under the Sea
    - *under the sea* video
- Session 1 Extension Activity: Move a Character Diagonally

Teacher Preparation

(Refer to the Preparing to Teach section of this guide for instructions)

- Plan a device management strategy. ScratchJr saves student work in an internal database. The files are not web-based and cannot be opened on another computer. For this reason, students must be assigned a specific device. It should be used throughout the TechnoWhiz project.
- Make the files in the Whiz folder available to students.
- View the *get moving* video to gain an understanding of the completed project.
- (optional) Gather flashcards to introduce and/or review programming blocks and ScratchJr tools.

Teaching Strategy

In this course, students jump into the world of coding. They learn how to use ScratchJr to sequence blocks to build animated scenes, collages, and games. This fun-filled introduction to ScratchJr will spark students' imaginations. Invite curious young minds to become whiz kids! Explain scenario:

> *In this course, you are going to become a programming whiz kid. You will learn how to use ScratchJr to code animations, collages, and games. Have fun designing silly scenes, feeding a pet monster, exploring a magical land, racing to the finish line, and more!*

In this session, students create an animated scene with a colorful background and moving characters. Explain scenario:

> *In this session, you will learn how to use ScratchJr to make an animated scene. To start, you will explore the program to discover the function of each Motion block. Next, you will learn how to join the coding blocks together to make a script that tells a character what actions to do. Afterwards, you will add a loop that makes the movement happen non-stop. Finally, you apply your knowledge to build a program that has characters walking and hopping across the stage. This silly scene is sure to attract attention!*

Assignment 1: Become a Programming Whiz Kid

In this assignment, students are introduced to programming. This activity is also available digitally. Refer to the *Become a Whiz Kid* file if you want to drag and drop the steps into place.

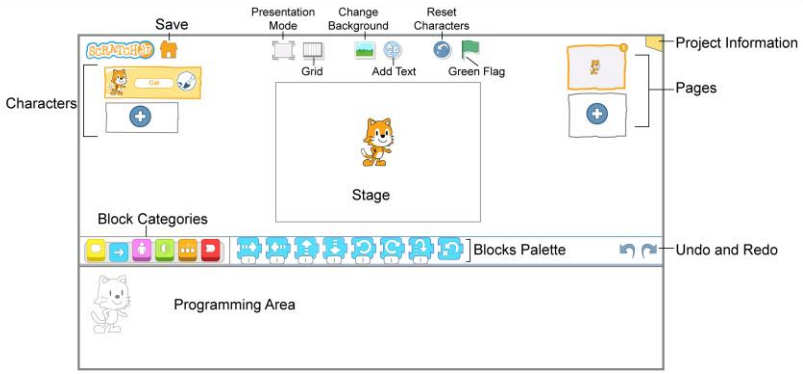
Introduce the following terminology:

- *program*: A list of steps that tell a computer what to do.
- *programmer*: A person who writes a computer program.

Assignment 2: Explore ScratchJr

In this assignment, students sequence blocks to make a character move. The goal is to have students understand how to use ScratchJr to build a script. To start, they explore the interface. They discover that the coding blocks are sorted into categories. Next, they learn how to add a block into the programming area. To identify the function of the Motion blocks they use each as an action button. Afterwards, they connect the blocks together to cause many actions to happen in order.

Parts of the ScratchJr Window



Introduce the following ScratchJr terminology:

- *stage*: A place used to make pictures stories, and games.
- *block:* A command that tells the computer what to do.
- *block categories*: A palette used to sort blocks into groups such as Motion, Looks, and Sound.
- *blocks palette*: A palette that shows the blocks in a category.
- *programming area*: The area where blocks are joined together to build scripts that tell characters what to do.
- *script*: A chain of commands that do a task.

Introduce the following ScratchJr programming blocks and tools:

 *Start on Green Flag*: Run the program when the Green Flag is clicked.

 *Move Right*: Step the character to the right.

 *Move Left*: Step the character to the left.

 *Move Up*: Step the character towards the top of the stage.

 *Move Down*: Step the character towards the bottom of the stage.

 *Turn Right*: Rotate the character clockwise.

 *Turn Left*: Rotate the character counterclockwise.

 *Hop*: Move the character up by an amount and then down again.

 *Go Home*: Return the character to its starting point.

 *Green Flag*: Run the program.

Assignment 3: Loop It

In this assignment, students build a simple script to learn about loops. At the beginning of the lesson they design a script that stops when it reaches the end. Next, they edit it to include a *Repeat Forever* block. This causes the action to run forever. Students will apply their coding skills in the follow assignment to create an animated scene with characters that move continuously around the stage.

Introduce the following terminology:

- *loop*: Repeat a script non-stop, over and over.

Introduce the following ScratchJr programming blocks and tools:

 *End*: Stop the script from running.

 *Repeat Forever*: Run a script over and over.

 *Stop:* Make a program stop running.

Assignment 4: Get Moving

In this assignment, students apply their coding skills to create an animated scene. It will include a background, as well as many characters. The action will loop to make the silly picture play non-stop. To spark creativity, show the *get moving* video prior to beginning the lesson. Encourage students to add many characters with different scripts.

Introduce the following terminology:

- *background*: An image used to set the scene.

Introduce the following ScratchJr tools:

 *Change Background*: Apply a background image to the stage.

 *New Character*: Insert a character from a library.

Lesson Plan

Assignment 1: Become a Programming Whiz Kid

- What is a program?
- Put the steps in order to understand that the sequence of blocks in a program controls when each action happens.

Assignment 2: Explore ScratchJr

- Open a new project in ScratchJr.
- Study the ScratchJr window. Identify the location of the stage and programming area.
- Click each block category to notice programming blocks are sorted by function.
- Add a Motion block to the programming area.
- Press the Motion block to move the character on the stage.
- Drag each Motion block into the programming area. Press each one to discover its function.
- Delete *Go Home* to learn how to remove unwanted blocks.
- Connect the Motion blocks in the programming area together to form a script.
- Apply coding skills to move the character around the stage.
- Close ScratchJr.

Assignment 3: Loop It

- Open a new project in ScratchJr.
- Add the *Start on Green Flag* block to the programming area to set how the script will run.
- Use Motion blocks to build a script that causes the character to move around the stage.
- Add an *End* block to the script to stop it from running when it reaches the last block.
- Run the program using the *Green Flag*.
- Edit the script to delete the *End* block from the script.
- Add the *Repeat Forever* block to the end of the script to cause it to loop.
- Run the programming using the *Green Flag* to see the character move non-stop.
- Stop the program from running.
- Apply coding skills to move the character around the stage forever.
- Close ScratchJr.

Assignment 4: Get Moving

- Open a new project in ScratchJr.
- Rename the project Student loop.
- Apply a background to the stage to create a backdrop for the scene.
- Delete the *Cat* character from the stage.
- Insert a new character onto the stage from a library.
- Apply coding skills to build a looping script that causes the character to move around the stage forever.
- Complete the animated scene by inserting more characters. Build scripts to make each one move differently.
- Close ScratchJr.

Learning Objectives

Programming Skills

Programming Basics

- recognize that a program has a list of steps that tell a computer what to do
- understand that the steps in a program are sequenced and they happen in order
- sequence the steps in a common task
- add a coding block into the programming area
- use a coding block as an action button
- connect coding blocks together to form a script
- edit a script by deleting or adding blocks

ScratchJr Basics

- open and close a program
- create a new project
- identify the parts of the program window including the stage, programming area, blocks categories, and blocks palette
- understand that programming blocks are organized into categories
- name a project
- run or stop a program

Trigger or End a Script

- trigger a script to run when the Green Flag is pressed
- end a script so that it stops running when it reaches the last block
- loop a script so that it repeatedly runs non-stop

Working with Characters

- add a character from a library
- delete a character from the stage
- position the start point of a character on the stage

Move a Character with Motion Blocks

- move a character left, right, up, or down on the stage
- turn a character clockwise or counterclockwise on the stage
- cause a character to hop up and down on the stage
- return a character to its starting position on the stage

Working with Backdrops

- add a backdrop from a library

Applied Technology

- explore the ScratchJr interface
- animate a character to move around the stage
- create a silly scene with characters that move non-stop using a loop

This is a preview of the teacher guide.
Pages have been omitted.

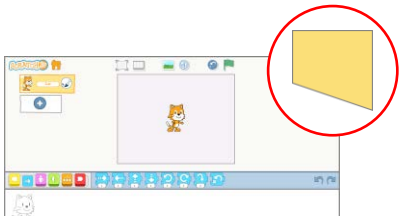
Assignment 4 Get Moving

You are a programming whiz kid! Use your skills to make characters move around the stage to create a silly picture.

Open a New Project in ScratchJr and Name It

- Open a new project in ScratchJr.

- Click *Project Information*.

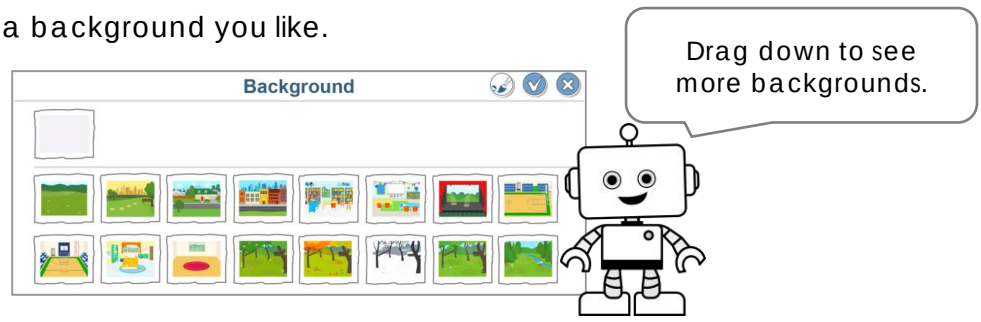


- Type Student loop. Click 

Add a Background

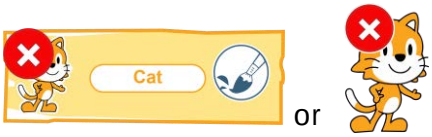
- Click *Change Background*.

- Double click on a background you like.



Delete the Cat Character

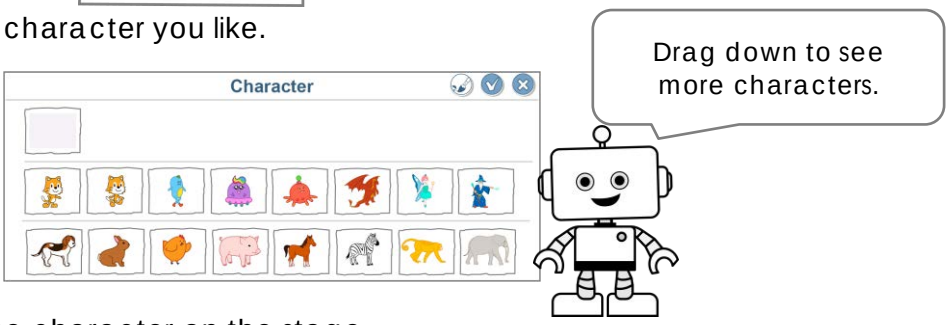
- Click and hold the character.
- Press *Delete*. 



or

Add a New Character

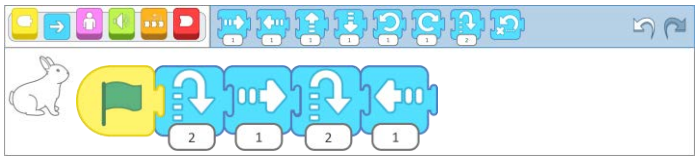
- Click *New Character*. 
- Double click on a character you like.



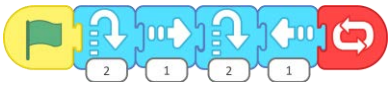
- Drag it to place the character on the stage.

Build a Script that Loops

- Click Triggering.  Add *Start on Green Flag*. 
- Make the character move. Click *Motion*.  Use your skills to make a script.



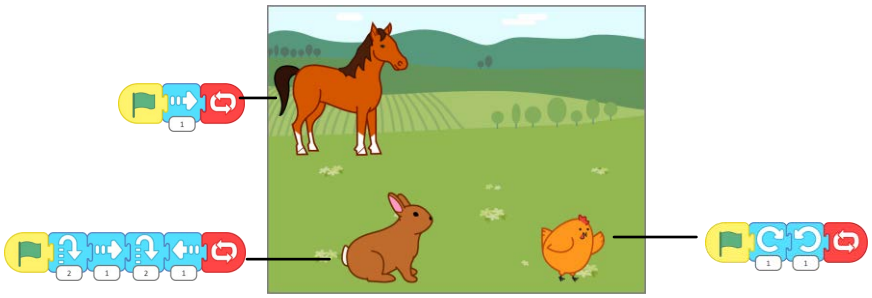
- Loop the script. Click *End*.  Add *Repeat Forever*  .



- Click *Green Flag*  to see the character move.

Make a Moving Picture

- Add more characters. Make each one move. Be creative!

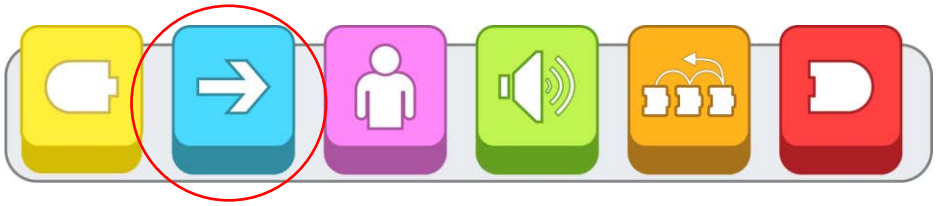


Close ScratchJr

Session 1 Review: About Coding

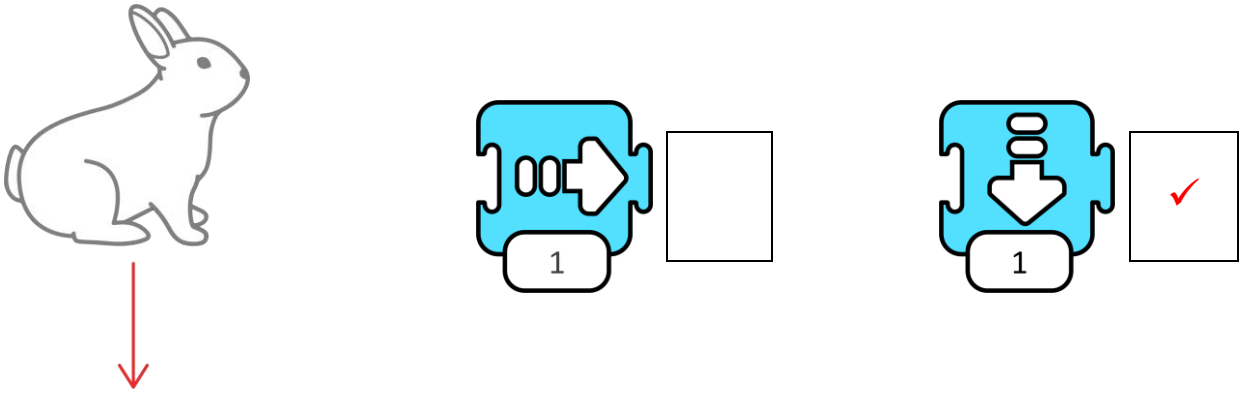
How Do You Do the Job?

1. Circle the tool that shows blocks to make something move.



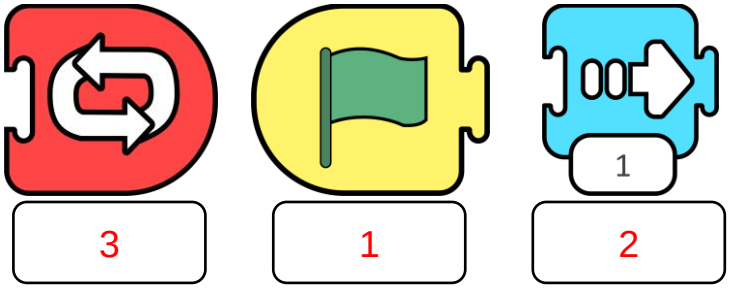
Match the Picture to the Block

2. Look at the picture. Which block will make the rabbit move down?



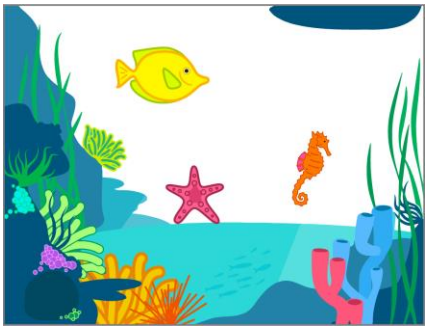
Put the Steps in Order

3. Number the steps. What block is first, second, and third?



Session 1 Skill Review: Under the Sea

Create a picture with fish that move.



---

1. Open ScratchJr.

2. Click *Home*.

3. Click *New Project*.

---

4. Click *Project Information*.

5. Type student sea. Click

---

6. Add a background.

7. Pick underwater.

8. Delete the Cat. 

9. Click *New Character*.  Pick Starfish. 

10. Drag the Starfish to place it on the stage.

11. Make the starfish spin forever. 

---

12. Click *New Character*.  Pick Fish. 

13. Drag the Fish to place it on the stage.

14. Make the fish swim across the stage. 

---

15. Click *New Character*.  Pick Seahorse. 

16. Make the seahorse swim up. 

---

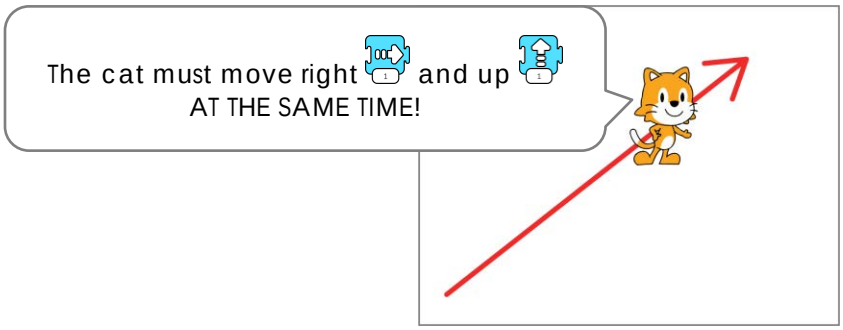
17. Run the program. 

18. Stop the program. 

19. Close ScratchJr.

Session 1 Extension Activity: Move a Character Diagonally

You can code a character to move *smoothly* in a diagonal direction. Try it!



A character can have more than one script.
This will make more than one action happen AT THE SAME TIME.

---

1. Open a new project in ScratchJr. 

2. Make the character move across the stage.

3. Make the character move up to the top of the stage.

4. Run the program. 

5. Stop the program. 

6. Close ScratchJr.

Session 2
Count and Code

In this session, the programming whiz kids take the Counting Challenge. They learn to direct how many steps a character moves to reach a goal. To start, they solve puzzles to help the cat eat the cake, the dog catch the ball, and the horse reach the barn. Next, they explore how to edit the Motion coding blocks to set the number of steps. Once they have mastered this skill, students build scripts to help characters get home. Can they do it?

Assignment 5: Count to Solve the Puzzles

Assignment 6: Code the Steps

Assignment 7: Take the Counting Challenge

Session 2 Review: Pick the Correct Code

Session 2 Skill Review: X Marks the Spot

Session 2 Extension Activity: Move Around the Town Game

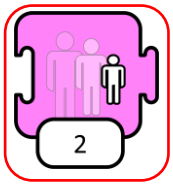
This is a preview of the teacher guide.
Pages have been omitted.

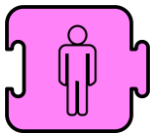
Session 2 Review: Pick the Correct Code

What Block Will Do the Job?

Circle the correct block.

1. Make a character small.





---

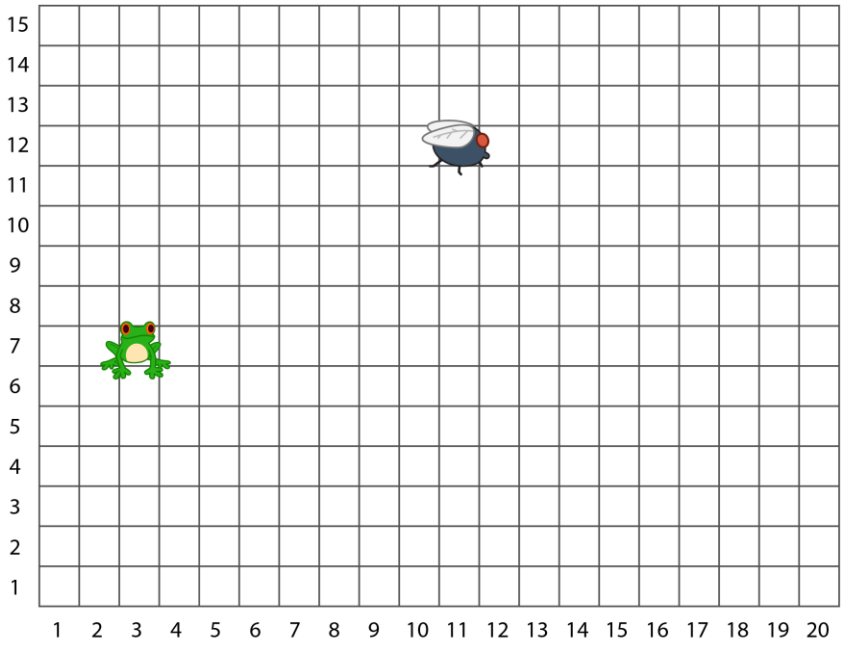
2. Make a character big.

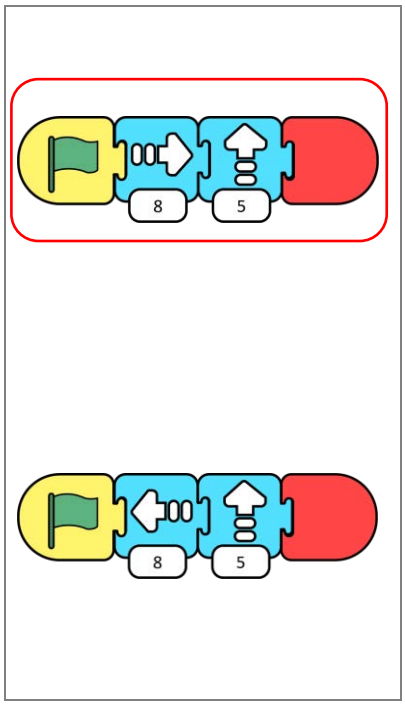
---

3. Change the background.

What Script Will Work?

4. Circle the script that will move the frog to catch the fly.





Session 2 Skill Review: X Marks the Spot

Use your skills to move a character to a special spot.

Plan what happens!

- Where do you want the character to start?
- Where do you want the character to end?
- Why are they going there?

IDEA: Get the toy from the shelf.

Use this idea or come up with your own.

---

1. Open a new project in ScratchJr.

2. Click *Project Information*.

3. Type student count. Click

---

4. Click *Change Background*.

5. Pick a background. Here are some ideas:


go to bench


go to chair


go to spotlight


go to shelf


go to house

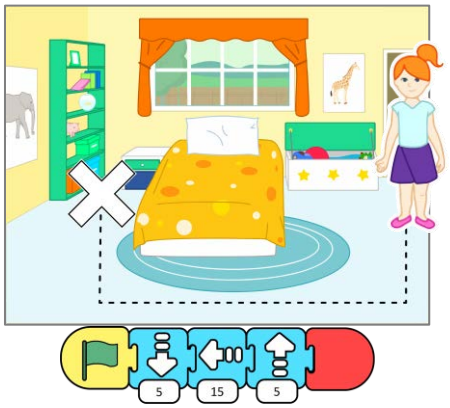
---

6. Delete the *Cat*.

7. Use your skills to add a character.

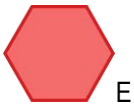
8. Click *Grid*. 

9. Make a script that moves the character from the start to the end. For example:

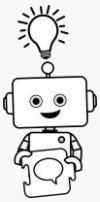


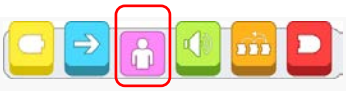
---

10. Test it! Run the program. 

11. Stop the program.  Edit it.

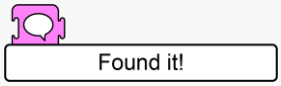
CODING CHALLENGE
Have your character say what they are doing.



Click *Looks*. 

Add a *Say* block. 

Type what is happening.

At the end of the script add another *Say* block. 

---

12. Share your work. Tell someone about your program.

13. Close ScratchJr.

This is a preview of the teacher guide.
Pages have been omitted.

Session 3
Create a Pet Monster Game

In this session, the programming whiz kids design a game. They build code that has a player feed a pet monster. To start, they create an opening sequence to attract attention. It has the character talk, grow, shrink, and blink. The action is controlled using the Wait and Repeat blocks. Afterwards, students apply their knowledge to add food to the stage that when clicked moves to the pet monster and then disappears. What does the pet monster like to eat?

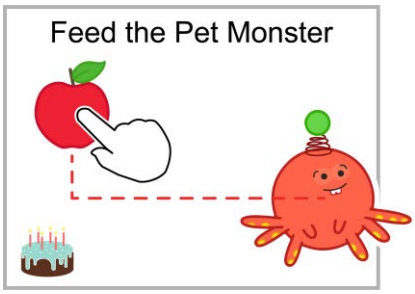
This is a preview of the teacher guide.
Pages have been omitted.

Assignment 9 Design a Pet Monster Game

Make a game for your friends to play.

What does your monster eat?

You will add food to the stage. When a player clicks on the food it will move to the pet monster. The food will then hide to make it look like the monster ate it.

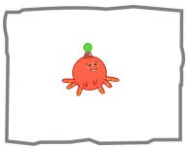


About the Start on Tap Block

To make the game you will use the *Start on Tap* block. A person must click or *tap* the item on the stage. This makes the script run to move the food.

Open the Saved Pet Monster Project

- ▷ Open ScratchJr.
- ▷ Click *Home*.
- ▷ In the *My Projects* list find Student pet. Click it.



Add a Title for the Game

- ▷ Click *Add Text*.
- ▷ Type a title such as Feed My Pet Monster. Or, come up with your own idea!

Format the Title

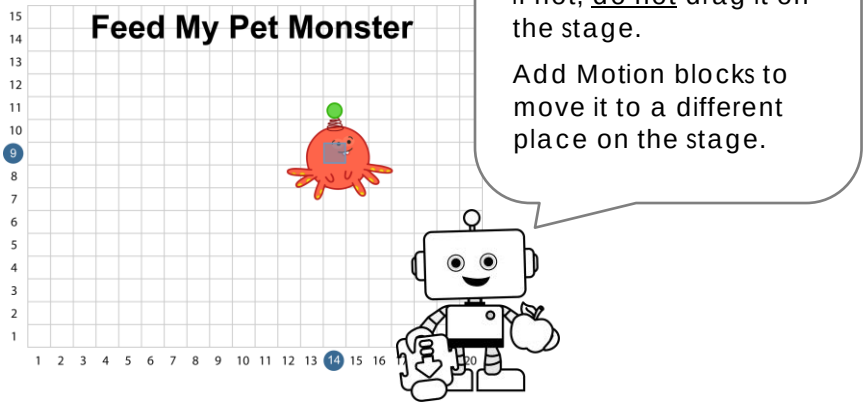
- Click *Font Size*.
- Pick a size.

- Click *Font Color*.
- Pick a color.

- Click on the stage to close the text box.
- Click the title and drag it on the stage to move it.

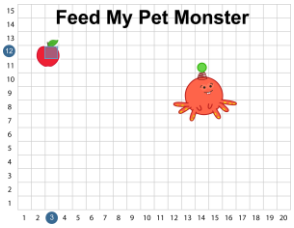
Play the Start of the Game to See Where the Food Needs to Go

- Turn on the grid.
- Click *Green Flag* to run the monster's script.
  - Where is the monster? This is where the food must go.

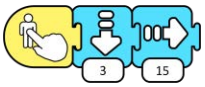


Feed the Pet Monster

- Click *New Character*. 
- Pick an item the monster will eat.  Place it on the stage.

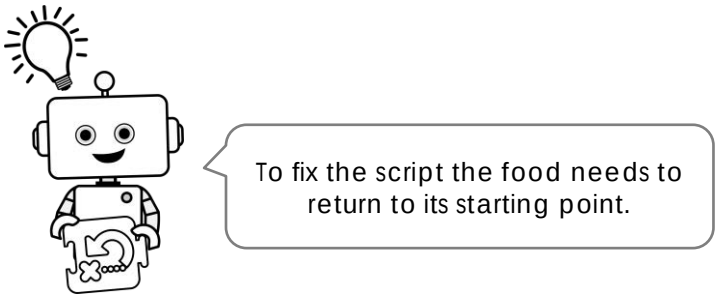


- Click Triggering.  Add *Start on Tap*. 
  Build a script to have it move to the monster.

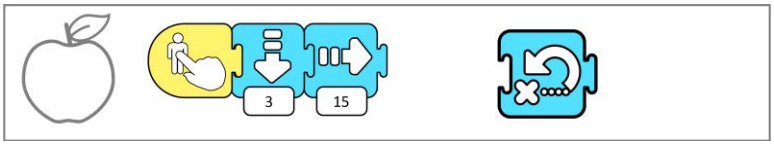


- Click the food to test it. Did it go to the monster?

Edit the Number of Steps to Move the Food



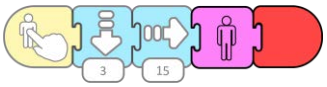
- Click *Motion*.  Drag *Go Home*  into the programming area.
- DO NOT connect it to the script.



- Press *Go Home*  to have the food return to its start point.
- Edit the number of steps. 
- Click the food to test it. Did it go to the monster?

Hide the Food to Make It Look Like the Pet Monster Ate It

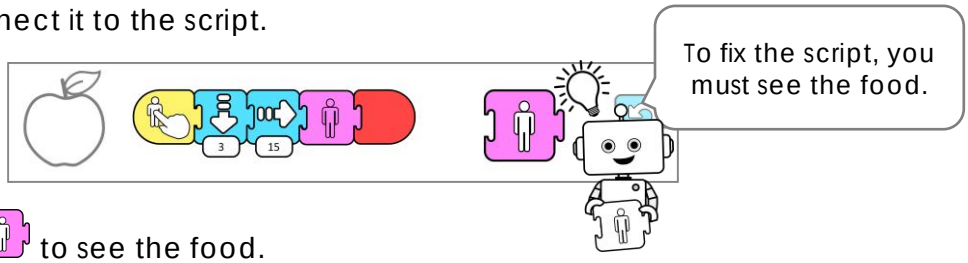
- ▷ Click *Looks*. Add a *Hide* block.
- ▷ Click *End*. Add an *End* block.



- ▷ Click the food to test it. Does it look like the monster is eating the food?

Do You Need to Make More Changes to the Script?

- ▷ Click *Looks*. Drag *Show* into the programming area.
- ▷ DO NOT connect it to the script.



- ▷ Press *Show* to see the food.
- ▷ Edit the script.
- ▷ Click the food to test it. Does it look like the monster ate it?

Be a Game Designer

- ▷ Use your skills to make the game fun to play. Pick from the ideas:
  - Have the monster say Feed me!
  - Add instructions that tell how to play the game.
  - Add more food items.
- ▷ Play the game:
  - o Click Presentation Mode.
  - o Click *Green Flag* to run the game. Feed the pet monster.
  - o When done, click.

Close ScratchJr

This is a preview of the teacher guide.
Pages have been omitted.

Session 6
Game Time

In this session, the programming whiz kids invite others to play their racing game. Players rate what they liked the most about the activity. Afterwards, the young game designers answer questions about coding the ScratchJr project.

Assignment 15: Racing Fun

Assignment 16: Tell Others About Your Game

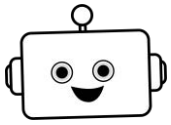
Session 6 Extension Activity: I am a Programmer

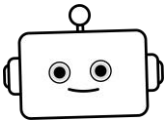
This is a preview of the teacher guide.
Pages have been omitted.

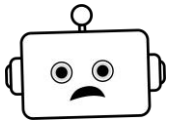
Session 6 Extension Activity: I am a Programmer

Answer the questions about the TechnoWhiz course.

1. How much did you learn about coding?

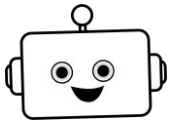

lots

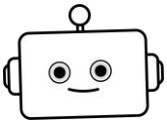

some

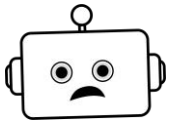

not much

---

2. What do you think about ScratchJr?


love it

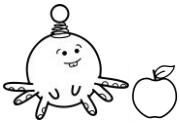

like it


dislike it

Why?

---

3. What game did you like to make the most?


feed pet monster


magic land


race

Why?

---

4. What one tip would you give to someone new to using ScratchJr?

Appendices

Refer to the appendices for additional resources:

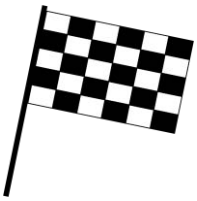
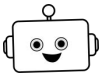
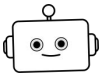
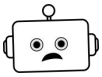
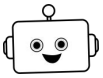
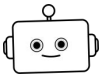
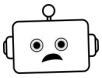
Appendix A - Assessment Tools

Appendix B - Glossary

Appendix C - Contact Information

This is a preview of the teacher guide.
Pages have been omitted.

Racing Game Marking Sheet

	Excellent Work!	Doing Great!	Keep Trying!
 The Race: The game has a title. The game has three racers that move at different speeds. The background has a start and end point for the race.			
 Game Play: The instructions tell the player how to play the game. The racer that won does a happy dance when clicked by the player. The racers that lost tell the player to *try again* when clicked.			
 Creativity: The race idea is original. The racers suit the type of race. The game is fun to play. The happy dance is creative.