**Lesson 4 - Bouncing Ball**

**Problem statement:**

Your team must design a new game and present your product to the leadership team of Games R Us, your goal is to get approval from the Games R Us executives to produce your game.

This lesson will teach students the basics of how to make a ball bounce using Scratch.

**Learning objectives:**

Students will be able to use graphic coding blocks to have a ball bounce around the screen with three added features.

**Washington State Computer Science Learning Standards (3-5)**

1B-A-2-1

Apply collaboration strategies to support problem solving within the design cycle of a program

1B-A-5-3

Create a plan as part of the iterative design process, both independently and with diverse collaborative teams (e.g., storyboard, flowchart, pseudo-code, story map).

1B-A-5-4

Construct programs, in order to solve a problem or for creative expression, that include sequencing, events, loops, conditionals, parallelism, and variables, using a block-based visual programming language or text-based language, both independently and collaboratively (e.g., pair programming).

1B-A-3-6

Decompose (break down) a larger problem into smaller sub-problems, independently or in a collaborative group.

1B-A-3-7

Construct and execute an algorithm (set of step-by-step instructions) that includes sequencing, loops, and conditionals to accomplish a task, both independently and collaboratively, with or without a computing device.

**Next Generation Science Standards (NGSS)**

3-5 ETS1-1

Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.

3-5 ETS1-2

Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.

3-5 ETS1-3

Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

**Common Core State Standards (CCSS) Mathematical Practices**

1. Make sense of problems and persevere in solving them.

6) Attend to precision.

7) Look for and make use of structure.

8) Look for and express regularity in repeated reasoning.

**Soft Skills:**

**Partnership for 21st Century Learning (P21)**

* Think creatively
* Reason effectively
* Use systems thinking
* Make judgements and decisions
* Solve Problems
* Communicate clearly
* Apply technology effectively
* Adapt to change
* Be flexible
* Manage goals and time
* Be self-directed learners
* Manage projects
* Produce results

**Connections to career and educational pathways:**

* Software applications developer
* Computer systems analyst
* Computer systems engineer
* Web developer
* Computer programmer

**Materials**:

* Computers
* Student’s Scratch account linked to teacher’s studio within Scratch (this allows for students to save the projects that they create)

**Lesson preparation:**

Students should create a free account on Scratch to be able to save their projects.

**Time required:**

60 minutes

**Grouping of students for instruction:**

Students should be one to one with a computer and sit next to a partner so they can help guide each other.

**What is the instruction?**

Students will be understanding, exploring, and resolving the challenge. Students will be understanding the concept of making the ball bounce around the room. They will then be exploring extra features that their ball will perform each time it bounces off of a wall.

**Lesson:**

1 - Have students work with a partner or individually up to you as a teacher.

2 - Students need to log into their Scratch account and then click on “Create” to start a new workspace to begin their coding project.

3 - First remove the cat, by clicking on the scissors (located above stop sign) and clicking on the cat.

4 - Next get a ball sprite from the sprites library.

5 - Next open up discussion that our goal is to get the ball to bounce off of the walls around the room that it is sitting in. Right now the ball is sitting there not moving. How are we going to get the ball to move? Try some student ideas together.

* The goal is to lead students towards putting the “Move 10 Steps” into a Forever Loop.
* Forever Loop is found in the Control commands
* You will need to add two more blocks into the Forever Loop: “Turn Right 15 Degrees” and “If on Edge, Bounce”
* These three blocks are all found within the Motion commands

6 - To activate the code so the ball will start bouncing they need to add a block called When flag clicked to the top of your code.

* Go to Events and add a When flag clicked block to your workspace.

7 - Test your code

* adjust the degrees turned more or less and
* move the ball to a different starting location by clicking on the stop sign, click on your ball and drag it to a new starting location and click flag.

8 - Now that we have a ball bouncing around the room ask the class what are some extra features that we can add?

* Ball changes color, changes size, makes sound, speed changes are a few ideas.

9 - Next talk about the If/Then block, and when should the ball change features

* Within the Sensing commands there is an option to add into the hexagon space called “Touching”. You can change the “mouse-pointer” to “edge”
* So If the ball touches the edge of the room what is going to happen Then
* This is where you use the brainstormed features list to add blocks to the If/Then loop

10 - Students are challenged to add two features from the brainstormed list and add a third feature that is new and creative.

11 - Have students name their project (located left of green flag). Have them Share their project. If you have your students in a Scratch Studio then have them click on Studio and then click on the specific studio that they want to share it in. once they see a green check mark it is shared with you.

11 - Have students fill out their exit slip within their engineering books and notes to remember about their features that they added.

**Accommodations:**

Students that are visually impaired can sit in the front of the class near the screen. Students that are deaf should be able to follow the steps of programming but will be more successful to have an interpreter to help communicate. Any students that are physically impaired can partner up with another student to be able to be successful and follow along with the lesson.

**Extensions**:

Students that finish early can push themselves to add more detail to their bouncing ball program . They can add a different sprite to give it more animation and effects.

**Assessment**:

The final product in successfully showing a ball bouncing around a room with three special features one of the three being unique.

**References/Resources:**

Students can refer to their engineering books to write notes about their special features added and to answer the exit slip