



Lesson Plan: Code Crusaders - Level Up Your Logic!

Grade Level: 5th-7th

Lesson Objective: Students will explore basic coding concepts like sequences, loops, and conditionals through engaging video games and unplugged activities, understanding how code controls actions and solves problems.

Lesson Motto(s):

- "Think like a coder, act like a hero! Every line matters."
- "Bugs bite? Debug and conquer! Coding challenges make you stronger."

Materials Needed:

- For Video Game Option:
 - Classroom computers with internet access
 - Kid-friendly coding/puzzle games:
 - Lightbot Jr. (free)
 - CodeMonkey Jr. (free trial)
 - Human Fall Flat (optional, less coding-focused but encourages problem-solving with logic)
- For Unplugged Activity:
 - Large whiteboard or chart paper
 - Markers or colored pencils
 - Lego Mindstorms EV3 Education Core Set (optional, for advanced learning)
- Post-Lesson Activity: Design your own "Code Crusader" badge template (see "Post Wrap-up" section)

Lesson Outline:

Warm-up (10 minutes):

- Play a "Simon Says" game with instructions incorporating coding terms like "repeat," "if," and "else."
- Show visuals of robots or machines following specific routines, discussing how instructions control their actions.
- Briefly introduce the concept of "code" as a set of instructions for computers.

Introduction (15 minutes):

- Introduce the students as "Code Crusaders" on a mission to understand the language of technology.
- Explain the three key coding concepts we'll explore:
 - Sequences: Ordering instructions one after another.
 - Loops: Repeating actions multiple times.
 - Conditionals: Making decisions based on certain conditions.
- Use simple analogies or real-life examples to illustrate each concept.

Instructional Activity (20 minutes):

- Divide students into teams.
- For Video Game Option: Teams explore their chosen coding game, gradually progressing through levels that require applying sequence, loop, and conditional coding concepts. Encourage them to discuss their strategies and troubleshoot challenges together.
- For Unplugged Activity: Teams brainstorm a simple task like making a peanut butter and jelly sandwich. On the whiteboard, they write down the steps in order (sequence), including any repetitions (loops) and branching instructions based on ingredients or preferences (conditionals).

Interactive Lab Activity (90 minutes):

- Deepen the learning experience with engaging challenges:
- Video Game Option:
 - Challenge 1: Level Design: Teams create their own levels within the game, incorporating all three coding concepts to lead a character to a specific goal. They can then share and play each other's levels.
 - Challenge 2: Debugging Adventure: Introduce intentional "bugs" (miscoded instructions) in existing levels. Teams work together to identify and fix the bugs, practicing problem-solving and critical thinking.
- Unplugged Activity:
 - Challenge 1: Code Charades: One team member silently acts out their chosen sequence, loop, or conditional code using physical movements. Others guess the type of code and write it down correctly.
 - Challenge 2: Robot Relay Race: If using Lego Mindstorms, teams program their robots to complete a simple obstacle course involving sequences, loops, and decision-making based on sensor inputs.

Culminating Activity/Competition (20 minutes):

- Each team presents their findings or creations from the Interactive Lab Activity.
- For video games, teams showcase their designed levels or demonstrate successful debugging. For the unplugged activity, teams present their charade solutions or robot performances.
- The class votes for the most creative, effective, or technically impressive use of coding concepts.

Wrap-up (10 minutes):

- Recap the key takeaways about sequences, loops, and conditionals and how they control actions in coding.
- Briefly discuss the real-world applications of coding across various fields.
- Introduce the post-lesson activity: designing their own "Code Crusader" badges.

Post Wrap-up (15 minutes):

- Provide each student with a blank badge template (a shield or coding symbol, for example). They can decorate it with drawings, colors, and slogans representing their newfound coding skills and favorite concepts.
- Encourage them to wear their badges with pride, symbolizing their journey as Code Crusaders conquering the world of logic and technology!

Homework/Extended Learning:

Coding Adventures:

- Programming Challenge: Encourage students to choose a simple task they do every day (brushing teeth, making breakfast) and write down the steps involved as a series of instructions, incorporating sequence, loops, and conditionals. They can even try translating these instructions into a short coding program using a beginner-friendly platform like Scratch or CodeCombat.
- Game Design Competition: Challenge students to design their own coding game based on a specific theme or genre. They can sketch out the game concept, plan the levels and challenges, and even outline the basic coding logic needed to bring their game idea to life.
- Code Breaker Puzzle: Present students with a simple text-based puzzle encoded with simple substitution ciphers or basic binary code. They can work individually or in pairs to decipher the code and reveal the hidden message.

Unplugged Activities:

- Board Game Design: Students can create their own board game with gameplay mechanics that involve logical thinking and sequential actions. They can design

the board, cards, game pieces, and rules, implementing concepts like dice rolls as random conditions, card draws as loops, and strategic decision-making based on game conditions.

- Robot Story Writing: Challenge students to write a short story featuring a robot character whose actions are controlled by a simple code. They can describe the robot's tasks, the code instructions it follows, and any challenges or opportunities it encounters due to its programmed logic.
- Treasure Hunt with Clues: Plan a small scavenger hunt around the classroom or schoolyard where students solve riddles or follow clues written in coded language (simple ciphers, symbols, binary code) to find hidden "treasure" objects. This encourages teamwork, problem-solving, and understanding basic coding principles.

Remember to choose activities that cater to your students' interests and abilities. Encourage them to explore different resources and platforms to further their coding journey, even if it's just creating simple animations or interactive stories. The key is to keep the learning process fun, engaging, and empowering, fueling their curiosity and fostering a love for logic and technology.