DRONEBLOCKS

LESSON 3: INTRODUCTION TO LOOPS

Overview:

Students will gain an understanding of the **loops** command and the importance of being able to repeat necessary commands when programming. Students will use DroneBlocks to program, test, and analyze the results of their missions. This lesson will give students a better understanding of commands within coding, while continuing to introduce them to the fundamental concepts of DroneBlocks. Extension lesson ideas are included to provide opportunities for students to add depth to their learning.

Objective:

Students will:

- Learn the importance of loops in programming and will practice what they learn to gain a better understanding of coding and programming in computer science
- Program their own mission, using the loop command
- View a test-flight of the program through Preview Flight
- Launch a field mission that will result in a successful mission
- Discuss and analyze their missions
- Troubleshoot to resolve problems or improve the mission

Common Core State Standards:

Common Core Standards provided focus on grades 5 and 6, as a baseline. Educators should adjust standards based on the ability and/or experience of their students.

CCSS.ELA-LITERACY.RST.6-8.3: Follow precisely a multistep procedure when carrying out experiments, taking measurements, or performing technical tasks.

CCSS.MATH.CONTENT.5.OA.A: Using operations and algebraic thinking, write and interpret numerical expressions.

CCSS.MATH.CONTENT.5.MD.A.1: Convert among different-sized standard measurement units within a given measurement system (e.g., convert 5 cm to 0.05 m), and use these conversions in solving multi-step, real world problems.

CCSS.MATH.CONTENT.6.G.A: Solve real-world and mathematical problems involving area, surface area, and volume.

CCSS.MATH.CONTENT.6.EE.A: Apply and extend previous understandings of arithmetic to algebraic expressions.

CCSS.ELA-LITERACY.RST.6-8.2: Determine the central ideas or conclusions of a text; provide an accurate summary of the text distinct from prior knowledge or opinions.

CCSS.ELA-LITERACY.RST.6-8.8: Distinguish among facts, reasoned judgment based on research findings, and speculation in a text.

International Society for Technology in Education Standards:

ISTE 1.a Apply existing knowledge to generate new ideas, products, or processes

ISTE 1.c Use models and simulations to explore complex systems and issues

ISTE 2.a Interact, collaborate, and publish with peers, experts, or others employing a variety of digital environments and media

ISTE 2.d Contribute to project teams to produce original work or solve problems

ISTE 3.d Process data and report results

ISTE 4.b Plan and manage activities to develop a solution or complete a project

Materials Needed:

- Access to internet to explore code.org
- Pre-Flight Checklist
- DroneBlocks App Downloaded on iOS devices for students
- Drone
- Propeller Guards

Part 1: What is a Loop?

Teach your students the definition of loop. Use real-world examples of loops, such as when you repeat a song on your playlist! Invite them to research or share their own thoughts as to why loops are a significant tool in coding.

Loops can be used to repeat a sequence of code. One of the basic and most useful functions in programming to repeatedly apply a section of code is the loop, which carries out a sequence of commands multiple times and shortens the code that is managed within the loop. A loop block will execute the commands inside the block a specified number of times and is used for repetitive tasks.

Invite your students to view this video of Mark Zuckerburg explaining the importance of looping, or repeating, commands: https://youtu.be/mgooqyWMTxk, then have your children code with Angry Birds: https://studio.code.org/s/course2/stage/6/puzzle/2. If you do not have computers or devices available for all students, demo the lesson for them. Tell your children they need to decide upon each command and tell you which block to add. Your classroom will become a buzz of students collaborating, coding, and problem solving!

Visit https://studio.code.org/ for more activities for you and your students. Code.org is one of our favorite organizations. Their websites include many activities, both digital and unplugged, to engage students of all ages and introduce them to programming and coding. For grasping the concept of looping in coding, have your students try Bee, Maze, Code with Ana and Elsa, or Artist. All of these online activities are free and located on www.Code.org and www.Studio.Code.org. They may be accessed on any computer or tablet.

Did you know? 9 out of 10 parents want their children to learn computer science but only 1 out of 4 schools teach computer science. DroneBlocks believes every child should have the opportunity to learn about programming, be introduced to how algorithms work, and what computer science means. We also believe that this opportunity should be available to girls and boys, students of all color, in any place in the world. Discuss why this is a very important topic. Remind students how fortunate they are to be able to engage in STEM education with DroneBlocks.

How can they share their knowledge? How can they make sure students in other places can be introduced to these skills? Research the “Computer Science for All” initiative. This is an excellent topic to discuss with your students and community…tweet about it, join an education forum, or ask students to write their thoughts! Don’t forget to join our discussion on the DroneBlocks Twitter feed, Facebook page, or Google Forum.

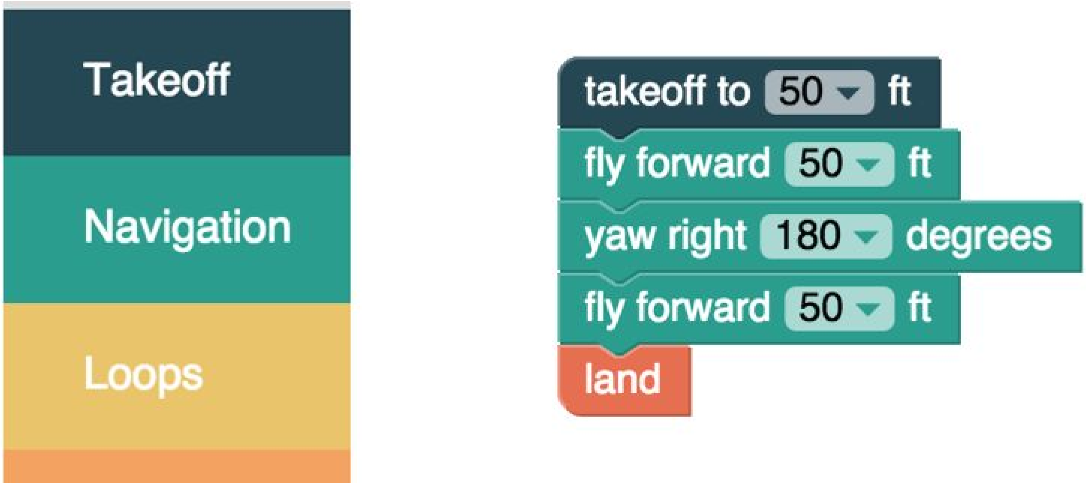
Part 2: The Mission Loop

This section will apply the concept of loops in programming DroneBlocks missions. Students will program the drone to travel forward and back repeatedly, but will use a loop command to reduce the amount of programming required.

Key components of the mission:

- Takeoff to altitude (recommend 50 feet or a height that is clear of nearby obstacles)
- Repeat the following 6 times:
  - Fly forward 50 feet
  - Yaw 180 degrees
- Land

To highlight the power of loop commands, without flying the mission, have the students initially program a mission that would fly out 50 feet and return (simple “out and back”). This will allow students to problem solve on a smaller scale before programming the full mission and will help identify the need for adding a loop command. **Figure 3.1** shows an example of this program.



```
takeoff to 50 ft
fly forward 50 ft
yaw right 180 degrees
fly forward 50 ft
land
```

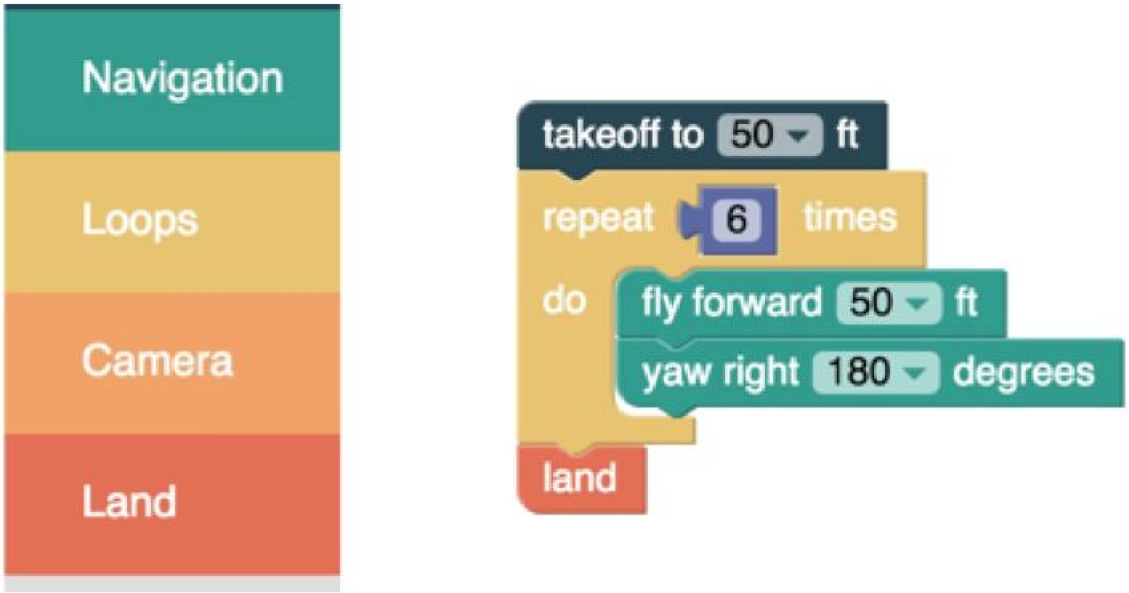
Figure 3.1

Once the “out and back” trial mission is finished, discuss the following: If you plan to make 6 trips, you must enter 12 commands. Ask, “**How can we problem-solve and program using fewer commands by implementing the loop block?**”

Students will now program their mission using loops. How many times do they need to repeat portions of their program to accomplish the mission?

Educator note: It may help some students to think of this as an order of operations math problem, while other students may benefit by being able to draw out the path for each line of code. Be mindful of different types of learners and allow students to share their ideas and thinking processes.

The final program should look similar to **Figure 3.2**:



```
takeoff to 50 ft
repeat 6 times
do  fly forward 50 ft
    yaw right 180 degrees
land
```

Figure 3.2

Part 3: Field Mission

Ask students to explain their missions and trial them with the Preview Mission block:

Preview Mission

REMEMBER:

- **When running preview missions indoors, the aircraft and remote should be powered off.**
- **Never let the students connect to the aircraft indoors, as they could accidentally activate the mission, resulting in an extremely dangerous indoor flight.**

Previewing missions allows you to see if your mission is programmed correctly. When programming you may forget a simple instruction. Seeing the code run on the preview simulator is a clear indicator of what needs to be fixed.

Now that the mission has been reviewed by the teacher and run in the mission preview, it is time to take the program to the field. Most of the hard work has already been done and now it is time to enjoy the rewards of programming and testing. Before running the mission at the field make sure to review the safety steps as you have done previously. These steps will apply to every field mission.

IMPORTANT:

- **Take a copy of the Pre-Flight Checklist to the flying area, and complete every step of the pre-flight checklist before running the mission.**

- **As the Phantom 3 is flying, always be prepared to manually take control of the aircraft if it behaves unexpectedly by switching back to P-mode. If operating a Phantom 4, manually resume control by switching to S-mode. Click here for a video discussing flight modes.**

- **Note: As tempting as it may be to take a photo directly beneath your drone, don't ever do this. You should not operate a drone above or within 25 feet of people under any circumstances.**

Your mission is now complete! Share your dronies with @droneblocks and tag us #droneblocks so that we can see your hard work. This is an opportunity to let the world see what you have programmed and created!

NOTE: As tempting as it may be to take a photo directly beneath your drone, don't ever do this. You should not operate a drone above or within 25 feet of people under any circumstances.

Part 4: Extensions

Challenge your students to build variations of this mission with additional steps within each block. Try having the aircraft fly out at one altitude (25 feet) and back at another altitude (50 feet). Or have them take a photo at the end of each leg.

Did you know? In the 1950's, a team of IBM programmers designed the first software to convert math formulas into code. This formula translation, or FORTRAN, was the first scientific language that was simple to use (in comparison to other machine codes of the time), was faster, and allowed programmers to focus on problem solving. Ask your students to research whether FORTRAN is still used today? If so, in what type of applications would this language best be utilized?

Additional Resources

Would you like to learn more about promoting computer science in education?

Start by visiting these websites:
https://code.org/promote.
https://www.whitehouse.gov/blog/2016/01/30/computer-science-all
http://www.csecoalition.org/

Would you like to learn more about programming or have students that are ready to learn to code?

Start by visiting these websites:
https://www.khanacademy.org/computing
http://www.edutopia.org/blog/teach-kids-coding-resources-parents-matt-davis
http://ocw.mit.edu/courses/intro-programming/

Disclaimer

The Contents of this guide are not necessarily complete and up-to-date and should not be used to replace any written report of DroneBlocks, statements, notices, or other materials provided by DroneBlocks. Educators, students, and other persons should use the contents in the same manner as any other educational medium and should not rely on the contents to the exclusion of their own judgment. Information obtained by using this guide is not exhaustive and not meant to be exhaustive and does not cover all issues, topics, or facts that may be relevant to your goals or particular needs.

You use DroneBlocks and this information at your own risk. Its Contents are provided “as is,” without warranties of any kind, either expressed or implied. DroneBlocks hereby disclaims all warranties, including but not limited any implied warranties. DroneBlocks does not represent or warrant that the functions or contents contained in the software and / or curriculum will be uninterrupted or error-free, that defects will be corrected. DroneBlocks does not warrant or make any representation regarding use, or the result of use, of the contents in terms of accuracy, reliability, or otherwise. The contents may include technical inaccuracies or typographical errors, and DroneBlocks may make changes or improvements at any time.

You, and not DroneBlocks, assume the entire cost of all necessary servicing, repair, or correction in the event of your loss or damage arising from the use of DroneBlocks. DroneBlocks makes no representations or warranties that your use of any of the contents on this guide will not infringe the rights of others, and DroneBlocks assumes no responsibility or liability for errors or omissions in any such contents. Any and all warranties that are implied by applicable law and are incapable of exclusion, restriction, or modification under applicable law are limited to the shortest period allowed by applicable law.