MINECRAFT
EDUCATION EDITION

Educator Guide

Minecraft Python Course – Lesson 5
45 minutes

A Helper For The Home

AGENT AND LOOPS

EDUCATION.MINECRAFT.NET

LESSON OBJECTIVES:

In this lesson the students will learn:

- the concept of loops.
- to use the for loop.
- the concept of, and how to use, nested loops.
- to use indentations.
- to code the Agent in Minecraft.

LESSON INTRODUCTION: 10 minutes

Theme:

Tell the students that they need to help CodingMine with the development of their new top-secret project called the Agent. The Agent is a robot that can be controlled using code to do a wide range of actions. The Agent is a companion that can help in many different situations ranging from housework to production in factories. Tell the students that they will be coding the Agent to do different jobs around a house in this lesson. Tell the students that they will be using different commands to control the Agent. These commands are agent move, agent place, agent collect all and agent drop all.

Coding Concepts:

Loops

Tell the students that in this lesson they will learn about **loops**. **Loops** are important when coding. Tell the students, **loops** repeat part of a code either continuously or for a certain number of times. Looping a part of a code also stops the need to type out the same piece of code many times. When coding in Python, loops are a great way to do some repetitive tasks and make your code smaller. Tell the students that in this lesson they will be using the for loop command to repeat part of a code a certain number of times. Tell the students that every loop is written with a **colon** at the end of the first line and the code that should be repeated is indented straight after. When using multiple loops, they should each have a unique name for their variable. By default, the for loops are named **i** or **index**.

Nested loops
When coding in Python a loop can be placed inside another, this is called nested loops. The number of nested loops can be more than two. When coding, nested loops are a great way to do more complicated tasks.

User Interface:

In this lesson the students will:

- see displayed world coordinates in the top left-hand corner.
- see displayed Agent world coordinates in the top left-hand corner.

Syntax / Operators:

Tell the students in this lesson they will learn about:

Colon :
Colon is a symbol used at the end of a command declaration. After the **colon** the code that precedes it is indented as it acts as a part of that command.

Indentation :
Indentations in Python are used when we want to declare that a piece of code belongs to the command above. To indent a piece of code, use the tab key on the keyboard. Example:

```
for i in range(2):
    agent.move(FORWARD, 1)
```

CODING ACTIVITIES: 30 minutes

Activity 1: Too heavy.

Objective: Explain to the students that the developer needs their help to code the Agent to make it lift heavy objects around the house. This function of the Agent is going to greatly help people who cannot move such objects without injuring themselves. Tell the students that they will do this Activity in two parts. Firstly, the students need to code the Agent to move one heavy box. Then they need to code the Agent, using a loop, to move multiple heavy boxes.

Part 1: Tell the students to firstly to code the Agent to collect one box and then place it in the marked area. *(Hint: The Agent does not have to face the item to collect it).* When the students run the code, the Agent will pick up the box, move forwards and place it.

Part 2: Now tell the students that they need to make the Agent do the same thing, but for multiple boxes. *(Hint: Tell the students to make sure that the code that they want repeated is indented).* When the students run the code, the Agent will pick up and move four boxes that will then be stacked on top of each other automatically. The students have now finished Activity 1.

Code snippets:

Part 1:

```
agent.collect_all()
agent.move(FORWARD, 6)
agent.place(FORWARD)
```

Part 2:

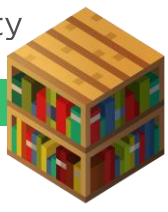
```
for i in range(4):
    agent.collect_all()
    agent.move(FORWARD, 6)
    agent.place(FORWARD)
    agent.move(BACK, 6)
```

Activity 2: Spin cycle.

Objective: Explain to the students that the developer needs their help to code the Agent to make it do laundry. They will have to code the Agent to pick up dirty laundry, put it in the washing machine and then get the Agent to spin around to wash the clothes using a loop. The Agent should then place the clean clothes in a pile. The students will carry out this Activity in two parts, the second being coding the Agent to do multiple loads of laundry using a second loop.

Part 1: Tell the students that they firstly need to make the Agent do one load of laundry. When the students run the code, the Agent will take the dirty laundry, go into the washing machine, wash it, go back out and place it on the clean area.

Part 2: Now tell the students that they need to make the Agent do the same thing but for multiple loads of laundry. *(Hint: You can rename a loop to anything that you want but the name must not be the same as another loop).* When the students run the code, the Agent will do three loads of dirty laundry. After this, Activity 2 is complete.



Code snippets:

Before:

```
# Replace the lines below with your code #
# loop number 2 set to 3                                  | Part 2
agent.collect_all()
agent.move(FORWARD, 7)
agent.drop_all(FORWARD)
# loop number 1                                 | Part 1
# make the Agent turn left 20 times           | Part 1
# end of loop 1
# make the Agent collect all                    | Part 1
# make the Agent move back                      | Part 1
# make the Agent drop everything to the left | Part 1
# end of loop 2
```

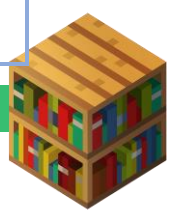
After:

Part 1:

```
# replace with loop number 2 set to 3         | Part 2
agent.collect_all()
agent.move(FORWARD, 7)
agent.drop_all(FORWARD)
for i1 in range(20):
    agent.turn(LEFT_TURN)
agent.collect_all()
agent.move(BACK, 7)
agent.drop_all(LEFT)
# end of loop 2
```

Part 2:

```
for i2 in range(3):
    agent.collect_all()
    agent.move(FORWARD, 7)
    agent.drop_all(FORWARD)
    for i1 in range(20):
        agent.turn(LEFT_TURN)
    agent.collect_all()
    agent.move(BACK, 7)
    agent.drop_all(LEFT)
```



Activity 3: Cleaning up.

Objective: Explain to the students that the developer needs their help to code the Agent to make it vacuum carpets. Tell them that this activity will be done in two parts. They first have to code the Agent to vacuum a small carpet and in the second part they need to code the Agent to vacuum a large carpet and throw the dirt in the trash. *(Hint: Do not forget that loops cannot have the same name and cannot be named a number).*

Part 1: Tell the students to start by coding the Agent to vacuum the smaller carpet. When the students run the code, the Agent will move across the carpet and vacuum.

Part 2: Now tell the students that they need to make the Agent do the same thing but for the larger carpet. When the Agent finished cleaning the carpet and drops all the dirt in the trash the students will have completed Activity 3 and the lesson.

Code snippets:

Before:

```
# Replace the lines below with your code #
# loop number 3                                    | Part 2
# loop number 1                          | Part 1
agent.collect_all()
agent.move(FORWARD, 1)
# end of loop 1
agent.move(RIGHT, 1)
# loop number 2                          | Part 1
# make the Agent collect all             | Part 1
# make the Agent move back               | Part 1
# end of loop 2
# make the Agent move right                        | Part 2
# end of loop 3
# make the Agent drop all to the right             | Part 2
```

After:

Part 1:

```
# replace with loop number 3           | Part 2
for i1 in range(7):
    agent.collect_all()
    agent.move(FORWARD, 1)
agent.move(RIGHT, 1)
for i2 in range(7):
    agent.collect_all()
    agent.move(BACK, 1)
# make the Agent move right            | Part 2
# end of loop 3
# make the Agent drop all to the right | Part 2
```

Part 2:

```
for i3 in range(3):
    for i1 in range(7):
        agent.collect_all()
        agent.move(FORWARD, 1)
    agent.move(RIGHT, 1)
    for i2 in range(7):
        agent.collect_all()
        agent.move(BACK, 1)
    agent.move(RIGHT, 1)
agent.drop_all(FORWARD)
```

LESSON CONCLUSION: 5 minutes

Ask the students about the skills that they have learned during the lesson, to reinforce the concepts.

1. Q. What is a for loop?
   A. A for loop is a coding command that repeats a piece of code a certain number of times.
2. Q. What do you use indentations with loops for?

A. To show what code needs to be looped.

3 Q. How do we make the Agent move in a certain direction?

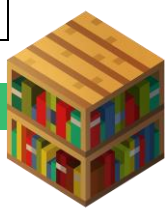
A. Using the agent move command.

4. Q. What is a nested loop?

A A nested loop is a loop inside another loop.

EDUCATION STANDARDS:

CSTA K-12	
1A-AP-10	Develop programs with sequences and simple loops, to express ideas or address a problem.
1B-AP-11	Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.
1B-AP-15	Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended
ISTE	
7A	Provide alternative ways for students to demonstrate competency and reflect on their learning using technology.
3B	Establish a learning culture that promotes curiosity and critical examination of online resources and fosters digital literacy and media fluency.
6B	Manage the use of technology and student learning strategies in digital platforms, virtual environments, hands-on makerspaces or in the field.



COMMANDS:

For loop

```
for i in range(4):
    pass
```

for

Description: Repeat a piece of code for a specified number of times.

name parameter: Loops individual name.

repeat parameter: How many times the loop should repeat

Agent move

```
Agent.move(FORWARD, 1)
```

agent move **direction** by **1**

Description: Tells the Agent to move in a specified direction by a specified number of blocks.

direction parameter: What direction the Agent should move in.

block parameter: The number of blocks the Agent should move.

Agent turn

```
agent.turn(LEFT_TURN)
```

agent turn **TurnDirection.Left**

Description: Tells the Agent to turn in a specified direction.

turn direction parameter: What direction the Agent should turn in.

Agent place

```
agent.place(FORWARD)
```

agent place **direction**

Description: Tells the Agent to place a block or item in a specified direction.

direction parameter: What direction the Agent should place a block in.

Agent collect all

```
agent.collect_all()
```

agent collect all

Description: Tells the Agent to collect all items that are near.

Agent drop all

```
agent.drop_all(FORWARD)
```

agent drop all **direction**

Description: Tells the Agent to drop its whole inventory in a specified direction.

direction parameter: In what direction the Agent should drop its inventory.