Course: Python Programming 1 & 2 I Module: Karel in Python

Lesson 1.4: Functions in Karel

https://codehs.com/course/21076/lesson/1.4

Description	In this lesson, functions will be used to teach Karel a new word or command. Using functions allows programs to be broken down into smaller pieces and makes it easier to understand.
Objective	Students will be able to: • Understand what functions are, how they are used and how using them improves programs • Design and implement their own functions to solve problems
Activities	[1.4.1 Video: Functions in Karel] [1.4.2 Check for Understanding: Functions in Karel Quiz] [1.4.3 Example: Turn Around] [1.4.4 Exercise: Pancakes] [1.4.5 Exercise: Backflip] [1.4.6 Exercise: Digging Karel] [1.4.7 Debugging: Build a Shelter]
Prior Knowledge	• Karel's four basic commands • Defining and calling functions • Basic understanding of the sequential order of commands • Proper syntax for typing commands
Planning Notes	• Review the slides and the exercises in the lesson. • Find a video of a line or step dance (electric slide, cupid shuffle, macarena, etc) that will work for the opening activity (or find a student who would like to volunteer). • There is a handout that accompanies this lesson. It can be used as an in-class activity or a homework assignment. Determine how and if this handout will be used and make the appropriate number of printouts prior to the class period.
Standards Addressed	
Teaching and Learning Strategies	**Lesson Opener:** • *Dance Commands:* Have students watch a video of a line or step dance (electric slide, cupid shuffle, macarena, etc) or have a student volunteer to perform one! Direct students to write down each individual step to the dance as a command. Do any steps repeat? Could the steps be combined? Have the students create functions to combine dance steps. They should continue to combine functions as best as they can to keep simplifying the dance. Can they get the dance to just a few commands? *Optional:* Use the *Dancing with Functions* handout. [15 mins]

```
def to_the_left():
    step_left()
    step_left()
    step_left()
    step_left()

def to_the_right():
    step_right()
    step_right()
    step_right()
    step_right()

def now_kick():
    kick_right_foot()
    kick_left_foot()
    kick_right_foot()
    kick_left_foot()

# this one is hard!
def walk_it_by_yourself():
    twist_your_knees_while_turning_left()

# Cupid Shuffle Example:
to_the_left()
to_the_right()
now_kick()
walk_it_by_yourself()
```

- Have students brainstorm and write down answers to the discussion questions listed below. Students can work individually or in groups/pairs. Have them share their responses. [5 mins]

Activities:

- Watch the lesson video and complete the corresponding quiz. [7-10 mins]
- Let students play around with the *Turn Around* example. [2 mins]
- Complete the *Pancakes* exercise. [10 mins]
- Complete the *Backflip* exercise. [10 mins]
- Complete the *Digging Karel* exercise. [15 mins]
- Complete the *Build a Shelter* exercise. [5 mins]
- Complete the *Debugging Functions* handout. [15 mins]

Lesson Closer:

- Have students reflect and discuss their responses to the end of class discussion questions. [5 mins]
- Complete *Naming Functions* handout. [5 mins]

Discussion Questions

Beginning of Class:

- What is the difference between defining and calling a function?
    - *Defining a function is teaching the computer (or Karel) how to execute the command. Calling the function is telling the computer (or Karel) to actually perform the function task.*

End of Class:

- List at least 3 ways functions are helpful for writing and reading programs.
    - *Functions simplify your code by avoiding repetition. Functions make it easier for others to read your code. Functions are reusable and make writing your code more efficient.*
- Write out a function for an everyday task that you perform, and break down the instructions inside the function into even smaller functions, for example:

```
def eat_sandwich():
    bring_sandwich_to_mouth()
    bite_down()
    chew()

def bring_sandwich_to_mouth():
    place_sandwich_in_between_hands()
    move_arms_in_45_degree_angle()
```

Resources/Handouts

[Debugging Functions (student)]

	Debugging Functions (teacher) Naming Functions (Student) Naming Functions (Teacher) Dancing with Functions (student) Dancing with Functions (teacher)

Vocabulary

Term	Definition
Define a Function	Defining a function means to teach the computer a new command and explain what it should do when receiving that command.
Call a Function	Calling a function actually gives the command, so the computer will run the code for that function.
Function Body	In Python functions, the function body is the indented block of code that comes after the `def my_function():` line. The function body is what will be executed when the function is called.
Snake Case	`snake case` refers to the style of writing in which each space is replaced by an underscore _ character and the first letter of each word is lowercase.

Modification: Advanced	Modification: Special Education	Modification: English Language Learners
• Have students create an original Sandbox program that defines and calls a minimum of 4 (four) functions.	• Pair programming with another student • Print out video slides for students to reference	• Complete a flowchart that diagrams how a function works in a program.