

Course: Intro to Computer Science in Python 3 | Module: Introduction to Programming with Turtle Graphics



Lesson 2.1: Intro to Python with Tracy the Turtle

<https://codehs.com/course/5657/lesson/2.1>

| | |
|---|---|
| Description | <p>In this lesson, students are introduced to coding with turtle graphics. Students will begin to recognize programs as sequences and groups of commands. Students will learn a few basic commands and then apply them right away by writing their first program.</p> |
| Objective | <p>Students will be able to:</p> <ul style="list-style-type: none"> • Define programming/coding • Use basic Tracy commands • Write their first program |
| Activities | <p>2.1.1 Video: Intro to Tracy. 2.1.2 Quiz: Intro to Tracy. 2.1.3 Example: Slinky. 2.1.4 Exercise: Stretched Slinky.</p> |
| Prior Knowledge | <ul style="list-style-type: none"> • Directionality- forward, backward • Parentheses- must be able to locate and type on a keyboard • Typos- incorrect spelling, capitalization, or character • Geometry of a circle- difference between radius and diameter |
| Planning Notes | <ul style="list-style-type: none"> • Decide if students will take notes in a notebook, on paper handouts, or through the “Take Notes” function on CodeHS. • Decide if students will keep a running list of learned commands in a notebook or on a paper handout from the lesson recap slide. • Provide time before lesson to have students set up student accounts and become familiar with CodeHS web navigation. • There is a handout that accompanies this lesson. It can be used as an in class activity or a homework assignment. Determine if and how this handout will be used and make the appropriate number of printouts prior to the class period. |
| Standards Addressed | |
| Teaching and Learning Strategies | <p>Lesson Opener:</p> |

- 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:

- Discuss with class if a dog is a computer and what would make it one based on their definition of what a computer is. Pair share what the difference is between a computer and a dog. (Teaching point: Students should understand that a computer *must follow* the commands it is being given, while a dog can *choose to follow* your command or not.) [5 mins]
- *CodeHS Tour* [10 mins]
 - Have students log into CodeHS with their usernames and passwords. It is suggested that a printed copy of student login info be handy in case some students forget their credentials.
 - Demonstrate to students an e-tour of the class page that will be used throughout the course.
- Allow students to watch the introductory video individually or watch together as a class. If needed students can take notes for later reference. Inform students that there will be a quiz after the video. [5-7 mins]
- Direct students to view the example. [5-7 mins]
 - Tell students to note how each command is written.
 - Encourage students to experiment with the results of changing the order of the instructions within the example.
 - In addition, encourage students to experiment with what happens if commands are capitalized, parenthesis are left off, or indentation is incorrect.
- Have students complete the *Stretched Slinky* exercise. [5-10 mins]
 - If there are students that are experiencing trouble with the program, allow them to examine a working program from a fellow student and discover their error.
- Students can complete the *Circles* handout activity in pairs or individually if time permits, or for homework. [7-10 mins]

Lesson Closer:

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

Discussion Questions

Beginning of Class:

- What is a computer?
 - *A computer is something that completes tasks we give to it in a language it understands.*
- Look around the room. Write down all the computers you see.
 - *Laptop, Chromebook, Calculator, etc. (Answers may vary)*

End of Class:

- What is coding?
 - *Giving commands to a computer*

| | |
|---------------------------|--|
| | <ul style="list-style-type: none"> • What do you do when you receive an error message? <ul style="list-style-type: none"> ◦ <i>Check for typos</i> • What type of mistakes did you find yourself making? <ul style="list-style-type: none"> ◦ <i>Spelling typo, capitalization typo, etc. (Answers may vary)</i> • How did you approach your first attempt at a program? <ul style="list-style-type: none"> ◦ <i>(Answers may vary)</i> ◦ Discuss those who started writing code immediately vs. planned it out first. Emphasize that planning will become more and more necessary as programs get more complex and difficult. |
| Resources/Handouts | <p>Circles (teacher)</p> <p>Circles (student)</p> |

Vocabulary

| Term | Definition |
|-----------------------------------|--|
| forward(a_number) | Command that lets you tell Tracy to move forward. In between the parentheses you need to put a number to tell Tracy how far to move forward. |

| Modification: Advanced | Modification: Special Education | Modification: English Language Learners |
|------------------------|---------------------------------|---|
| | | |