

# Fiero Code Curriculum

## CSTA & ISTE Standards

Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>Javascript</b></p> <p><b>Skills Mastery:</b> Learn how to create and use event handlers, event listeners, web animations, input fields, sequences, variables, conditionals, loops, and functions</p>	<p><b>2-AP-11</b> Create clearly named variables that represent different data types and perform operations on their values.</p> <p><b>2-AP-12</b> Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.</p> <p><b>2-AP-13</b> Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.</p> <p><b>2-AP-14</b> Create procedures with parameters to organize code and make it easier to reuse.</p> <p><b>2-AP-15</b> Seek and incorporate feedback from team members and users to refine a solution that meets user needs.</p> <p><b>2-AP-16</b> Incorporate existing code, media, and libraries into original programs, and give attribution.</p> <p><b>2-AP-17</b> Systematically test and refine programs using a range of test cases.</p> <p><b>2-AP-19</b> Document programs in order to make them easier to follow, test, and debug.</p> <p><b>3A-AP-14</b> Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.</p> <p><b>3A-AP-16</b> Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.</p> <p><b>3A-AP-21</b> Evaluate and refine computational artifacts to make them more usable and accessible.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.1.c</b> Use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways.</p> <p><b>1.2.c</b> Demonstrate an understanding of and respect for the rights and obligations of using and sharing intellectual property.</p> <p><b>1.4.c</b> Develop, test, and refine prototypes as part of a cyclical design process.</p> <p><b>1.5.a</b> Formulate problem definitions suited for technology assisted methods such as data analysis, abstract models and algorithmic thinking in exploring and finding solutions.</p> <p><b>1.5.c</b> Break problems into component parts, extract key information, and develop descriptive models to understand complex systems to facilitate problem-solving.</p> <p><b>1.5.d</b> Understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.</p> <p><b>1.6.b</b> Create original works or responsibly repurpose or remix digital resources into new creations..</p> <p><b>1.6.c</b> Communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations models or simulations.</p> <p><b>1.6.d</b> Publish and present content that customizes the message and medium for their intended audience.</p>

Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>Python</b></p> <p><b>Skills Mastery:</b> Learn how to create sequences, variables, loops, conditionals, and functions</p>	<p><b>2-AP-11</b> Create clearly named variables that represent different data types and perform operations on their values.</p> <p><b>2-AP-12</b> Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.</p> <p><b>2-AP-13</b> Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.</p> <p><b>2-AP-14</b> Create procedures with parameters to organize code and make it easier to reuse.</p> <p><b>2-AP-15</b> Seek and incorporate feedback from team members and users to refine a solution that meets user needs.</p> <p><b>2-AP-16</b> Incorporate existing code, media, and libraries into original programs, and give attribution.</p> <p><b>2-AP-17</b> Systematically test and refine programs using a range of test cases.</p> <p><b>2-AP-19</b> Document programs in order to make them easier to follow, test, and debug.</p> <p><b>3A-AP-14</b> Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.</p> <p><b>3A-AP-16</b> Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.</p> <p><b>3A-AP-21</b> Evaluate and refine computational artifacts to make them more usable and accessible.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.1.c</b> Use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways.</p> <p><b>1.2.c</b> Demonstrate an understanding of and respect for the rights and obligations of using and sharing intellectual property.</p> <p><b>1.4.c</b> Develop, test, and refine prototypes as part of a cyclical design process.</p> <p><b>1.5.c</b> Break problems into component parts, extract key information, and develop descriptive models to understand complex systems to facilitate problem-solving.</p> <p><b>1.6.b</b> Create original works or responsibly repurpose or remix digital resources into new creations.</p> <p><b>1.6.c</b> Communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations models or simulations.</p> <p><b>1.6.d</b> Publish and present content that customizes the message and medium for their intended audience.</p>

Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>Code Blocks</b></p> <p><b>Skills Mastery:</b> Learn how to create and use sequences, variables, loops, conditionals and events as you create a platformer game, and practice remixing existing projects.</p>	<p><b>1B-AP-09</b> Create programs that use variables to store and modify data.</p> <p><b>1B-AP-10</b> Create programs that include sequences, events, loops, and conditionals.</p> <p><b>1B-AP-11</b> Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.</p> <p><b>1B-AP-12</b> Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.</p> <p><b>1B-AP-13</b> Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences.</p> <p><b>1B-AP-14</b> Observe intellectual property rights and give appropriate attribution when creating or remixing programs.</p> <p><b>1B-AP-15</b> Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended.</p> <p><b>1B-AP-17</b> Describe choices made during program development using code comments, presentations, and demonstrations.</p> <p><b>1B-IC-20</b> Seek diverse perspectives for the purpose of improving computational artifacts.</p> <p><b>1B-IC-21</b> Use public domain or creative commons media, and refrain from copying or using material created by others without permission.</p> <p><b>2-AP-11</b> Create clearly named variables that represent different data types and perform operations on their values.</p> <p><b>2-AP-12</b> Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.</p> <p><b>2-AP-13</b> Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.1.c</b> Use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways.</p> <p><b>1.2.c</b> Demonstrate an understanding of and respect for the rights and obligations of using and sharing intellectual property.</p> <p><b>1.4.a</b> Know and use a deliberate design process for generating ideas, testing theories, creating innovative artifacts, or solving authentic problems.</p> <p><b>1.4.c</b> Develop, test, and refine prototypes as part of a cyclical design process.</p> <p><b>1.5.c</b> Break problems into component parts, extract key information, and develop descriptive models to understand complex systems to facilitate problem-solving.</p> <p><b>1.6.b</b> Create original works or responsibly repurpose or remix digital resources into new creations.</p> <p><b>1.6.c</b> Communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations models or simulations.</p> <p><b>1.6.d</b> Publish and present content that customizes the message and medium for their intended audience.</p>

**2-AP-14** Create procedures with parameters to organize code and make it easier to reuse.

**2-AP-15** Seek and incorporate feedback from team members and users to refine a solution that meets user needs.

**2-AP-16** Incorporate existing code, media, and libraries into original programs, and give attribution.

**2-AP-17** Systematically test and refine programs using a range of test cases.

**2-AP-19** Document programs in order to make them easier to follow, test, and debug.

**3A-AP-14** Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.

**3A-AP-16** Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.

**3A-AP-21** Evaluate and refine computational artifacts to make them more usable and accessible.

Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>Sphero</b></p> <p>Learn basic programming concepts such as loops, conditionals and functions.</p> <p>Learn how hardware interacts with software.</p>	<p><b>2-AP-11</b> Create clearly named variables that represent different data types and perform operations on their values.</p> <p><b>2-AP-13</b> Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.</p> <p><b>2-AP-15</b> Seek and incorporate feedback from team members and users to refine a solution that meets user needs.</p> <p><b>2-AP-17</b> Systematically test and refine programs using a range of test cases.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.1.c</b> Use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways.</p> <p><b>1.4.c</b> Develop, test, and refine prototypes as part of a cyclical design process.</p> <p><b>1.4.d</b> Exhibit a tolerance for ambiguity, perseverance and the capacity to work with open-ended problems.</p> <p><b>1.5.c</b> Break problems into component parts, extract key information, and develop descriptive models to understand complex systems to facilitate problem-solving.</p> <p><b>1.6.b</b> Create original works or responsibly repurpose or remix digital resources into new creations.</p> <p><b>1.6.c</b> Communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations models or simulations.</p> <p><b>1.6.d</b> Publish and present content that customizes the message and medium for their intended audience.</p>
Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>Ozobot</b></p> <p>Learn basic programming concepts like loops, conditionals and functions. Also</p>	<p><b>1B-AP-09</b> Create programs that use variables to store and modify data.</p> <p><b>1B-AP-10</b> Create programs that include sequences, events, loops, and conditionals.</p> <p><b>1B-AP-11</b> Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.1.c</b> Use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways.</p> <p><b>1.4.c</b> Develop, test, and refine prototypes as part of a cyclical design process.</p>

learn how hardware interacts with software.

**1B-AP-12** Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.

**1B-AP-13** Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences.

**1B-AP-14** Observe intellectual property rights and give appropriate attribution when creating or remixing programs.

**2-AP-11** Create clearly named variables that represent different data types and perform operations on their values.

**2-AP-13** Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.

**2-AP-15** Seek and incorporate feedback from team members and users to refine a solution that meets user needs.

**2-AP-17** Systematically test and refine programs using a range of test cases.

**1.4.d** Exhibit a tolerance for ambiguity, perseverance and the capacity to work with open-ended problems.

**1.5.c** Break problems into component parts, extract key information, and develop descriptive models to understand complex systems to facilitate problem-solving.

**1.6.b** Create original works or responsibly repurpose or remix digital resources into new creations.

**1.6.c** Communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations models or simulations.

**1.6.d** Publish and present content that customizes the message and medium for their intended audience.

Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>JS Web</b></p> <p>Learn how to access the contents of a web page with Javascript through the DOM API. Create interactive events and animations as you create a website project.</p>	<p><b>2-AP-11</b> Create clearly named variables that represent different data types and perform operations on their values.</p> <p><b>2-AP-12</b> Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.</p> <p><b>2-AP-13</b> Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.</p> <p><b>2-AP-14</b> Create procedures with parameters to organize code and make it easier to reuse.</p> <p><b>2-AP-15</b> Seek and incorporate feedback from team members and users to refine a solution that meets user needs.</p> <p><b>2-AP-16</b> Incorporate existing code, media, and libraries into original programs, and give attribution.</p> <p><b>2-AP-17</b> Systematically test and refine programs using a range of test cases.</p> <p><b>2-AP-19</b> Document programs in order to make them easier to follow, test, and debug.</p> <p><b>3A-AP-14</b> Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.</p> <p><b>3A-AP-16</b> Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.</p> <p><b>3A-AP-21</b> Evaluate and refine computational artifacts to make them more usable and accessible.</p> <p><b>3B-AP-15</b> Analyze a large-scale computational problem and identify generalizable patterns that can be applied to a solution.</p>	<p><b>1.1a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.1.c</b> Use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways.</p> <p><b>1.2.c</b> Demonstrate an understanding of and respect for the rights and obligations of using and sharing intellectual property.</p> <p><b>1.3.d</b> Build knowledge by actively exploring real-world issues and problems, developing ideas and theories and pursuing answers and solutions.</p> <p><b>1.4.c</b> Develop, test, and refine prototypes as part of a cyclical design process.</p> <p><b>1.6.b</b> Create original works or responsibly repurpose or remix digital resources into new creations.</p> <p><b>1.6.d</b> Publish and present content that customizes the message and medium for their intended audience.</p>

Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>Animation</b></p> <p>Learn basic web animations using Javascript as you create a marketing website.</p>	<p><b>2-AP-11</b> Create clearly named variables that represent different data types and perform operations on their values.</p> <p><b>2-AP-12</b> Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.</p> <p><b>2-AP-13</b> Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.</p> <p><b>2-AP-14</b> Create procedures with parameters to organize code and make it easier to reuse.</p> <p><b>2-AP-17</b> Systematically test and refine programs using a range of test cases.</p> <p><b>2-AP-19</b> Document programs in order to make them easier to follow, test, and debug.</p> <p><b>3A-AP-14</b> Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.</p> <p><b>3A-AP-16</b> Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.</p> <p><b>3B-AP-15</b> Analyze a large-scale computational problem and identify generalizable patterns that can be applied to a solution.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.1.c</b> Use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways.</p> <p><b>1.4.c</b> Develop, test, and refine prototypes as part of a cyclical design process.</p> <p><b>1.4.d</b> Exhibit a tolerance for ambiguity, perseverance and the capacity to work with open-ended problems.</p> <p><b>1.5.c</b> Break problems into component parts, extract key information, and develop descriptive models to understand complex systems to facilitate problem-solving.</p> <p><b>1.6.b</b> Create original works or responsibly repurpose or remix digital resources into new creations.</p> <p><b>1.6.c</b> Communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations models or simulations.</p> <p><b>1.6.d</b> Publish and present content that customizes the message and medium for their intended audience.</p>

Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>Code Blocks Games</b></p> <p>Use variables, loops, functions, conditionals, and events as you create three different games in Scratch.</p>	<p><b>1B-AP-09</b> Create programs that use variables to store and modify data.</p> <p><b>1B-AP-10</b> Create programs that include sequences, events, loops, and conditionals.</p> <p><b>1B-AP-11</b> Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.</p> <p><b>1B-AP-12</b> Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.</p> <p><b>1B-AP-13</b> Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences.</p> <p><b>1B-AP-14</b> Observe intellectual property rights and give appropriate attribution when creating or remixing programs.</p> <p><b>1B-AP-15</b> Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended.</p> <p><b>1B-AP-17</b> Describe choices made during program development using code comments, presentations, and demonstrations.</p> <p><b>1B-IC-20</b> Seek diverse perspectives for the purpose of improving computational artifacts.</p> <p><b>1B-IC-21</b> Use public domain or creative commons media, and refrain from copying or using material created by others without permission.</p> <p><b>2-AP-11</b> Create clearly named variables that represent different data types and perform operations on their values.</p> <p><b>2-AP-12</b> Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.</p> <p><b>2-AP-13</b> Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.1.c</b> Use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways.</p> <p><b>1.2.c</b> Demonstrate an understanding of and respect for the rights and obligations of using and sharing intellectual property.</p> <p><b>1.4.a</b> Know and use a deliberate design process for generating ideas, testing theories, creating innovative artifacts, or solving authentic problems.</p> <p><b>1.5.c</b> Break problems into component parts, extract key information, and develop descriptive models to understand complex systems to facilitate problem-solving.</p> <p><b>1.6.b</b> Create original works or responsibly repurpose or remix digital resources into new creations.</p> <p><b>1.6.c</b> Communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations models or simulations.</p> <p><b>1.6.d</b> Publish and present content that customizes the message and medium for their intended audience.</p>

**2-AP-14** Create procedures with parameters to organize code and make it easier to reuse.

**2-AP-15** Seek and incorporate feedback from team members and users to refine a solution that meets user needs.

**2-AP-16** Incorporate existing code, media, and libraries into original programs, and give attribution.

**2-AP-17** Systematically test and refine programs using a range of test cases.

**2-AP-19** Document programs in order to make them easier to follow, test, and debug.

**3A-AP-14** Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.

**3A-AP-16** Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.

**3A-AP-21** Evaluate and refine computational artifacts to make them more usable and accessible.

**3B-AP-09** Implement an artificial intelligence algorithm to play a game against a human opponent or solve a problem.

Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>Drawing</b></p> <p>Learn how to use the HTML Canvas to draw on a webpage with Javascript, and animate your drawings as you create an artistic webpage.</p>	<p><b>2-AP-11</b> Create clearly named variables that represent different data types and perform operations on their values.</p> <p><b>2-AP-12</b> Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.</p> <p><b>2-AP-13</b> Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.</p> <p><b>2-AP-14</b> Create procedures with parameters to organize code and make it easier to reuse.</p> <p><b>2-AP-17</b> Systematically test and refine programs using a range of test cases.</p> <p><b>2-AP-19</b> Document programs in order to make them easier to follow, test, and debug.</p> <p><b>3A-AP-14</b> Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.</p> <p><b>3A-AP-16</b> Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.</p> <p><b>3B-AP-15</b> Analyze a large-scale computational problem and identify generalizable patterns that can be applied to a solution.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.2.b</b> engage in positive, safe, legal, and ethical behavior when using technology, including social interactions online or when using networked devices.</p> <p><b>1.2.c</b> Demonstrate an understanding of and respect for the rights and obligations of using and sharing intellectual property.</p> <p><b>1.4.a</b> Know and use a deliberate design process for generating ideas, testing theories, creating innovative artifacts, or solving authentic problems.</p> <p><b>1.4.c</b> Develop, test, and refine prototypes as part of a cyclical design process.</p> <p><b>1.4.d</b> Exhibit a tolerance for ambiguity, perseverance and the capacity to work with open-ended problems.</p> <p><b>1.5.c</b> Break problems into component parts, extract key information, and develop descriptive models to understand complex systems to facilitate problem-solving.</p> <p><b>1.5.d</b> Understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.</p> <p><b>1.6.c</b> Communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations models or simulations.</p>

Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>JS Games</b></p> <p>Use variables, loops, functions, conditionals, methods, and debugging skills as you build four classic arcade games.</p>	<p><b>2-AP-11</b> Create clearly named variables that represent different data types and perform operations on their values.</p> <p><b>2-AP-12</b> Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.</p> <p><b>2-AP-13</b> Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.</p> <p><b>2-AP-14</b> Create procedures with parameters to organize code and make it easier to reuse.</p> <p><b>2-AP-15</b> Seek and incorporate feedback from team members and users to refine a solution that meets user needs.</p> <p><b>2-AP-16</b> Incorporate existing code, media, and libraries into original programs, and give attribution.</p> <p><b>2-AP-17</b> Systematically test and refine programs using a range of test cases.</p> <p><b>2-AP-19</b> Document programs in order to make them easier to follow, test, and debug.</p> <p><b>3A-AP-14</b> Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.</p> <p><b>3A-AP-16</b> Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.</p> <p><b>3A-AP-21</b> Evaluate and refine computational artifacts to make them more usable and accessible.</p> <p><b>3B-AP-09</b> Implement an artificial intelligence algorithm to play a game against a human opponent or solve a problem.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.1.d</b> Understand the fundamental concepts of technology operations, demonstrate the ability to choose, use and troubleshoot current technologies and are able to transfer their knowledge their to explore emerging technologies.</p> <p><b>1.4.c</b> Develop, test, and refine prototypes as part of a cyclical design process.</p> <p><b>1.4.d</b> Exhibit a tolerance for ambiguity, perseverance and the capacity to work with open-ended problems.</p> <p><b>1.5.d</b> Understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.</p> <p><b>1.6.c</b> Communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations models or simulations.</p> <p><b>1.5.d</b> Understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.</p>
Fiero Course Unit	CSTA Standards	ISTE Standards

<p><b>SQL</b></p> <p>Learn about databases, create and edit tables, and run queries as you gain the skills needed to isolate a single piece of data from a large data set.</p>	<p><b>2-DA-07</b> Represent data using multiple encoding schemes.</p> <p><b>2-DA-08</b> Collect data using computational tools and transform the data to make it more useful and reliable.</p> <p><b>2-DA-09</b> Refine computational models based on the data they have generated.</p> <p><b>3A-DA-09</b> Translate between different bit representations of real-world phenomena such as characters, numbers, and images.</p> <p><b>3A-DA-10</b> Evaluate the tradeoffs in how data elements are organized and where data is stored.</p> <p><b>3A-DA-11</b> Create interactive data visualizations using software tools to help others better understand real-world phenomena.</p> <p><b>3A-DA-12</b> Create computational models that represent the relationships among different elements of data collected from a phenomenon or process.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.1.d</b> Understand the fundamental concepts of technology operations, demonstrate the ability to choose, use and troubleshoot current technologies and are able to transfer their knowledge their to explore emerging technologies.</p> <p><b>1.5.a</b> Formulate problem definitions suited for technology assisted methods such as data analysis, abstract models and algorithmic thinking in exploring and finding solutions.</p> <p><b>1.5.b</b> Collect data or identify relevant data sets, use digital tools to analyze them, and represent data in various ways to facilitate problem-solving and decision-making.</p> <p><b>1.5.c</b> Break problems into component parts, extract key information, and develop descriptive models to understand complex systems to facilitate problem-solving.</p> <p><b>1.6.b</b> Create original works or responsibly repurpose or remix digital resources into new creations.</p> <p><b>1.6.d</b> Publish and present content that customizes the message and medium for their intended audience.</p>
--	---	--

Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>Makey Makey</b></p> <p>Learn about computer systems and how hardware interacts with software. Use basic programming skills, like sequences and events, to program the Makey Makey.</p>	<p><b>1B-CS-02</b> Model how computer hardware and software work together as a system to accomplish tasks.</p> <p><b>1B-CS-03</b> Determine potential solutions to solve simple hardware and software problems using common troubleshooting strategies.</p> <p><b>2-CS-02</b> Design projects that combine hardware and software components to collect and exchange data.</p> <p><b>2-CS-03</b> Systematically identifies and fixes problems with computing devices and their components.</p> <p><b>1B-AP-10</b> Create programs that include sequences, events, loops, and conditionals.</p> <p><b>1B-IC-21</b> Use public domain or creative commons media, and refrain from copying or using material created by others without permission.</p> <p><b>1B-CS-02</b> Model how computer hardware and software work together as a system to accomplish tasks.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.1.d</b> Understand the fundamental concepts of technology operations, demonstrate the ability to choose, use and troubleshoot current technologies and are able to transfer their knowledge their to explore emerging technologies.</p> <p><b>1.2.c</b> Demonstrate an understanding of and respect for the rights and obligations of using and sharing intellectual property.</p> <p><b>1.4.a</b> Know and use a deliberate design process for generating ideas, testing theories, creating innovative artifacts, or solving authentic problems.</p> <p><b>1.4.c</b> Develop, test, and refine prototypes as part of a cyclical design process.</p> <p><b>1.4.d</b> Exhibit a tolerance for ambiguity, perseverance and the capacity to work with open-ended problems.</p>

Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>Raspberry Pi</b></p> <p>Learn about computer systems and how hardware interacts with software through sensors, cameras, and more.</p>	<p><b>2-CS-02</b> Design projects that combine hardware and software components to collect and exchange data.</p> <p><b>2-CS-03</b> Systematically identify and fix problems with computing devices and their components.</p> <p><b>2-AP-11</b> Create clearly named variables that represent different data types and perform operations on their values.</p> <p><b>2-AP-11</b> Create clearly named variables that represent different data types and perform operations on their values.</p> <p><b>2-AP-12</b> Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.</p> <p><b>2-AP-13</b> Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.</p> <p><b>2-AP-14</b> Create procedures with parameters to organize code and make it easier to reuse.</p> <p><b>2-AP-15</b> Seek and incorporate feedback from team members and users to refine a solution that meets user needs.</p> <p><b>2-AP-15</b> Seek and incorporate feedback from team members and users to refine a solution that meets user needs.</p> <p><b>2-AP-16</b> Incorporate existing code, media, and libraries into original programs, and give attribution.</p> <p><b>2-AP-17</b> Systematically test and refine programs using a range of test cases.</p> <p><b>2-AP-19</b> Document programs in order to make them easier to follow, test, and debug.</p> <p><b>3A-AP-14</b> Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.1.d</b> Understand the fundamental concepts of technology operations, demonstrate the ability to choose, use and troubleshoot current technologies and are able to transfer their knowledge their to explore emerging technologies.</p> <p><b>1.4.a</b> Know and use a deliberate design process for generating ideas, testing theories, creating innovative artifacts, or solving authentic problems.</p> <p><b>1.4.c</b> Develop, test, and refine prototypes as part of a cyclical design process.</p> <p><b>1.4.d</b> Exhibit a tolerance for ambiguity, perseverance and the capacity to work with open-ended problems.</p> <p><b>1.5.d</b> Understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.</p> <p><b>1.6.c</b> Communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations models or simulations.</p> <p><b>1.6.d</b> Publish and present content that customizes the message and medium for their intended audience.</p>

	<p><b>3A-AP-16</b> Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.</p> <p><b>3A-AP-16</b> Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.</p> <p><b>3A-AP-21</b> Evaluate and refine computational artifacts to make them more usable and accessible.</p>	
Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>HTML</b></p> <p>Learn basic HTML syntax and the most popular tags as you create an online job posting.</p>	<p><b>2-AP-15</b> Seek and incorporate feedback from team members and users to refine a solution that meets user needs.</p> <p><b>2-AP-16</b> Incorporate existing code, media, and libraries into original programs, and give attribution.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.2.b</b> engage in positive, safe, legal, and ethical behavior when using technology, including social interactions online or when using networked devices.</p> <p><b>1.2.c</b> Demonstrate an understanding of and respect for the rights and obligations of using and sharing intellectual property.</p> <p><b>1.4.a</b> Know and use a deliberate design process for generating ideas, testing theories, creating innovative artifacts, or solving authentic problems.</p> <p><b>1.4.c</b> Develop, test, and refine prototypes as part of a cyclical design process.</p> <p><b>1.4.d</b> Exhibit a tolerance for ambiguity, perseverance and the capacity to work with open-ended problems.</p> <p><b>1.5.d</b> Understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.</p> <p><b>1.6.d</b> Publish and present content that customizes the message and medium for their intended audience.</p>

Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>Design</b></p> <p>Learn basic design principles such as a design process, and specific web design principles as you create a website for a fictional town.</p>	<p><b>2-AP-13</b> Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.</p> <p><b>2-AP-15</b> Seek and incorporate feedback from team members and users to refine a solution that meets user needs.</p> <p><b>2-AP-16</b> Incorporate existing code, media, and libraries into original programs, and give attribution.</p> <p><b>2-AP-17</b> Systematically test and refine programs using a range of test cases.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.1.d</b> Understand the fundamental concepts of technology operations, demonstrate the ability to choose, use and troubleshoot current technologies and are able to transfer their knowledge their to explore emerging technologies.</p> <p><b>1.4.a</b> Know and use a deliberate design process for generating ideas, testing theories, creating innovative artifacts, or solving authentic problems.</p> <p><b>1.4.d</b> Exhibit a tolerance for ambiguity, perseverance and the capacity to work with open-ended problems.</p> <p><b>1.6.c</b> Communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations models or simulations.</p> <p><b>1.6.d</b> Publish and present content that customizes the message and medium for their intended audience.</p>
Fiero Course Unit	CSTA Standards	ISTE Standards
<p><b>Fiero Missions</b></p> <p>In Fiero, we have over 50 projects for students to work on. These vary by coding discipline (web, games, scripting), and are opportunities for students to use</p>	<p><b>1B-AP-09</b> Create programs that use variables to store and modify data.</p> <p><b>1B-AP-10</b> Create programs that include sequences, events, loops, and conditionals.</p> <p><b>1B-AP-11</b> Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.</p> <p><b>1B-AP-12</b> Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.</p>	<p><b>1.1.a</b> Articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.</p> <p><b>1.1.c</b> Use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways.</p> <p><b>1.1.d</b> Understand the fundamental concepts of technology operations, demonstrate the ability to choose, use and troubleshoot current technologies and are able to transfer their knowledge their to explore emerging technologies.</p>

<p>their own creativity to complete the project.</p>	<p><b>1B-AP-13</b> Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences.</p> <p><b>1B-AP-14</b> Observe intellectual property rights and give appropriate attribution when creating or remixing programs.</p> <p><b>1B-AP-15</b> Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended.</p> <p><b>1B-AP-17</b> Describe choices made during program development using code comments, presentations, and demonstrations.</p> <p><b>1B-IC-20</b> Seek diverse perspectives for the purpose of improving computational artifacts.</p> <p><b>1B-IC-21</b> Use public domain or creative commons media, and refrain from copying or using material created by others without permission.</p> <p><b>2-AP-11</b> Create clearly named variables that represent different data types and perform operations on their values.</p> <p><b>2-AP-12</b> Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.</p> <p><b>2-AP-13</b> Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.</p> <p><b>2-AP-14</b> Create procedures with parameters to organize code and make it easier to reuse.</p> <p><b>2-AP-15</b> Seek and incorporate feedback from team members and users to refine a solution that meets user needs.</p> <p><b>2-AP-16</b> Incorporate existing code, media, and libraries into original programs, and give attribution.</p> <p><b>2-AP-17</b> Systematically test and refine programs using a range of test cases.</p> <p><b>2-AP-19</b> Document programs in order to make them easier to follow, test, and debug.</p>	<p><b>1.2.b</b> engage in positive, safe, legal, and ethical behavior when using technology, including social interactions online or when using networked devices.</p> <p><b>1.2.c</b> Demonstrate an understanding of and respect for the rights and obligations of using and sharing intellectual property.</p> <p><b>1.4.c</b> Develop, test, and refine prototypes as part of a cyclical design process.</p> <p><b>1.5.a</b> Formulate problem definitions suited for technology assisted methods such as data analysis, abstract models and algorithmic thinking in exploring and finding solutions.</p> <p><b>1.5.c</b> Break problems into component parts, extract key information, and develop descriptive models to understand complex systems to facilitate problem-solving.</p> <p><b>1.5.d</b> Understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.</p> <p><b>1.6.b</b> Create original works or responsibly repurpose or remix digital resources into new creations..</p> <p><b>1.6.c</b> Communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations models or simulations.</p> <p><b>1.6.d</b> Publish and present content that customizes the message and medium for their intended audience.</p>
--	---	--

	<p><b>3A-AP-14</b> Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.</p> <p><b>3A-AP-16</b> Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.</p> <p><b>3A-AP-21</b> Evaluate and refine computational artifacts to make them more usable and accessible.</p> <p><b>3B-AP-09</b> Implement an artificial intelligence algorithm to play a game against a human opponent or solve a problem.</p> <p><b>3A-AP-15</b> Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made.</p> <p><b>3A-AP-19</b> Systematically design and develop programs for broad audiences by incorporating feedback from users.</p> <p><b>3A-AP-21</b> Evaluate and refine computational artifacts to make them more usable and accessible.</p>	
<p><b>CSS</b></p> <p>Learn the basic syntax of CSS and common properties as well as advanced concepts such as positions, grid, and flexbox as you create a library digital display.</p>		