



Internet of Things (IoT) at



*UCF RET Site: Collaborative Multidisciplinary
Engineering Design Experiences for Teachers*

2003340: Chemistry 1

LYNNE N COHEN

2003340: Chemistry 1

2019-2020 School Year

Why Should I Use these Lessons?

Integration of engineering and computer science in secondary education supports student success and increases the likelihood that students will pursue STEM careers (Gullen & Sheldon, 2014). Computer programming can be linked to many subjects but has specific potential in Chemistry curricula to support both digital and scientific literacy:

- *Chemistry courses teach problem-solving skills grounded in practical applications & may directly translate to lucrative engineering & computer science careers (Carr, 2014).*
- *Computer science and chemistry share similar skill sets of defining problems, developing models, designing piece-wise solutions, interpreting data, and procedurally testing methods and solutions (National Research Council, 2012).*
- *Computing is required in modern experimentation for modeling and calculation, i.e., MATLAB which uses syntax similar to the Java programming language (Carr, 2014).*
- *Computer science challenges are authentic applications for science modeling, formula use, and unit analysis – all essential to 6-12 chemistry standards.*
- *Students report increased motivation for hands-on computer-based activities involving authentic problems, creative freedom, and development of student expertise (Gullen & Sheldon, 2014).*

For more information, see “Other Helpful Information” section at the end of this document.

Activities for Modeling Chemical Concepts

Please note: *These lessons are not intended to be taught continuously. Instead, these series of lessons are meant to be inserted throughout the year as the introductory chemistry teacher touches on relevant topics. The purpose of these lessons is to increase student engagement with chemical concepts that are otherwise vague and cerebral. The method by which students are engaged is through hands-on digital model building using programming concepts.*

Order of Instruction: *The lessons are ordered roughly to align with the recommended order of instruction within Orange County Public Schools within the Semester 1 (first two academic quarters), AND to proceed in complexity in terms of scratch concepts introduced. Feel free to rearrange and make tasks more or less complex based on the order of your instruction.*

Progression of Scratch Concepts: *Lessons are arranged in this document to introduce concepts in this order:*

1. **Activity One:** *Create a sprite from library & Use drag and drop to create a simplistic three-block script involving an event (When green flag is clicked), a motion (have the sprite move in any way), and a reaction (print a string).*
2. **Activity Two:** *Use the “Draw Sprite” function in addition to all previous concepts.*
3. **Activity Three:** *Use the “Import Sprite” function and deepen exploration of “move” and “control” functions (Loops & if/then)*
4. **Activity Four:** *Use sensing (user input), operators (math), and variables in addition to all previous skills.*

RET Site Lesson Plan: Introduction to Scratch

Subject Area(s): Chemistry

Course(s): Chemistry 1 (2003340), Regular or Honors

Grade Level: 9-12

Suggested Length of Lesson: 1-2 days

Lesson Summary: Students are introduced to the Scratch platform and are given a few simple tasks to become acquainted with its basic structure and functions, including Sprites, events, responses, and motion. This will prepare them for more advanced future tasks.

Prerequisite Knowledge:

1. Basic computer functions (how to use a computer, access a web browser, search the internet, download images, etc.)
2. Recent introduction of several new vocabulary terms.

Materials/Technology Needed

- One internet-enabled computer for every student OR student pair.
- Access to the scratch.mit.edu website.

Where this Fits/Lesson Dependency

- This lesson can be adapted to fit any early-year topic, after the introduction of core vocabulary, or when introducing a historical event or person.
- This example is in context of Unit 0: Safety.

Lesson Objective(s)/Learning Goal(s) (2-4)

- SWBAT express three rules of lab safety and conduct.
- SWBAT use the Scratch Programming platform to perform the following tasks:
 - Create and/or import a sprite
 - Use drag and drop to create a minimum of a three-block script involving an event (When green flag is clicked), a motion (have the sprite move in any way, AND...), and a reaction (have the sprite state three rules of lab safety and conduct).

Standard(s)/Benchmark(s) Addressed (2-4)

- *Standards:*
 - *SC.912.N.1.1 Define a problem based on a specific body of knowledge... Pose answers, explanations, or descriptions of events.*
 - *LAFS.1112.RST.1.2 Determine the central ideas or conclusions of a text; summarize complex concepts, processes, or information presented in a text by paraphrasing them*

Standards for Mathematical Practice

- MAFS.K12.MP.1.1 Make sense of problems and persevere in solving them.

Instructional Strategies

- Gradual Release
 - Show teacher-constructed model, Provide very basic instructions, Encourage and allow time for individualization and elaboration
- Differentiated Instruction
 - Audio, Visual, and Hands-On Lesson Components
- Universal design
 - Visual block-based programming is “low floor, high ceiling”.

Evidence of Learning (Assessment Plan)

- Student learning will be assessed once through their participation in the “What’s wrong with this lab?” Discussion Critique.
- Student learning will again be assessed through their Scratch program.

Description of Lesson Activity/Experiences

1. After a short slide show detailing lab safety and conduct, students will engage in a class activity and discussion about safety, involving a “what’s wrong with this lab?” image in which they attempt to list all the problems visually wrong with a photo of lab misconduct.
2. Show students an example Scratch program with a very simple code:
<https://scratch.mit.edu/projects/320713618/>
3. Students will be asked to make a Scratch account (log in for free with google), and either alone or in pairs, perform the following:
 - Create a new project
 - Create a sprite
 - Choose and insert an initiating event block (Click flag, Click sprite, or Press a certain key)

- Insert a motion block of choice
- Insert a “say” block and insert three rules of the lab.

4. Students will raise their hands when done to show the teacher.

Recommended Assessment(s) and Steps

- Student learning will be assessed once through their participation in the “What’s wrong with this lab?” Discussion Critique.
- Student learning will again be assessed through their Scratch program.

List of Materials/Resources Used

- One internet-enabled computer for every student OR student pair.
- Access to the scratch.mit.edu website.

Engineering Connection (60-100 words/3 sentences)

- Students demonstrate their understanding of a simple lesson by using the Scratch Visual Block-Based Programming Language, in anticipation of later, more advanced Computer Science connections.

Engineering Category (choose one)

X	relating science and/or math concepts to engineering (primarily science & math with some engineering)
	engineering analysis or partial design (primarily engineering with some science/math)
	engineering design process (full engineering design)

Key Words

Programming, Scratch, Computer Science, High School, Science, Lab Safety

Introduction/Motivation (written as if talking to students)

(After the lesson has been delivered)

Who here has watched a movie with computer graphics? CG?
 Who here has played a video game?
 Who here has visited a web page?

All these things involve CODE - that’s the language “behind the scenes” that tells a computer what to show the audience. Today we’ll be making our lesson a little more interesting by animating it. Let’s start by logging into scratch.mit.edu and logging in with our school google accounts.

If you’d like to work with a partner, please feel free to partner up now.

Lesson Closure (written as if talking to students)

Congratulations, you’ve passed “Lab Safety 101”, and you’ve used your computer to make a simple program! Computers are very useful to us when we are doing science. We can use computers to better understand our observations, to check our math so we don’t make errors, and to make visual models to help others understand us better. This year, we’ll be using “Scratch” a few times, so feel free to play with it on your own and try to make your sprite do new and interesting things.

Lesson Background & Concepts for Teachers

- Expertise with Scratch is NOT required for this lesson, but recommended that you play with at least the functions the students will be using.
- This lesson can be adapted to any lesson which would otherwise be wrote memorization; new vocabulary, etc. It can be extended in a variety of ways (i.e., History, import a picture of a famous person as the sprite, and have them say their favorite quote... Ice Breaker, import a picture of your favorite animal, and have them say what your favorite food is... etc.)
- In this lesson, allow students to become sidetracked with other programming blocks if they wish to. As long as they demonstrate understanding of the learning goal (which should be kept simple), distraction within the Scratch platform will only lead to better understanding and more advanced models later in the year.

RET Site Lesson Plan: Atomic Models

Subject Area(s): Chemistry

Course(s): Chemistry 1 (2003340), Regular or Honors

Grade Level: 9-12

Suggested Length of Lesson: 2 days

Lesson Summary: Students use Scratch to represent one of the atomic models, historical or modern.

Prerequisite Knowledge:

1. Prior experience with Scratch in Introductory Lesson
2. Knowledge of subatomic particles (protons, neutrons, electrons)

Materials/Technology Needed

- One internet-enabled computer for every student OR student pair.
- Access to the scratch.mit.edu website.
- Atomic Model & Scratch Tips Slide set at https://docs.google.com/presentation/d/1-SQe1vKwp_zl_vun93EjvTfM8nqS99r3WgqjK5UETBl/edit?usp=sharing

Where this Fits/Lesson Dependency

- This lesson should be provided near the start of the Atom unit, prior to detailed instruction about the specific models in order to maintain individual interpretation and creativity.

Lesson Objective(s)/Learning Goal(s) (2-4)

- SWBAT provide a digital drawing and/or animated drawing of an atomic model from the following options:
 - Democritus or Dalton
 - Thompson
 - Rutherford
 - Bohr
 - Quantum
- SWBAT explain how their model matches that historical model.
- SWBAT use the Scratch Programming platform to perform the following tasks:
 - Use the “Draw Sprite” function.
 - Continue incorporating previous concepts: Use drag and drop; Use event, motion, and “say” reaction
- Extended: SWBAT compare and contrast projects and express improvements to be made.

Standard(s)/Benchmark(s) Addressed (2-4)

- *Standards:*
 - *SC.912.P.8.3 Explore the scientific theory of atoms (also known as atomic theory) by describing changes in the atomic model over time*
 - *SC.912.P.8.4 Explore the scientific theory of atoms (also known as atomic theory) by describing the structure of atoms in terms of protons, neutrons and electron,...*

Standards for Mathematical Practice

- MAFS.K12.MP.4.1 Model with mathematics.
- MAFS.K12.MP.7.1 Look for and make use of structure.

Instructional Strategies

- Gradual Release
 - Show teacher-constructed model, Provide very basic instructions, Encourage and allow time for individualization and elaboration
- Differentiated Instruction
 - Audio, Visual, and Hands-On Lesson Components
- Universal design
 - Visual block-based programming is “low floor, high ceiling”.
- Student choice

Evidence of Learning (Assessment Plan)

- Student learning will be assessed through observation as teacher circulates during the activity and answers questions.
- Student learning be assessed by their reflection on their models.

	<ul style="list-style-type: none"> ○ Students have freedom to explore motion blocks to create their desired model.
<p>Description of Lesson Activity/Experiences</p> <ul style="list-style-type: none"> ● On Day One, students are given a short lecture with slideset Atomic Models & Scratch: <ul style="list-style-type: none"> ○ https://docs.google.com/presentation/d/1-SQe1vKwp_zl_vun93EjyTfM8nqS99r3WgqjK5UETBI/edit?usp=sharing ○ Part 1: The need for models in chemistry for nanoscale objects (Dalton, 1808; Boston Museum of Science, n.d.) ○ Part 2: Historical Atomic Models (Chamrat, 2009; “The history of the atom – theories and models”, 2013) ○ Part 3: Scratch Overview (http://scratch.mit.edu) ○ Given time, students may begin work on their models. ● On Day Two, students work in pairs on their scratch project models. <ul style="list-style-type: none"> ○ Fifteen minutes prior to the end of the period, students share links to their projects in preparation for a full class discussion. ○ Show-and-tell: Arrange models in terms of chronological order. Compare and contrast. 	
<p>Recommended Assessment(s) and Steps</p> <ul style="list-style-type: none"> ▪ Student learning will be assessed through observation as teacher circulates during the activity and answers questions. ▪ Student learning be assessed by their reflection on their models. 	
<p>List of Materials/Resources Used</p> <ul style="list-style-type: none"> ▪ One internet-enabled computer for every student OR student pair. ▪ Access to the scratch.mit.edu website. ▪ Atomic Model & Scratch Tips Slide set at https://docs.google.com/presentation/d/1-SQe1vKwp_zl_vun93EjyTfM8nqS99r3WgqjK5UETBI/edit?usp=sharing 	
<p>Engineering Connection (60-100 words/3 sentences)</p> <ul style="list-style-type: none"> ▪ Students will be using a digital tool in order to draw their model, and will use blocks of logical code in order to animate their model in a meaningful way. ▪ Students will approach the task with creativity and independence, and use trial and error to optimize the use of a computing tool to accomplish meet project requirements. 	
<p>Engineering Category (choose one)</p>	
X	relating science and/or math concepts to engineering (primarily science & math with some engineering)
	engineering analysis or partial design (primarily engineering with some science/math)
	engineering design process (full engineering design)
<p>Key Words Programming, Scratch, Computer Science, High School, Science, Chemistry, Atomic Models, Historical Models, Nanoscale, Protons, Electrons, Neutrons, Democritus, Dalton, Rutherford, Thompson, Bohr</p>	
<p>Introduction/Motivation (written as if talking to students)</p> <p>What does it mean to you when someone says the word “model”?</p> <ul style="list-style-type: none"> ● <i>Fashion models, model planes/cars, etc... some students may mention scientific models. Gather all ideas on the board.</i> <p>What do all these models have in common?</p> <ul style="list-style-type: none"> ● <i>They are all used to show an example of something.</i> <p>Why is it useful to have an example of something? For example, why do you like seeing a picture of something before buying it online?</p> <ul style="list-style-type: none"> ● <i>It helps you know what you will get - what you will be able to do with it.</i> <p>If you were looking to buy shoes online, what would you look for in the picture?</p> <ul style="list-style-type: none"> ● <i>Expect different student answers - heels, no heels, athletic shoes, dress shoes, etc.</i> ● <i>Why are there different answers? We might want different shoes for different purposes.</i> <p>In science, we use models for many practical reasons, and it all depends on what we want to do with them. Let’s learn more about why we use models in chemistry specifically. (Segue to Slide set).</p>	

Lesson Closure (written as if talking to students)

- Now that you've created your own model and seen the other models made here in our class, can you describe which models are the "best"?
- Are some models better for one purpose or another? What might those purposes be?
- If you had unlimited time to improve your model, what would you do?
 - **Exit ticket:** On an index card, please name and describe three atomic models in *words*.

Lesson Background & Concepts for Teachers

- Expertise with Scratch is NOT required for this lesson, but recommended that you play with at least the functions the students will be using.
- If students finish early and become distracted within the Scratch platform, allow the deviation if you can. Playful interaction within the platform will only lead to better understanding and more advanced models later in the year.

RET Site Lesson Plan: Modeling Molecular Motion

Subject Area(s): Chemistry

Course(s): Chemistry 1 (2003340), Regular or Honors

Grade Level: 9-12

Suggested Length of Lesson: 2 days

Lesson Summary: Students use Scratch to model the motion of molecules in at least one, but up to three different states: Solid, Liquid, and Gas.

Prerequisite Knowledge:

3. Prior experience with Scratch in Introductory Lesson
4. Recent introduction to molecular motion in solids, liquids, and gases.

Materials/Technology Needed

- One internet-enabled computer for every student OR student pair.
- Access to the scratch.mit.edu website.

Where this Fits/Lesson Dependency

- This lesson might be incorporated as a formative assessment during unit introducing Kinetic Molecular Theory and/or Phases of Matter.
- Students MUST have learned about the motion of solid, liquid, and gas particles.
- Good preparation for hands-on project lesson such as “Obleck”

Lesson Objective(s)/Learning Goal(s) (2-4)

- SWBAT explain the motion of molecules in one to three different phases, both visually and textually.
- SWBAT use the Scratch Programming platform to perform the following tasks:
 - Import a sprite
 - Create a programming block of medium complexity (5 or more blocks) which includes control (loops), more complex explorations of motion and a “say” action.

Standard(s)/Benchmark(s) Addressed (2-4)

- *Standards:*
 - *SC.912.P.12.10 Interpret the behavior of ideal gases in terms of kinetic molecular theory.*
 - *SC.912.P.12.11 Describe phase transitions in terms of kinetic molecular theory.*

Standards for Mathematical Practice

- MAFS.K12.MP.1.1 Make sense of problems and persevere in solving them.
- MAFS.K12.MP.4.1 Model with mathematics.
- MAFS.K12.MP.7.1 Look for and make use of structure.

Instructional Strategies

- Gradual Release
 - Show teacher-constructed model, Provide very basic instructions, Encourage and allow time for individualization and elaboration
- Differentiated Instruction
 - Audio, Visual, and Hands-On Lesson Components
- Universal design
 - Visual block-based programming is “low floor, high ceiling”.
- Student choice
 - Students have freedom to explore motion blocks to create their desired model. Further choice is provided on Day 2 when students may choose to perfect their original model, or try one of the others.

Evidence of Learning (Assessment Plan)

- Student learning will be assessed through observation as teacher circulates during the activity and answers questions.
- Student learning be assessed by their reflection on their models.

Description of Lesson Activity/Experiences

1. Provide a teacher-led presentation or reading on solids, gases and liquids (macroproperties AND molecular motion).
 - Suggested resource if you do not have your own: <https://preparatorychemistry.com/KMT.html> (Bishop, 2012).
2. Students are given a checkpoint assessment: Draw on a piece of paper what solid, liquid, and gas particles look like.
 - Ask: What was your challenge? Why is a static image not enough? What would you need to better show this?
3. Students will log into their Scratch accounts and begin a new project. Provide instructions: **May be done in singles or pairs**
 - Download an image of a water molecule from the internet.
 - Upload the sprite into your project.
 - Under the sprite, enter an event block to start on Flag click.
4. At this point, ask students (or pairs) to count off “1”, “2”, or “3”.
 - “1”s will model solids; “2”s will model liquids; “3”s will model gases.
 - Allow remainder of class time for free exploration of motion.
 - **Be sure** to go over the “forever” repeat block no later than five minutes into free exploration phase, as students may be struggling with the concept.
 - If a student asks about more than one molecule, demonstrate how to duplicate a sprite. *Remind students that duplicating will duplicate all code at that moment - but if change is made to the code, they will have to change both, or re-duplicate.*
 - **This step will last at least the remainder of the period; allow them to continue on Day 2.**
5. 10-15 minutes into Day 2, ask students to pause.
 - Each “1”, “2”, “3” group will come together to show their progress and explain what they are trying to model. *If groups are odd-numbered, group the students so that everyone is comparing at least two models of **different** phases.*
 - Time 90 seconds to show and explain; time another 90 seconds for questions and feedback.
 - Repeat two more times, so each model is demonstrated and receives feedback.
6. Students are now to add a “say” command that explains the difference between solid, liquid, and gaseous molecular motion.
7. Once done, students may spend the remainder of the second period perfecting their original model, or may try modeling one of the other phases.

Optional: Show students a prepared model of each phase: [solids](#), [liquids](#) & [gases](#) . Ask for critiques and/or ask how they think it was done.

Recommended Assessment(s) and Steps

- Student learning will be assessed through observation as teacher circulates during the activity and answers questions.
- Student learning be assessed by their reflection on their models.

List of Materials/Resources Used

- One internet-enabled computer for every student OR student pair.
- Access to the scratch.mit.edu website.

Engineering Connection (60-100 words/3 sentences)

- Students are presented a challenge -- to model a visual concept (molecular motion in different phases of water) which has been described to them verbally and with simple animated gifs. To accomplish this, students must use logic and visual programming blocks in order to control the motion of a sprite on the Scratch platform. Through trial and error they will develop a model that they believe approximates the motion, and then will give and receive feedback with other groups. Through this process they will need to

consider the requirements set forth (model the motion of a molecule), design a solution (construct a block of programming commands), execute the script, and make alterations to their design. They will need to consider constraints of the platform, and of allotted time. Through the reflection, they will consider improvements that could be made.

Engineering Category (choose one)

X	relating science and/or math concepts to engineering (primarily science & math with some engineering)
	engineering analysis or partial design (primarily engineering with some science/math)
	engineering design process (full engineering design)

Key Words

Programming, Scratch, Computer Science, High School, Science, Phases of Water, Water, Chemistry, Models, Kinetic Molecular Theory

Introduction/Motivation (written as if talking to students)

Which is a best model of phase changes: [Static Image](#) (Lumen Learning, n.d.), [Static Image with Motion Symbols](#) (Science Learning Hub, 2010) or [Animation](#) (University of Colorado, 2019)?

Why?

Today we're going to be making our own animations. Let's have nine groups of 2-3 students each.

Lesson Closure (written as if talking to students)

Now that you've all made a model and discussed the differences between Solids, Liquids, and Gases on the molecular scale, we can see that not only do they all behave quite differently, but it's also a little challenging to model them, right?

What are some challenges you had, solid groups? Liquid groups? Gas groups?

Which was the easiest state to model, do you think? The hardest?

If you had more time, how would you improve your model?

For the rest of the period, go ahead and perfect your model, OR choose another state to model. Remember to share and post your projects at the end.

Lesson Background & Concepts for Teachers

- Expertise with Scratch is NOT required for this lesson, but recommended that you play with at least the functions the students will be using.
- If students finish early and become distracted within the Scratch platform, allow the deviation if you can. Playful interaction within the platform will only lead to better understanding and more advanced models later in the year.

RET Site Lesson Plan: Stoichiometry Calculator

Subject Area(s): Chemistry

Course(s): Chemistry 1 (2003340), Regular or Honors

Grade Level: 9-12

Suggested Length of Lesson: 7 Days

Lesson Summary:

Each year, some students find it difficult to learn the math of chemical reactions - a topic we call "Stoichiometry". Stoichiometry involves four major tasks:

1. Convert between moles (of gas) and volume. This requires multiplying by a constant simple variable:
 - a. $X \text{ mol} * 22.4 \text{ L/mol} = \text{___ L}$
2. Convert between moles and particles. This requires multiplying by a constant variable with scientific notation:
 - a. $X \text{ mol} * 6.02 \times 10^{23} \text{ particles/mol} = \text{___ particles}$
3. Convert between moles and grams. This requires using a reference document to look up the multiplying value:
 - a. Identify the species on the periodic table, and look up its atomic weight (g/mol).
 - i. *In the case of a molecule, add all to find the atomic weight of the entire molecule.*
 - b. $X \text{ mol} * Y \text{ g/mol} = \text{___ grams}$
4. Use mole ratios in a stoichiometric problem:
 - a. Understand the syntax of a chemical equation to interpret the location of information such as number of moles.
 - b. Analyze a chemical equation to compare moles of reactants with moles of products.
 - c. Given a number of moles of one chemical species, interpret a relevant chemical equation to determine the number of moles of any other species in the reaction.

Prerequisite Knowledge:

1. Previous experience with scratch (creating and editing sprites, motion). See examples in previous activities.

Materials/Technology Needed

- One internet-enabled computer for every student OR student pair.
- Access to the scratch.mit.edu website.
- Guiding slide show:
<https://docs.google.com/presentation/d/1mgMDUZ77bJQerefTHFePGkx8mz1-cUqlcfduW3QcikE/edit?usp=sharing>

Where this Fits/Lesson Dependency

- Lesson plan covers the first 6 days of the Mole unit.

Lesson Objective(s)/Learning Goal(s) (2-4)

- SWBAT gain insight into mole conversion practices and internalize relevant operations by expressing the operations procedurally in code.
- SWBAT use the Scratch Programming platform to perform the following tasks:
 - Create multiple sprites using any method (choose, import, draw)
 - Use movement, say, and event blocks (previously covered)
 - Use sensing (user input), operators (math), and variables (advanced concept) in addition to all previous skills.

Standard(s)/Benchmark(s) Addressed (2-4)

- *Standards:*
 - *SC.912.P.8.9 Apply the mole concept and the law of conservation of mass to calculate quantities of chemicals participating in reactions.*

Standards for Mathematical Practice

- MAFS.K12.MP.4.1 Model with mathematics.

Instructional Strategies

- Gradual Release

<ul style="list-style-type: none"> MAFS.K12.MP.8.1 Look for and express regularity in repeated reasoning. 	<ul style="list-style-type: none"> Show teacher-constructed model, Provide very basic instructions, Encourage and allow time for individualization and elaboration Differentiated Instruction <ul style="list-style-type: none"> Audio, Visual, and Hands-On Lesson Components Universal design <ul style="list-style-type: none"> Visual block-based programming is “low floor, high ceiling”. Student choice <ul style="list-style-type: none"> Students have freedom to explore motion blocks to create their desired model. Further choice is provided on Day 2 when students may choose to perfect their original model, or try one of the others.
<p>Evidence of Learning (Assessment Plan)</p> <ul style="list-style-type: none"> Teacher should observe students, cycle, and ask guiding questions throughout. Sample Conversion questions should be prepared and asked throughout. Ask students to solve first, then check on their calculator. Student’s Scratch project will be evidence of student’s processing of lesson objectives. 	
<p>Description of Lesson Activity/Experiences</p> <p>Day 1: What is a mole?</p> <ul style="list-style-type: none"> Avogadro's Number and mole concept; Create a Scratch with a Mole sprite. <p>Day 2: Conversion Practice Practice conversions & Unit analysis.</p> <ul style="list-style-type: none"> Provide example calculator: https://scratch.mit.edu/projects/318420521/ Ask students several questions: <ul style="list-style-type: none"> Is this helpful? How do you think it is made? Could you create this on your own? Could you improve it? <p>Day 3: Converting with Sci Notation & Introduce Variables</p> <ul style="list-style-type: none"> Convert from moles to particles. Add a calculator to scratch to do the conversion for you! Input * 6.02E23 = Output <p>Day 4: Moles to Volume & Continue with variables</p> <ul style="list-style-type: none"> Prove 22.4L per mole at STP with ideal gas law. Add to calculator! Input * 22.4 = Output <p>Day 5: Molar mass</p> <ul style="list-style-type: none"> On this day, use any established lesson regarding molar mass and converting moles to grams. Recommended if you do not have an established lesson: https://www.cpalms.org/Public/PreviewResourceLesson/Preview/120543 Ensure that early day practices with single elements & diatomics, and ends with simple molecules. Finish lesson with discussion: <ul style="list-style-type: none"> Q: So what number do we need to multiply by to get from moles to grams? A: There is no set number - it's whatever the mass of the species is. Make connection to variables. Q: Can we add moles to grams into our calculator? A: Yes, but we need to know molar mass. Q: How can we do that? A: We need a second input from the user. <p>Day 6: Molecular Molar Mass & Use 2 inputs!</p> <ul style="list-style-type: none"> Begin day with practicing with simple molecules. Add to calculator! Input 1 * Input 2 = Output <p>Day 7: Final Touches & Demonstration</p>	
<p>Recommended Assessment(s) and Steps</p> <ul style="list-style-type: none"> Teacher should observe students, cycle, and ask guiding questions throughout. Sample Conversion questions should be prepared and asked throughout. Ask students to solve <i>first</i>, then check on their calculator. Student’s Scratch project will be evidence of student’s processing of lesson objectives. 	

List of Materials/Resources Used

- One internet-enabled computer for every student OR student pair.
- Access to the scratch.mit.edu website.
- Guiding slide show:
<https://docs.google.com/presentation/d/1mgMDUZ7bJQerefTHFePGkx8mz1-cUqlcfdQuW3QcikE/edit?usp=sharing>

Engineering Connection (60-100 words/3 sentences)

- Students internalize a series of computational concepts through creating a procedural map of the operations using blocks of code (Computer Science). Students may approach the task with as much creativity as they like; while core block concepts are required (operations and variables), a variety of design options remain open to students throughout the project. The resulting product acts as a study aid and mind map of the procedure students will need to perform the operations on their own.

Engineering Category (choose one)

X	relating science and/or math concepts to engineering (primarily science & math with some engineering)
	engineering analysis or partial design (primarily engineering with some science/math)
	engineering design process (full engineering design)

Key Words

Programming, Scratch, Computer Science, High School, Science, Chemistry, Models, Mole Map, Mole Conversions, Avogadro’s Number, Stoichiometry, Mole Ratio, Molar Mass

Introduction/Motivation (written as if talking to students)

Who loves doing math by hand?

Who loves using calculators instead?

A huge part of chemistry is actually doing math--however, scientists *love* using technology to make their lives easier. Since we’re learning what it’s like to think like a scientist, we’re going to be doing the same thing.

To do this, we’re going to be again using Scratch. Now, please sign into scratch and create a new project. *If you’d like to work in pairs, you may - remember that you will be working together all week.*

Lesson Closure (written as if talking to students)

Over the past week and a half, we have learned four different conversions we can make from moles. We’ve done the calculations by hand, but we’ve also created our own “Mole Map Calculators” to have a visual reference. As a bonus, we’ve also used the calculator to check our practice answers.

I’d like you to bookmark this calculator right now. During normal class periods, you may use your calculator *for the rest of the year* to check your answers. Just remember you won’t be able to use it on the test!

Because of that, let’s take some time now to add some more notes to our calculators, to remind ourselves what each piece is doing. If you need some examples, look again at some [Mole Maps](#). If you forget which way is multiply and which is divide, make a note! Or, make a note to yourself to use unit analysis to line up the units.

I’ll be coming around now to help and answer any questions. Don’t forget to share and post your link when you feel ready.

Lesson Background & Concepts for Teachers

- Expertise with Scratch is NOT required for this lesson, but recommended that you play with at least the functions the students will be using.
- If students finish early and become distracted within the Scratch platform, allow the deviation if you can. Playful interaction within the platform will only lead to better understanding and more advanced models later in the year.

Vocabulary

Chemistry Terms

Term	Definition
Atom	The smallest unit of matter. Comprised of a central nucleus of one or more protons and zero or more neutrons, which is surrounded by a cloud of electrons. A neutral atom includes electrons equal to the number of protons.
Element	One of the species of the periodic table, of which all atoms have the same number of protons. All atoms of the same element also share specific properties such as electronegativity, metallic nature, and ionization energy.
Molecule	Two or more elements covalently bonded together and acting as a unit.
Ionic Compound	Two or more elements ionically bonded together and acting as a unit.
Atomic mass	The mass of a single atom, determined as the sum of protons and neutrons in the atom (electron masses are negligible and not counted). Different atoms of the same element may have different atomic masses if they are isotopes of each other (having different numbers of neutrons). The unit of atomic mass is the "atomic mass unit" (amu)
Atomic weight	The weighted average of the atomic masses of all the isotopes of an element. Displayed on the periodic table, and used as an average atomic mass. Unit is the "atomic mass unit" (amu), but it also equal to grams per mole (g/mol).
amu	"Atomic mass unit" is 1/12 the mass of a Carbon-12 atom. In practice, it is the mass of exactly one proton or one neutron. Also equal to grams per mole.
Molecular weight	The sum of the atomic weights of all atoms in a molecule or ionic compound.
STP	"Standard Temperature and Pressure", defined as 1 atm (air pressure at sea level) and 273.15K (the freezing point of water at 1 atm).
Volume of gas at STP	The amount of three-dimensional space taken up by any gaseous species at STP, defines in liters: 22.4L/mol
Mole	An expression of quantity that equals exactly 6.02×10^{23} units; used most often to express quantities of nanoscale particles (atoms or molecules).
Chemical Species	Any element or molecule.
Chemical Reaction	An interaction among one or more chemical species that results in a chemical change (defined as the alteration or creation of a new chemical species)
Reactant	The chemical species involved in a chemical reaction
Product	The chemical species resulting from a chemical reaction
Chemical Formula	The written expression of an element, molecule, or ionic compound, which describes the number of atoms of each element in the species or the ratio of elements in an ionic compound.
Chemical Equation	A written expression of a chemical reaction. Includes: the chemical formulas of all reactants and products, as well as the ratios of those chemical species participating in the reaction. The ratios are expressed in terms of moles, and written in the chemical equations as coefficients.
Mole Ratio	In a chemical equation, the number of moles of one species as a ratio to the number of moles of another species.

Scratch/Programming Terms

Term	Definition
Code	A small instruction that's part of a larger program, or animation.
Block	A piece of code in the Scratch platform.
Sprite	An image that can be animated in scratch.
Event	Something that starts a program running.
Input	Keystrokes or mouse clicks from the user of a program.
Variable	An unknown value or placeholder in a program

Troubleshooting Tips

Make sure that Scratch platform is available on your District's network permissions prior to planning these lessons. More information on Scratch and research supporting Scratch as a student learning tool can be found at <https://www.scratchfoundation.org/media-kit>.

Note the potential difficulty levels of different Scratch tasks in order to plan for areas that may require additional time or instruction.

Based on a 2013 case study of 423 scripted projects (Maloney et. al., 2008):

- **Pre-Programming:** Import Images, Draw Sprites, Record Sounds.
- **Beginner Scripting:** Animate a single sprite, Small stacks of code, Simple user interaction (click to start script), Simple motion.
- **Intermediate Scripting:** Multiple sprites, Larger stacks of code, Keyboard input, Loops, Simple Conditional (if/then), Simple Broadcasting & When-Receive.
- **Advanced Scripting*:** Boolean statements (And, Or, Not), Randomization, Number generators, Uncommon math operations (absolute value), Variables
 - ** According to Maloney et. al., students are less likely to approach these concepts without guidance.*

Other Helpful Information

Adaptation - Projects can be altered for a variety of topics and subjects. Guiding points:

- **Keep it simple.** The first activity (and repeated variations) should require only 2-5 blocks. Intermittent small challenges encourage familiarity and confidence.
- **Build slowly.** Add one or two new concepts each time and consider difficulty.
- **Creativity takes time.** Plan for plenty of time for creative or exploratory tasks.

Gradual Release - Give structured guidance at the start; expect independence at the end.

- **"I do"** - Clarify expectations by demonstrating with a visual image or simplistic sample program. Feel free to provide sample code blocks to be edited, or other scaffolding as needed, particularly in the starting lessons.
- **"We do"** - Provide explicit instructions and have available sample code to achieve the minimum requirements of task, sufficient to get students started. This will also allow every student to participate in the activity and work towards their science standards, regardless of whether they are interested in the coding aspect. Encourage work in pairs.
- **"You do"** - Encourage exploration beyond the minimum and pose challenges through questions: Does that move? What happens if I click on it? How is this useful as a model? *Since creative endeavors take time, consider making available an alternate assignment or quiz for students who are satisfied with simpler programs. That allows them to continue on-task while still allowing other students time for creative deepening.*

Attachments

- **Activity 1: Scratch Exit Ticket**
 - *Slideshow not included - use any relevant slideshow with new vocabulary.*
 - Sample Scratch Project - Speaking Sprite:
 - <https://scratch.mit.edu/projects/320713618/>
- **Activity 2: Atomic Model**
 - Slideshow: [Atomic Models & Scratch](#)
 - Sample Scratch Project - Rutherford Atom:
 - <https://scratch.mit.edu/projects/318444659/>
 - Sample Scratch Project - Bohr Atom:
 - <https://scratch.mit.edu/projects/318603642/>
- **Activity 3: Molecular Motion**
 - *Slideshow not included - use any prepared lecture covering states of matter.*
 - Sample Scratch Project - Solids:
 - <https://scratch.mit.edu/projects/318890475/>
 - Sample Scratch Project - Liquids:
 - <https://scratch.mit.edu/projects/318598711/>
 - Sample Scratch Project - Gases:
 - <https://scratch.mit.edu/projects/318436658/>
- **Activity 4: Molar Conversion Calculator**
 - Slideshow: [Mole Conversion Scratch Calculator](#)
 - Sample Scratch Project:
 - <https://scratch.mit.edu/projects/318420521/>

References

- Bishop, M. (2012). *KMT*. Chiral Publishing Company. Retrieved from <https://preparatorychemistry.com/KMT.html>
- Boston Museum of Science. (n.d.). Amazing Brothers Nano Show. Retrieved from <https://www.mos.org/amazing-nano-brothers>
- Brunning, A. (2016, October 13). The history of the atom – theories and models. *Compound Interest*. Retrieved from <https://www.compoundchem.com/2016/10/13/atomicmodels/>
- Carr, K. (2014, July 17). Computer science adds new dimension to study of chemistry. *The Stanford Daily*. Retrieved from <https://www.stanforddaily.com>
- Chamrat, S. (2009). *Exploring Thai grade 10 chemistry students' understanding of atomic structure concepts and the nature of science through the model-based approach*. (Doctor of Philosophy (Science Education)). Chiang Mai University, Retrieved from <https://www.researchgate.net>
- Dalton, J. (1808). *A new system of chemical philosophy, Volumes 1-2*. Manchester: R. Bickerstaff, Strand, London.
- Eames Office LLC. (1977). *Powers of ten [Video]*. Retrieved from <https://www.eamesoffice.com/the-work/powers-of-ten/>
- Florida Department of Education. (2015). Chemistry 1 - 2003340. Retrieved from <http://www.cpalms.org/Public/PreviewCourse/Preview/4360>
- Gullen, K., & Sheldon, T. (2014). *Synergy sparks digital literacy: Redefined roles create new possibilities for teachers and students*. *The Learning Professional*, 35(2). Retrieved from <https://learningforward.org/journal/april-2014-issue/synergy-sparks-digital-literacy/>
- Lumen Learning. (n.d.). Kinetic molecular theory of matter. *Boundless Chemistry*. Retrieved from <https://courses.lumenlearning.com/boundless-chemistry/chapter/kinetic-molecular-theory-of-matter/>
- Maloney, J., Peppler, K., Kafai, Y., Resnick, M., & Rusk, N. (2008). *Programming by choice: urban youth*

learning programming with Scratch. Paper presented at the SIGCSE'08 - Proceedings of the 39th ACM Technical Symposium on Computer Science Education.

National Research Council. (2012). *A framework for K-12 science education: Practices, crosscutting concepts, and core ideas* (pp. 400). doi:10.17226/13165

Science Learning Hub. (2010, Apr 12). *States of matter*. Retrieved from

<https://www.sciencelearn.org.nz/resources/1499-states-of-matter>

University of Colorado. (2019, July 11). States of matter. *PHeT™ interactive simulations*. Retrieved from

https://phet.colorado.edu/sims/html/states-of-matter/latest/states-of-matter_en.html

Acknowledgements

Authors

Cohen, Lynne N.

Supporting Program

COMET RET Program, College of Engineering and Computer Science, University of Central Florida. This content was developed under National Science Foundation grant #1611019.

Special Thanks

Special thank you to James Ebbert for coding support, and Barbara Barnard-Figaro for inspiration and introducing Scratch to the cohort.

Contact information

Lynne N. Cohen

Incohen@knights.ucf.edu

Images

Images are either original or used with permission under CC-BY-2.5 under the GNU Free Documentation License and source is documented.