Lesson 6: Creating a Secure Password Generator

In this lesson, students will learn how to create a secure password generator using randomization and string-handling techniques. They will understand the importance of having a strong and unique password for each online account and the potential risks of using weak passwords. Through direct instruction, guided practice, and independent practice, students will gain the skills to manipulate strings, use the random module in Python, and create a program that generates secure passwords. They will also have the opportunity to enhance their program by adding additional features and explore more advanced string-handling techniques and randomization methods. The lesson will culminate with an exit ticket where students will share their programs and explain their approach to creating a secure password.

## **Objectives:**

- Students will be able to create a program that uses string-handling techniques.

- Students will be able to generate a secure password using randomization and string-handling techniques.

## **Materials:**

- Computers with Python programming environment installed

- Internet access (optional, for additional resources)

## **Bell-Ringer Activity:**

- Ask students to brainstorm different methods they use to create passwords.

- Discuss the importance of having a secure password and the potential risks of using weak passwords.

- Share a few examples of common weak passwords and explain why they are not secure.

## **Introduction:**

- Explain to students that in today's lesson, they will be creating a program that generates a secure password using randomization and string-handling techniques.

- Emphasize the importance of having a strong and unique password for each online account.

- Discuss the criteria for a secure password, such as length, complexity, and uniqueness.

## **Direct Instruction:**

- Introduce the concept of randomization in programming and its importance in creating secure passwords.

- Explain the use of the random module in Python and its functions, such as `random.choice()` and `random.shuffle()`.

- Demonstrate how to use the random module to generate random words from a given list.

- Show examples of string-handling techniques, such as concatenation and slicing, that can be used to manipulate and combine strings.

## **Guided Practice:**

- Provide students with a starter code that imports the random module and includes a list of words.

- Instruct students to write a function that randomly selects three words from the list and combines them to create a password.

- Guide students through the process of using string-handling techniques to manipulate the selected words and create a secure password.

- Encourage students to test their code and make any necessary adjustments.

## **Independent Practice:**

- Challenge students to enhance their program by adding additional features, such as allowing the user to specify the length of the password or including special characters.

- Encourage students to research and implement additional string-handling techniques or randomization methods to make their program more robust.

- Provide resources and support for students who may need additional guidance.

## **Exit Ticket:**

- Ask students to share their programs and explain how they used string-handling techniques and randomization to create a secure password.

- Assess students' understanding of the concepts covered in the lesson through their explanations and the functionality of their programs.

## **Closure:**

- Recap the main concepts covered in the lesson, including string-handling techniques, randomization, and the importance of secure passwords.

- Emphasize the practical application of the skills learned in creating a secure password generator.

- Encourage students to continue exploring and practicing their programming skills to create more advanced projects in the future.

**Common Core Standards:**

- CCSS.ELA-LITERACY.RST.9-10.3: Follow precisely a complex multistep procedure when carrying out experiments, taking measurements, or performing technical tasks, attending to special cases or exceptions defined in the text.

- CCSS.ELA-LITERACY.RST.9-10.4: Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to grades 9-10 texts and topics.