Lesson 10 Data Protection and Passwords

Worksheet

K-W-L Chart

What I Know	What I Wonder	What I Learned

New Commands: make a list of all the new Python commands you learn throughout the lesson

Activity 1:

Read the program.

```
1.  import cyberpi
2.
3.  cyberpi.display.clear()
4.  t = 0
5.
6.  while True:
7.      print("Create a password")
8.      pin_1 = input("Type password: ")
9.      pin_2 = input("Type password again: ")
10.     if pin_2 == pin_1:
11.         print("Success!")
12.         break
13.     else:
14.         print("Passwords don't match. Try again.")
15.
16.
17. print("Sign in your account")
18. cyberpi.console.println("Sign in")
19.
20. while t < 3:
21.     pin = input("Password: ")
22.     cyberpi.console.print("Password: ")
23.     cyberpi.console.println(pin)
24.     if pin == pin_1:
25.         cyberpi.console.println("Success!")
26.         break
27.     else:
28.         cyberpi.console.println("Incorrect. Try again.")
29.         t += 1
30.         if t == 3:
31.             cyberpi.console.println("Too many failed attempts.")
```

Answer:

- How to create a password for a new account?
- How to verify the password entered by a user?
- Compare the two while loops used in the example program. What is the difference between them?
- Identify the syntax that enables the following function: 'A user is given 3 attempts to enter the password of the account.' If the user fails 3 times, the user is blocked from entering the password .

Draw a Flowchart for the example program:

Activity 2:

Task 1: Add a function that allows the user to create a username when the user signs in.

Task 2: Add some lighting effects as the indicator.

Task 3: Modify the second 'while' loop. Use the LEDs as an indicator to visualize the number of attempts. (Note: The user has 5 login attempts.)

Task 4: Modify the conditional expressions. Verify the following three conditions:

The input username is incorrect;

The input password is incorrect;

Both the username and password are incorrect.

Activity 3:

Develop a bank card reader based on the structure and logic of the example program.

- Like the example program, first create a PIN for the bank account and store it on the computer.
- When a sender starts a transaction, ask the sender to type the name (or other identification code) of the receiver and the amount of the transit money.
- Then ask the sender to type the PIN.
- Generate a random 4-digit verification code and display it on CyberPi's screen. Ask the sender to enter the verification code.
- Check the PIN and verification code. If both are correct, display the transaction information (including the receiver's name or ID and the transaction amount) on the screen.
- However, if either the PIN, the verification code or both are incorrect, ask the sender to type them again. The sender has limited attempts (for example, 3 attempts).
- Ask the sender to check the transaction information and confirm the transaction by pressing Button B of CyberPi.

Extension:

- Allow students to expand their application by implementing an ATM.
- After they create their account allow the user should add money to their account
- The user should be able to take money out and add as they like but must input their pin before every transaction.