**uest - day 2: Flow Control Fun and Coding Best Practices**

[Unit 1 - Introduction to Python](https://my.ironhack.com/cohorts/64f9d729a7ab01002ae1abce/lms/courses/course-v1:IRONHACK+DAFT4+202409_RMT_ES/modules/ironhack-course-chapter_1)

**Quest - day 2: Flow Control Fun and Coding Best Practices**

**MINI PROJECT**

Generate a quiz about this unit

Summarize this unit

Have any doubt about this content? Ask!

**Introduction**

Welcome to Day 2 of the Python Analytics Quest! Today, dive into modules and packages in Python, enhance your game with functions and perfect your flow control. Let’s start enhancing your game!

**Daily Overview & Schedule**

**Part 1: Collaborative Development & Research Hour (pairs) | 15:00 - 16:00**

Today’s roles reverse: yesterday’s Dev team becomes the Research team and vice versa. Enhance your game by crafting functions and controlling the game flow, to make your game more interactive. Note that on Day 3 you will have more time to work on the game’s functions and flow control.

**Dev Team:**

* Develop functions for game actions, like room navigation, item usage, etc.
* Implement flow control to manage the progression of the game.
* Test your functions and flow control to make sure they work as expected.

**Hints for getting started**

* Once your data structures are set up, start implementing flow control in your game. This will govern how a player moves from room to room and how they interact with items in each room. Consider the following points:
  + How will the player move between rooms? What commands will you use to facilitate this?
  + How will the game respond when the player interacts with an item?
  + What happens if a player tries to move in an impossible direction or interact with an item that isn’t there?
* Define functions to handle different game actions. For example, you can create a function explore\_room(room) to display the room description and available actions. This function can take the current room as an argument and retrieve its details from the game\_state dictionary.
* Implement functions to handle specific actions, such as examine\_item(item) to provide information about a specific item in the room, unlock\_door(door) to unlock a door if the player has the corresponding key, and go\_to\_next\_room(room) to move the player to the next room if the door is unlocked.

**Research Team:**

* Read lesson in student portal: Coding best practices
* Study best coding practices for clean, maintainable code.
* Learn about proper function and module documentation.

**Part 2: Interactive Knowledge Exchange (full group) | 16:00 - 16:30**

Exchange progress and learnings (15 mins per team).

**Dev Team:**

* Explain implemented functions and flow control to the Research team. Share challenges and solutions.
* Illustrate how these changes enhance the game so it is more interactive and dynamic.

**Research Team:**

* Share your research findings on good coding practices and documentation techniques.
* Discuss how these practices can be applied to the game development.

**Part 3: Application & Testing Phase (full group) | 16:30 - 17:30**

This is where the Research team gets hands-on coding experience, and the Dev team asks questions.

**Research Team:**

* Apply the best coding practices and documentation techniques learned to the game code.
* Refactor and document the code where necessary.

**Dev Team:**

* Review and test the changes made by the Research/Innovation team.
* Help troubleshoot any errors that might occur during this process.

Remember to communicate with your team members, make frequent saves, and refresh your work on Google Colab to ensure everyone is updated. Have a productive Day 2!