BEGIN PROGRAM "One Must Go" Admin Program

IMPORT json MODULE

IMPORT os MODULE

FUNCTION input_something(prompt)

REPEAT

DISPLAY prompt

value ← USER INPUT WITH LEADING/TRAILING SPACES REMOVED

IF value IS NOT EMPTY THEN

RETURN value

ELSE

DISPLAY "Input cannot be blank. Please try again."

END REPEAT

END FUNCTION

FUNCTION input_int(prompt, max_value)

REPEAT

DISPLAY prompt

user_input ← INPUT FROM USER

IF user_input IS NOT DIGIT THEN

DISPLAY "Invalid input. Please enter a number."

CONTINUE LOOP

END IF

num ← CONVERT user_input TO INTEGER

IF num BETWEEN 1 AND max_value THEN

```
        RETURN num

    ELSE

        DISPLAY "Please enter a number between 1 and", max_value

    END IF

END REPEAT

END FUNCTION

-------------------------------------------------


FUNCTION save_data(data)

    TRY

        OPEN "data.txt" FOR WRITING AS file

        WRITE data IN JSON FORMAT WITH INDENTATION TO file

        CLOSE file

    EXCEPT ANY ERROR e

        DISPLAY "Error saving data:", e

    END TRY

END FUNCTION

-------------------------------------------------


FUNCTION load_data()

    IF FILE "data.txt" EXISTS THEN

        TRY

            OPEN "data.txt" FOR READING AS file

            loaded_data ← READ JSON DATA FROM file

            IF loaded_data IS A LIST THEN

                RETURN loaded_data

        EXCEPT (FileNotFoundError OR JSONDecodeError)

            SKIP ERROR

    END IF
```

```
        RETURN EMPTY LIST

END FUNCTION

------------------------------------------------


FUNCTION main()

    data ← load_data()

    DISPLAY "Welcome to the Disass 'One Must Go' Admin Program."


    REPEAT FOREVER

        DISPLAY MENU:

            "[a] add, [l] list, [s] search, [v] view, [d] delete or [q] quit."

        choice ← LOWERCASE(INPUT("> "))


        ------------------------------------------------

        IF choice = 'a' THEN

            category_name ← LOWERCASE(input_something("Enter a category name: "))


            IF ANY(c["category"].lower() = category_name) IN data THEN

                DISPLAY "The category already exists."

                CONTINUE LOOP

            END IF


            options ← EMPTY LIST

            DISPLAY "Enter 5 options for this category:"

            WHILE LENGTH(options) < 5 DO

                option ← input_something("Option " + (LENGTH(options)+1) + ": ")

                IF LOWERCASE(option) IN [LOWERCASE(o) FOR EACH o IN options] THEN

                    DISPLAY "That option already exists. Enter a different one."

                    CONTINUE LOOP
```

```
        END IF

        APPEND option TO options

    END WHILE


    APPEND {"category": category_name, "options": options} TO data

    CALL save_data(data)

    DISPLAY "Category added successfully."


-----------------------------------------------

ELSE IF choice = 'l' THEN

    IF data IS EMPTY THEN

        DISPLAY "No categories have been added yet."

    ELSE

        DISPLAY "Current categories:"

        FOR EACH item WITH INDEX i IN data DO

            DISPLAY i, ".", CAPITALIZE(item["category"]), "(", LENGTH(item["options"]), "options)"

        END FOR

    END IF


-----------------------------------------------

ELSE IF choice = 's' THEN

    IF data IS EMPTY THEN

        DISPLAY "No categories available to search."

        CONTINUE LOOP

    END IF


    search_term ← LOWERCASE(input_something("Enter search term: "))

    matches ← ALL ITEMS WHERE search_term IN LOWERCASE(item["category"])
```

```
    IF matches IS EMPTY THEN

        DISPLAY "No categories found containing", search_term

    ELSE

        DISPLAY "Search results:"

        FOR EACH item WITH INDEX i IN matches DO

            DISPLAY i, ".", CAPITALIZE(item["category"]), "(", LENGTH(item["options"]), "options)"

        END FOR

    END IF


-----------------------------------------------

ELSE IF choice = 'v' THEN

    IF data IS EMPTY THEN

        DISPLAY "No categories available to view."

        CONTINUE LOOP

    END IF


    DISPLAY "Select a category to view:"

    FOR EACH item WITH INDEX i IN data DO

        DISPLAY i, ".", CAPITALIZE(item["category"])

    END FOR


    category_num ← input_int("Enter category number: ", LENGTH(data))

    selected ← data[category_num - 1]


    DISPLAY "Category:", CAPITALIZE(selected["category"])

    DISPLAY "Options:"

    FOR EACH opt WITH INDEX j IN selected["options"] DO

        DISPLAY j, ".", opt

    END FOR
```

ELSE IF choice = 'd' THEN

   IF data IS EMPTY THEN

      DISPLAY "No categories available to delete."

      CONTINUE LOOP

   END IF

   DISPLAY "Select a category to delete:"

   FOR EACH item WITH INDEX i IN data DO

      DISPLAY i, ".", CAPITALIZE(item["category"])

   END FOR

   category_num ← input_int("Enter category number: ", LENGTH(data))

   deleted ← REMOVE data[category_num - 1]

   CALL save_data(data)

   DISPLAY "Category deleted successfully."

ELSE IF choice = 'q' THEN

   CALL save_data(data)

   DISPLAY "Thank you for using Disass's One Must Go admin program!"

   BREAK LOOP

ELSE

   DISPLAY "Invalid choice. Please enter a, l, s, v, d, or q."

   END IF

END REPEAT

END FUNCTION

IF PROGRAM IS RUN DIRECTLY THEN

   CALL main()

END IF

END PROGRAM