Admin Login Example

SET **adminList** *=* “tor”

OUTPUT *user for their* **name**

READ *user’s input from* console

STORE *input to* var**name**

IF **name** *is in* **adminList**

OUTPUT *welcome message*

SET **admin** *to* True

ELSE

OUTPUT *welcome message +* **name**

SET **admin** *to* False

**END** IF

OUTPUT *to user an array of options*

READ *user’s input from console*

STORE *input to var* **answer**

SWITCH **answer**

CASE *option 1*

CALL method1

CASE *option 2*

CALL method2

CASE *option 3*

CALL method3

**END**

Simple Calculator

Write a program that asks the user for 2 numbers adds them together then prints the result

OUTPUT *user for first number*

READ *the console for user’s input*

STORE *input to* **num1**

OUTPUT *user for second number*

READ *the console for user’s input*

STORE *input to* **num2**

SET **result** *to* **num1** *+* **num2**

OUTPUT **result** *to console*

Temperature Converter

Write a program that asks for a temperature, then reads from user. Then asks whether to convert to Celsius or Fahrenheit. Finally calculate the conversion, and print to screen

SET **safteyNet\_1** *is* false

WHILE **safetyNet\_1** *is* false

OUTPUT *user to convert from either 1 - (Celsius -> Fahrenheit), or 2 - (Fahrenheit -> Celsius)*

READ *user’s input from console*

SET *response to* **userOption**

IF **userOption** *is* “1” OR “2”

**safetyNet\_1** *is* true

ELSE

OUTPUT *a message to notify the user of their invalid input*

**END** WHILE

OUTPUT *user’s temperature that they intend to convert*

READ *user’s input from console*

SET *response to* **num**

SWITCH **userOption**

CASE “1”

SET **result** *is* **num** \* 1.8 + 32

OUTPUT **result**

CASE “2”

SET **result** *is* **num** - 32 \* 5/9

OUTPUT **result**

**END** SWITCH

**END** PROGRAM

Odd or Even Guesser

Write a program in that picks a random number between 1 and 200 then asks the user to choose whether it is odd or even and then output the success of their guess

SET **num** *to* RANDOM int between 1 and 200

SET **oddOrEven** *to* **num % 2**

SET **safetyNet** *to* true

OUTPUT *user to guess whether the number is Odd or Even*

WHILE **safetyNet** is true

READ *console for user’s input*

SET **guess** to *user’s input included with an* ***uppercase******function***

IF **guess** is “EVEN” or “ODD”

SET **safetyNet** *to* false

ELSE

OUTPUT *a prompt to notify the user of their incorrect input*

**END** WHILE

SWITCH **guess**

CASE “EVEN”

IF **oddOrEven** is equals 0

OUTPUT *prompt the user that they’re correct!*

ELSE

OUTPUT *prompt the user that they’re incorrect*

CASE “ODD”

IF **oddOrEven** is equals 1

OUTPUT *prompt the user that they’re correct!*

ELSE

OUTPUT *prompt the user that they’re incorrect*

**END** SWITCH

**END** PROGRAM - ***\* I added an additional play again feature in the source code\****

Reverse Guessing Game

The computer guesses a random number between 1 and 100. The player then says if the guess is too high or too low. The game continues until the computer guesses right. If the player says the guess is too high, the computer must remember not to guess any higher. If the player says the guess is too low, the computer must remember not to guess any lower.

BONUS: The computer figures out if the player has given it something impossible.

SET rangeLow *to* 0

SET rangeHigh *to* 100

WHILE *(Computer has yet to guess the answer)*

SET **computerGuess** *to a guessed number from between* rangeLow *and* rangeHigh

OUTPUT **computerGuess**

OUTPUT *Is the number correct, too high, or too low?*

INPUT *Read users input from console, either ‘correct’, ‘too high’, or ‘too low’*

SET *User’s input to* **response**

SWITCH **response**

CASE “correct”

OUTPUT *A celebratory message!*

CASE “too high”

SET rangeHigh *to* **computerGuess**

CASE “too low”

SET rangeLow *to* **computerGuess**

DEFAULT

OUTPUT *Prompt to notify user of their invalid input*

IF **rangeLow** > **rangeHigh**

OUTPUT *Prompt to notify the user that they aren’t playing fairly*

**END** PROGRAM

**END** SWITCH

**END** WHILE

**END** PROGRAM