

Objective	Write a guest program for a hotel.
-----------	------------------------------------

Requirements

You have been given the job of writing a program for Cloudy Skies Resort to track information about the current guest staying in a 300-room hotel:

The program will present the user with a menu as follows:

Cloudy Skies Resort

1. **Check a guest into a room**
2. **Check a guest out of a room**
3. **Display information about a room**
4. **Display number of guest rooms available**
5. **Exit**

Instructions:

Menu: The menu will display at when the program is run and allow selection of the options listed. It must continue to display after each option. The program will end only when the exit option is selected. If you prefer to use letters instead of numbers for menu selection, be sure it accepts both upper and lower case letters.

Option 1: Check a guest into a room:

You can either set it up to get the next available room or let the user enter a room. If you allow them to enter the room then you must verify the room is not already taken before adding the guest.

For the guest check-in you must allow entry of the following:

First Name, Last Name, Number of guests, Number of keys, Check in Date, Check out Date, Price.

Option 2: Check a guest out of a room:

The user will enter a room number and the information about the guest staying in the room will display on screen. You will then

ask if they would like to remove the guest. If the user wants to remove the guest then make the room available.

Project

Option 3: Display information about a room:

The user will be prompted for a room number. If the room is occupied then the information stored for the guest will be displayed, otherwise a message that the room is available will be displayed.

Option 4: Display number of guest rooms available:

This option will print out the number of available guest rooms, those which do not have a guest staying in them.

Option 5: This option will exit the program.

The room information can be implemented as a class stored in an array or as multiple parallel arrays. Do not make 300 variables to store each room, you will be marked down if you do this.

Project

Objective

Write a number guessing game using Ruby. – Worth 100 Points

Requirements

You are working at a summer camp named Techisfun Camp, and have been asked to create a game to keep campers occupied. The camp counselors want you to write the following game:

Write a guessing game where the program selects a random number from 1 to 10 and the user gets three attempts to guess it. If, after a given attempt, the user does not guess correctly, display a hint using the following rules:

* user's guess is 3 or more values away, display "cold" * user's guess is 2 away, display "warm"

* user's guess is 1 away, display "hot"

If the user guesses the right number in three attempts or less, print out a winning message; otherwise, print out a failure message.

Repeat the entire process another nine times so that the user plays ten rounds.

Keep track of the total number of rounds the user wins.

After the tenth round, the program should print out how many of the ten rounds were won and how many were lost. Then, describe the skill level of the player as follows:

* Players who win 7 or fewer rounds are rated as "amateurs" *
Players who win 8 rounds are rated as "advanced"

* Players who win 9 rounds are rated as "professionals"

* Players who win all 10 rounds are rated as "hackers"

Hints:

- Get a random number between 1 and 10 then set it to secret number. You will need to put this in the correct

location, so be sure to think where in your program you need to get either the first number or another number since you need a different number for each of the 10 rounds.

- To find how far away from the secret number the users guess is you may want to find the absolute value. You can do this by either multiplying the number by a -1 if it

Project

is negative or looking up how to find the absolute value

using Ruby (Do a search for ruby absolute value).

- You may want to setup a nested loop with the outer loop

tracking the 10 rounds and the inner loop tracking the 3 guesses per round.

Your program should meet the technical requirements listed below:

1. Make sure your program is adequately commented, uses consistent naming conventions for variable and module names, and uses appropriate indentation and line spacing to enhance the readability of the code.
2. **Name the program Final.rb**

