1. Explain when it is advisable to use the following repetition control structures
2. **For Loop**-In situations where you need to repeat a block of code a specific number of times, the for loop is your go-to tool. It's designed to handle tasks with a set number of iterations. It kicks off by setting up a control variable, which is the heart of the loop. As the loop runs, it checks a certain condition related to this control variable. The loop keeps going as long as this condition remains true, with each iteration dutifully executed. The control variable gets updated every time the loop runs, ensuring progress. This makes the for loop ideal for tasks like going through arrays, handling collections with a known number of elements, or simply iterating over a set range of values.
3. **While Loop**-When you're dealing with situations where the number of iterations is uncertain and depends on changing conditions, the while loop is your ally. This loop doesn't work with a fixed number of iterations. Instead, it starts by evaluating a condition. If the condition is true, the loop's body runs. After each run, it checks the condition again to decide if it should continue. This makes the while loop perfect for scenarios where you need to keep going until a certain condition is met or changes, like waiting for user input or reading data until there's none left. The while loop excels in adapting to these changing situations.
4. **Do while Loop** -The do-while loop stands out with a unique feature: it ensures that the loop's body is executed at least once before any condition check. This loop jumps right into action, running the loop's body first before pausing to consider the condition. Only after this initial run does it check whether to continue with more iterations. This is especially useful in cases where you need to perform an action at least once, regardless of other conditions, like showing a menu in a user interface before reacting to user input. The do-while loop is invaluable for ensuring that initial action takes place.
5. Explain the differences between while loop and do while loop

a. The while loop is structured to assess the condition before the loop's body takes action. If the condition is false right off the bat, the loop's body won't even get a chance to execute. This trait of the while loop is particularly handy in scenarios where running the loop might not be necessary, depending on the initial condition. In contrast, the do-while loop follows a different approach. It runs the loop's body first and then checks the condition. This approach ensures that the loop's body is executed at least once, irrespective of the condition's initial state. This unique feature of the do-while loop is perfect for cases where at least one loop execution is mandatory, with subsequent iterations dependent on a condition evaluated post the first run.