

ser usado apenas dentro do corpo de um loop. Usá-lo em qualquer outro lugar causa um erro de sintaxe.

Quando a declaração continua é executada, a iteração atual do loop fechado é encerrado e a próxima iteração começa. Isso significa coisas diferentes para diferentes tipos de loops:

Em um loop de tempo, a expressão especificada no início do loop é testado novamente e, se for verdade, o corpo do loop é executado a partir do topo.

Em um loop de fazer/enquanto, a execução pula para o fundo do loop, onde a condição do loop é testada novamente antes de reiniciar o loop no topo.

Em um loop for, a expressão de incremento é avaliada e a expressão do teste é testada novamente para determinar se outra iteração deve ser feita.

Em um para/de ou para/em loop, o loop começa com o próximo valor iterado ou o próximo nome da propriedade sendo atribuído ao variável especificada.

Observe a diferença de comportamento da declaração continuação no enquanto e para loops: um loop de tempo retorna diretamente à sua condição, mas um loop for primeiro avalia sua expressão de incremento e depois retorna à sua condição. Anteriormente, consideramos o comportamento do loop for termos de um "equivalente" enquanto loop. Porque continuam

A declaração se comporta de maneira diferente para esses dois loops, no entanto, não é realmente possível simular perfeitamente um loop com um pouco de loop sozinho.