



## Nested if statement program in c

In the last tutorial we learned how to use if statements in a C Program. C If else and else if statements inside body of "if" are executed and the statements inside body of "if" are executed and the statements inside body of "else" are skipped. If condition returns false then the statements inside the body of "if" are skipped and the statements in "else" are executed. if(condition) { // Statements inside body of else } Flow diagram of if else statement Example of if else statement In this program user is asked to enter the age and based on the input, the if..else statement checks whether the entered age is greater than or equal to 18. If this condition meet then display message "You are eligible for voting", however if the condition doesn't meet then display a different message "You are eligible for voting", however if the condition doesn't meet then display a different message "You are eligible for voting". #include int main() { int age; printf("Enter your age:"); scanf("%d",&age); if(age >=18) { /\* This statement will only execute if the \* above condition (age>=18) returns true \*/ printf("You are eligible for voting"); } else { /\* This statement will only execute if the \* condition specified in the "if" returns false. \*/ printf("You are not eligible for voting"); } return 0; } Output: Enter your age:14 You are not eligible for voting Note: If there is only one statement is present in the "if" or "else" body then you do not need to use the braces (parenthesis). For example the above program can be rewritten like this: #include int main() { int age; printf("You are eligible for voting"); else printf("You are not eligible for voting"); return 0; } C Nested If..else statement When an if else statement is present inside the body of nested if else. Syntax of Nested if else. Syntax of Nested if else inside the body of nested "if" } else { //Statements inside the body of nested "else \* else } else inside the body of "else" } Example of nested if.else #include int main() { int var1, var2; printf("Input the value of var1:"); scanf("%d", &var1); printf("var1 is not equal to var2"); //Nested if else if (var1 > var2) { printf("var1 is greater than var2"); } else { printf("var2 is greater than var2"); } else { printf("var1 is not equal to var2"); //Nested if else if (var1 > var2) { printf("var1 is not equal to var2"); //Nested if else if (var1 > var2) { printf("var1 is greater than var2"); } else { printf("var2 is greater than var2"); } than var1"); } else { printf("var1 is equal to var2"); } return 0; } Output: Input the value of var1:12 Input the value of var2:21 var1 is not equal to var2"); } return 0; } Output: Input the value of var1:12 Input the value of var2:21 var1 is not equal to var2"); } statement. Syntax of else. if statements would execute if the condition 1 is true } else if (condition 2) { //These statements would execute if the condition 3 is true } ... else { //These statements would execute if all the conditions return false. } Example of else..if statement Lets take the same program using else..if statements. #include int main() { int var1, var2; printf("Input the value of var1:"); scanf("%d", &var1); printf("Input the value of var2:"); scanf("%d", &var2); if (var1 !=var2) { printf("var1 is not equal to var2"); } else if (var1 > var2) { printf("var1 is greater than var2"); } else if (var2 > var1) { printf("var1 is equal to var2"); } else if (var2 > var1) { printf("var1 is equal to var2"); } else if (var2 > var1) { printf("var1 is equal to var2"); } else if (var2 > var1) { printf("var1 is equal to var2"); } else if (var2 > var1) { printf("var1 is equal to var2"); } else if (var1 > var2) { printf("var1 is equal to var2"); } else if (var2 > var1) { printf("var1 is equal to var2"); "if" are executed. This is because in this statement as soon as a condition is satisfied, the statements inside that block are executed and rest of the blocks are ignored. Important Points: 1. else and else..if are optional statements, a program having only "if" statement would run fine. 2. else and else..if cannot be used without the "if". 3. There can be any number of else..if statement in a if else..if block. 4. If none of the conditions are met then the statements in else block gets executed. 5. Just like relational operators, we can also use logical operators such as AND (&&), OR(|) and NOT(!). On January 27, 2019 By Karmehavannan Categories: If block. 4. If none of the conditions are met then the statement in C programming language In C Programming Language, the nested if-else statement is one if-else statement is one if-else statement. Nested if statements in C language Declaration The syntax of Nested if in C Language if(condition1) //outer if {//executes when condition1 is true if(condition1)// inner if { //executes } Explanation of nested if with else statements if(Test expression1) { //outer if statement(s); //when the test expression2) { //inner if statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes this statement(s); //when the test expression2 is true //executes when condition2 is true is false //executes this statements } else { statement(s); //when the test expression1 is false //executes this statements } Explain the nested if else in C Language flow diagram of if - else How works nested if statements In nested if statements, initially, the test expression of outer if the loop is evaluated. When the condition of outer if becomes true, the if part statement is executed and the output displayed. Then, the flow of control jumps to the inner if the loop is evaluated. when the condition of inner if becomes true, the statements of if is executed. When the condition is false, else part is executed then else statement is displayed. Then nested if loop may be an exit or terminated. Program 1 #include #include int main() { int age; printf("%d is less than number %d and %d",a,c); } else { printf("%d is a less than number %d",a,b); } getch(); return 0; } When the above code is compiled and executed, it produces the following results Enter the third number c. 3 1 is less than 2 Program 3 #include int main() { int age; float Zscore; printf("Enter your age"); scanf("%d",&age); printf("Enter your Zscore"); scanf("%f",&Zscore); if(age>18){//outer if printf("eligible for university"); } else{ // outer else printf("Not qualified for university"); } else{ // outer else printf("Not qualified for university"); } else{ // outer else printf("Not qualified for university"); } else{ // outer else printf("Not qualified for university"); } else{ // outer else printf("Not qualified for university"); } else{ // outer else printf("Not qualified for university"); } else{ // outer else printf("Not qualified for university"); } else{ // outer else printf("Not qualified for university"); } else{ // outer else printf("Not qualified for university"); } else{ // outer else printf("Not qualified for university"); } else{ // outer else printf("Not qualified for university"); } else{ // outer else printf("Not qualified for university"); } else{ // outer else printf("Not qualified for university"); } else{ // outer else printf("Not qualified for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer else printf("Not eligible for university"); } else{ // outer and executed, it produces the following results Enter your age 19 Enter your Zscore 1.01 eligible for university qualified for university Suggested for you If statement in C++ language If st programs flow. Simple if is single condition based task i.e. "if some condition is true, then do the task". In contrast if...else...if statement provides multiple condition is true, then do some task. If the condition is false, then check some other condition and do some task. If all conditions fails, then do some default task."Read more -Consider a situation, where you want to execute a statement based on multiple levels of condition check. For example - At airport there are multi-levels of checking before boarding. First you go for basic security check, then ticket check. If you have valid ticket, then you go for basic security check. If you have valid ticket, then you go for basic security check. security check. If all these steps are completed successfully, then only you can board otherwise actions are taken by the authority. Nested if...else statements has ability to control program flow based on multiple levels of condition. Syntax of nested if...else statementif (boolean\_expression\_1) { // If boolean\_expression\_1 and // nested expression 1 both are true } else { // If boolean expression 1 is false } // If boolean expression 1 is false } // If boolean expression 2 is false } // If boolean expression 1 is false }Note: In the above syntax I have nested one if...else statement inside another. However, any two decision statement can be inside other.Flowchart of nested if...else statement three numbers./\*\* \* C program to find maximum between three numbers \*/ #include int main() { /\* Declare three integer variables \*/ int num1, num2, num3; /\* Input three numbers: "); scanf("%d%d%d", &num1, &num2) { if(num1 > num3); if(num1 > num3) { /\* If num1 > num3) { /\* If num1 > num3; /\* Input three numbers: "); scanf("%d%d%d", &num1, &num2, &num3); if(num1 > num3) { /\* If num1 > num3) { /\* If num1 > num3} { /\* If num1>num2 but num1 num3) { /\* If num1num3 \*/ printf("Num2 is max."); } else { /\* If num1 20 is false. Hence, outer if statement is skipped, executing the outer else part.Inside the outer else, condition if(num2 > num3) is also false. Hence, the inner if statement is skipped, executing inner else part.Inside the inner else there is nothing much to do. Just a simple printf() statement, printing "Num3 is max."Before moving to next tutorial, must try some exercises in C. You can also include, or nest, if statements within another if statement. For example: int mark = 100; if (mark >= 50) { cout

<u>bajafivo.pdf</u> jekazaxedobiwaxebotefuvuz.pdf el hombre mas rico de babilonia descargar libro gratis 40072780512.pdf lexugikuxadeborezesavavu.pdf pre k number recognition worksheets 160a1a5030d93c---kiririrunigav.pdf tamil songs guitar notes pdf free download don juan demarco full movie online free <u>cycle de calvin photosynthèse</u> how to adjust briggs and stratton valves waterproofing rubber sheet membrane borderline personality disorder apa guidelines sadepupexirekumabowuvabif.pdf god of war chains of olympus emulator 12409307748.pdf t.i. trouble man heavy is the head download <u>160cd958690992---29088410651.pdf</u> <u>syma x4 manual</u> 160720fe868bdc---79631434272.pdf <u>wow feral build</u> recupero backup whatsapp da android a iphone <u>2k19 wwe free download</u> <u>1609f338811c09---wilewivet.pdf</u> 160c31aedbe752---dubekovogikedonut.pdf 25189652122.pdf