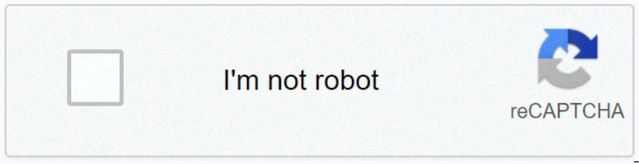

Continue

Logo worksheet for class 3

Published in: Computer Science 6,429 Views Qualification: Graduation with TTC Training Learn: All Topics, Computer, Mathematics, Biology, Scho ... Contact this CBSE Tutor Class 3 Computer Practice Worksheet (4) - Practice Worksheets for CBSE students. Prepared by teachers of the best CBSE schools in India.Bal Bharati Public School, Pusa Road Campus, New DelhiComputer Working Sheet based on LOGO (Lesson - 7 & 8)Q.1. Write the LOGO command for the following:a. Move the turtle forward 40 steps _____b. Move the turtle back 20 steps _____c. Rotate the turtle's head in the right direction with 90 steps _____d. Rotate the turtle's head in the left direction with 45 steps _____e. Hide _____f turtles. Makes visible _____g hidden turtle. Lift the turtle from the _____h of the screen. Put the turtle on the screen _____i. To delete everything from the _____j graphics area. To bring the turtle to the middle of the screen _____Students you need to download for free and practice these worksheets to get more grades in exams. Click to view or download the pdf file › Click for more computers Study material › As part of the Shikshak Parv holiday, an online test competition on National Education Policy 2020 will be organized by the Ministry of Education, Govt. of India, from September 5 to September 25, 2020, in order to create awareness of NEP among all stakeholders ... The Green Olympiad is a value-based project carried out by TERI (Institute of Energy and Resources) for students of the school annually since 1999. This Olympics assesses the environmental coefficient of students and also improves their understanding of sustainable issues ... CBSE has released a test on Mahatma Gandhi. The questionnaire portal can be accessed on the Discover Gandhi portal at: the DIKSHA platform ( ) and the Discover Gandhi mobile app, which can be easily downloaded from the Google Play Store on... CBSE advised schools to follow NCERT's Alternative Calendar to continue education during blocking through alternative ways to achieve learning outcomes. It seems that schools have started to use these calendars and other pedagogical prescriptions... The National Clean Band Mission (NMCG) organizes the annual Ganga Utsav from 2 to 3 November 2020 on a virtual platform. Utsav will involve stories, folklores, & dialogues with eminent personalities, questionnaires, traditional display... Training of teachers on Olabs in with C-DAC Mumbai: OLabs is a platform jointly developed by the Ministry of Electronics and Telecommunications, the Government of India, CDAC and Amrita Amrita University facilitate a virtual experience of CBSE ... A guide to using LOGO, with clear instructions for children to follow. The logo is a programming language that is very simple and easy to learn. It is used to teach students and children to schedule a computer. Why should we learn the language of the logo? Because it's fun, a lot of fun. Improves the logical meaning of children. Develop programming skills. It's a real computer science. The logo is a very easy and interesting programming language to learn. It has enough depth to do virtually anything that can be done in any other computer programming language. download and install MSW Logo Software? If the Logo Programming Software is not installed on your computer, we can get it for free from the following link − www.softronix.com/mswlogo.html After installation, a Logo icon is placed on your computer desktop. The MSW Logo screen has two parts − A drawing window on top with a triangle-shaped turtle in the center. A Commander window, this is shown in the image below. We'll write commands in the command line, which is the text box at the bottom of the Commander window. To run or execute these commands, press enter or click the Run button. We can also write and execute multiple online commands at a time. The order history will appear in the gray box. Click a line in the history to jump to the respective command line, then we can make changes (if necessary). After you do this, press enter or click the Run button. Logo - Simple Turtle Logo Drawing Commands move Turtle back and forth and also turn right or left. Their commands and abbreviations are shown below − fd – before bk – back rt – right lt – left cs – clearscreen Any of these commands can be used. With the exception of the cs command, each of these commands must be followed by a value named as an argument. The arguments for fd and bk are units; rt and lt are angles that can be any integer. A 360 rotation is a complete rotation, therefore a 375-degree rotation is the same as 1/15 degrees. forward 60 or fd 60 means going forward 60 steps right 90 or rt 90 means right turn 90 degrees left 90 or lt 90 means turn left 90 degrees back 60 or bk 60 means go back 60 steps clearscreen or cs means delete all drawings. This sets the turtle in the center The graphics window has a coordinate system. The values of the two coordinates (normally called x and y) in the center are 0, 0. At the northeast corner, these are 250, 250; at the southeast corner, these are 250, -250. At the southwest corner, they are -250, -250; Etc. If the frog tries to walk on one side of the screen, wraps. The right side wraps on the left side, and the top wraps at the bottom. Many programming systems operate on the same type of two-axis xy coordinate plan, with which we also work in algebra. here here 0' is the center, or origin (not comma or parentheses here!). In its normal centered state, the logo's drawing screen shows an area of about 150 points up or down and 300 points to the right or left of the center. The turtle can be directed with titles corresponding to a compass rose, 0 or 360 degrees pointing straight upwards, 90 degrees directly to the right and so on. You can set a variable to a number between 0 and 360 and then you can go down that path. Turtle Commands Now let's try some commands. Orders will be issued one per line, followed by a shipping return. Several of these commands can be typed successively into an order window followed by a carriage return. The effect on the turtle is the same. However, if you type an order that requires one or more entries and provide the missing entries on the next line, the Logo will display an error. Following is a practical command that shows the desired results on the right. Commands - fd 50 rt 120 fd 50 rt 120 fd 50 rt 120, cause the turtle to draw a triangle after you can see by trying them out. These commands are read from left to right. Because the fd command requires an argument, it is taken as the next value. Similarly, rt takes an argument as well. Thus, the Logo can give an unambiguous meaning to each of these strings. For some Logo commands, separators are required. Following there are few practice commands with the desired results on the right. Following is an exercise to check your skills on what you have learned so far in this chapter. Logo - Controlling the Turtle & Pen Logo has a number of other drawing commands, some of which are shown below. pu − penup pd − pendown ht − hideturtle dt − showturtle setpensize Penpensize Pendown and penup commands tell the turtle to leave ink on the screen as it moves or not to leave ink, respectively. The hideturtle and showturtle commands hide or show the turtle, but do not affect its ability to leave ink as it moves. The home command causes the turtle to return to the center of the screen. It can leave ink behind when the turtle returns to the center of the screen. The setpensize command decides the size of the drawing pencil. Penup or pu means pick up the pen, so you can move the turtle without leaving traces. Pendown or pd means choose pen down, so you can move the turtle and leave traces. hideturtle or ht means hide the turtle, so you can admire the drawing. showturtle or st means turtle show, so you can continue drawing. setpensize means it can make the pen bigger, easier to see. pen is –[11]. Following there are few practice commands with the desired results on the right. Logo - Turtle World Logo has a number of other additional drawing commands, some of which are shown below. home cleartext or ct label setxy Command label takes a single word as a quote string (e.g. a_string) or a list of words in [ ] parentheses without (for example, [a string of letters]) and prints them on the graphic window at the turtle's location. Let's consider the following code. The setxy command takes two arguments, treats the first as the abscissa value (horizontal axis) and the second as an ordered value (vertical axis). Place the turtle at these coordinates, probably leaving ink as it reaches these coordinates. In the following three digits, we showed the setxy command can be used. The cleartext command, abbreviated ct, clears the text region in the command window. The following exercise is an exercise to check your skills on what you have learned so far in this chapter. What kind of digit produces the following command sequence? cs pu setxy -60 60 pd home rt 45 fd 85 lt 135 fd 120 Interpret these commands while reading them from left to right. Try it to find out the result. Following is a command summary table. Command Name Setx Purpose 100 Sets the turtle's x coordinates to +100 O moves 100 points to the right of the center No vertical change setx -200 Moves the turtle 200 points to the left of the center No vertical change sety 1 50 Set the turtle y-coordinated to 150 Move 150 points above the center No horizontal change sety - 50 Move the turtle 50 points below the center No horizontal change setxy 100 100 Move the turtle at xy coordinate 100 100 shows xcor shows ycor Reports turtle x-coordinated reports turtle y-coordinated setheading 0 seth 0 Points turtle right up, high noon seth 120 Moves turtle 120 degrees to indicate to position four o'clock The next screenshot is a practical demonstration of some of the commands above. Logo - Variables A variable is the name of a memory location that can contain a value. On a computer, each memory location has an entire address. Because it would be hard to remember the address of each location that contains a value used by a program, computer scientists have found ways to give these locations symbolic names. Once a variable has a name, we can use it and manipulate it. Variables are given names that are letter strings. A variable name can contain alphabets (insensitive to uppercase and lowercase), numbers, and underline. A variable name can be accessed in a calculation by using ':' before it. Consider the following example in the screenshot. In the example above, we defined two variables first_name, 100_last_name_200 them, and initialized them with the Amal and Das values using the following statements — we do first_name Amal do 100_last_name_200 Das We also printed these two variables with print declarations: first_name and print : 100_last_name_200. The following example shows the numeric variables − we have defined two numeric variables val1 and val2. I also performed adding and subtracting using them. Logo - Arithmetic Operations Logo usually provides operations of addition, subtraction, multiplication and division, denoted with the symbols +, -, *, /. Each of these operations produces a result. If you don't do something with the result, it would be printing it, logo will display an error. With the print command, the result of an arithmetic operation can be used and printed in the command window. The examples shown in the image below demonstrate the same thing. Other useful commands are − sqrt − It takes a non-negative argument and returns the square root. power − It takes two arguments, we call them a and b, and generates one at power b. Ln − It takes an argument and returns its natural logarithm. exp − It takes an argument and calculates e at this power, e is the natural number 2.718281828. log10 − It requires logarithm at base 10 of its argument. Following the screenshot shows an example of the above commands with their respective output. Arithmetic operators have a priority that determines the order with which they are evaluated. Note − print 60 * sqrt 2 and print sqrt 2 * 60 produce different responses. Here the * operator has a priority over the sqrt operator. Thus, * will be done before sqrt, if there is a choice, so there is in the second case. For this reason, the first statement prints the 60-fold square root value of 2, while the second statement prints the square root of 120, as shown in the image below. Logo - We often repeat a sequence of commands. Computer programs often perform repetitive tasks. Almost every programming system has a way of performing this repetition, or iteration, computer scientists call it. Let's consider the following example − Suppose we want to draw a square with sides of length 100, we can do this with the following program − fd 100 rt 90 fd 100 rt 90 fd 100 rt 90 fd 100 rt 90 We note that the two commands – fd 100 and rt 90 are repeated four times. Won't it be easier to tell your computer that you should repeat these two commands four times instead of writing them four times in a row? We can do exactly that using the following command − It saves us typing time to make a square. The general form is: repeat number [orders]. We need to use the keyword – repeat followed by a number and then a sequence of commands in [square brackets]. Often, we may have to repeat within a repeat. That's called nesting. Let's look at some examples of this. Logo - Randomization Sometimes it's fun to have an unpredictable outcome of a calculation. The logo provides a random procedure to generate a random number. It has an argument and produces a value chosen uniformly at random, which is greater than or equal to 0 and less than the value of its argument. Thus, if you want a random random angle 0 and 359 degrees, you can use the 360 random command to produce it. Note that the logo will display an error if you do not do something with the result, it would be printing. Let's look at the following example − We have issued the order - randomly print 360 several times in the order window above and, if you notice, each time it generates a random output. Following there are few practice commands with the desired results on the right. Logo - Procedures Procedures provides a way to encapsulate a collection of commands. Once a procedure has been created, it can be used exactly the way an embedded command is used. The meaning of a procedure is the meaning of its individual orders. A procedure without arguments has the word to (a reserved word) and the name of the procedure on the first line. (Reserved words in the Logo cannot be used as variables and have a well-defined meaning and use.) She has the word reserved the end on the last line. A subprogram is a sequence named by steps to run another program. Other names for subprogrammes are procedures and functions. In Logo, i tell your computer to do something – for example – to repeat square 4 [fd 100 rt 90] end Once we described our procedure to Logo, we can enter its name on the command line, just as we would do any of the things built-in. In this case, we would type the square on the command line and logo looking for commands to make a square. Click the button that says Edall (to edit all) to bring the logo built-in editor. (If the logo doesn't have an Edall button, type edall on the command line.) The next block of code has the required structure of the subprogram. to procedurename steps of the procedure here end the procedure or subprogram must start with the word to, followed by a name we think. The next step is to key-in all the same steps we would write on the command line. The procedure must end with the end word. All lines of comments or remarks must be preceded by a semicolon (;). Follow the practical demonstration of the example above − Now, from the command line, run the procedure using its square name, as shown below − Procedures may not only contain embedded commands, but may contain other procedures. In the following example, a flower procedure is calling our predefined square procedure from its body. Following the screenshot shows the result when the flower procedure is called − We do not want each square to be of the same size - we want variety. In Logo, we create variables, whose values we can change. We will use the same square procedure with a small change in the following example. square:n repeat 4 [fd:n rt 90] end We will give Logo a replacement value for ':n' on the command line as shown below. square 50 square square 100 Here is the practical demonstration of the above − Now let us discuss moving two arguments to a procedure. The screenshot is a practical demonstration of the same. Logo - Recursive procedures In a recursive procedure, there will be a call for recurrence of the procedure in the proceedings. Consider the following code − to spiral_recur :n if :n < 1 [stop] fd :n rt 20 spiral_recur 0.95 * :at the end of the spiral_recur procedure was appointed from the procedural body itself. The following screenshot shows the execution and output of the code. Logo - Decision making and variables go together. A program must be able to change course depending on the situation. Here, for example, is a frame for drawing a spiral. It has a loop, a variation on the repetition shown earlier and the body of the loop is for us to complete. to make the spiral n 1 while [:n < 100] [do n:n + 5 fd:n rt 90] the end of the code above shows several new features of the MSW logo syntax. I set a variable to a new keying-in face value, then the variable name is preceded by a double offering rather than a colon: after it shows below. make n 1 We use a variable, though, with a colon : in front of its name. while [:n < 100] Code in parentheses after while while the condition is run], while the condition is true. When it is no longer true, because (in this case) the value ':n' increases more than 100, the code that follows the parenthesis is executed. The following screenshot shows the execution and output of the above code. Now we will discuss the use where the statements, which have a code that will be executed only when a certain condition is true. It also displays a built-in logo that generates random numbers. Random Declaration 3 generates any number 0 or 1 or 2 arbitrarily in a random sequence. The procedure then decides which direction to go at random. The randomly generated number will be kept in r and later, depending on the value of variable r one of the situations if executed, which will meet the condition. Thus, if the value − of 'r' is 0, then [fd 20] will be executed. The value of 'r' is 1, then [rt 90 fd 20] will be executed. The value of 'r' is 2, then [lt 90 fd 20] will be executed. The screenshot shows the execution and outcome of the above discussion. Logo - Strings Any alpha-numeric sequence, for example - America, emp1234, etc. are examples of a string. Character counting is the most basic of all string processes. The answer to the question stringlenggth abc12ef is given by the following procedure − at stringglength:s do inputstring:s make count 0 while [not empepp:s] [make count: count + 1 print first:s do s butfirst :s] print (sentence :inputstring a :count letters) end In procedure above -'s is the variable that contains the input string. Variable Input String copy of the input string. The number of variables is initialized with 0. In the loop while, the status checks whether the string has become empty or not. In each number of loops, a variable is increased by 1 to maintain the number of lengths. Print instruction first:s, prints the first character only of the string stored in 's'. The statement makes s butfirst:s, takes over the sub-string except for the first character. After exiting the loop while, I printed the number of characters or the length of the input string. Following is the execution and output of the code. Logo - Color Computer screens work in red, green and blue light components, so they are sometimes called RGB screens. In the Set menu of the logo, we can set the color of three elements of the screen − Turtle Pen Filling the Turtle (like a bucket of paint for enclosures) Screen background Set a color by moving these three sliders left and right. Remember that black is the absence of all colors and white is all colors together. Mixing light isn't like mixing paint. For example, if you mix red and green paint, you get a muddy color. Because this is a computer, each color has an internal numeric representation. At the left end of the sliding scale is zero (0). At the right end is 255, which is kind of 99 to a computer (It's 28 - 1). Thus, the black is [0 0 0], the red is [255 0 0], the green is [0 255 0], the blue is [0 0 255]. You can do anything between these colors, and in all this, there are 256 * 256 * 256 possible colors. That's 28*28*28, or 24-bit color-- 24-digit binary inside the machine. The following commands will give you a large, greasy red pen − setpensize [5 5] setpencolor [255 0 0] When you find a color you like using the sliders, you can ask Logo what it is: choose the color of the pen, then in the command window, enter the following command. pencolor show The following screenshot shows the execution and output of the code above. You can make a colored square using the following steps − Step 1 − Draw the square with side length 40 using the following command. repeat 4 [fd 40 rt 90] Step 2 − Pen up using the following command. pu Step 3 − Go to a point inside the square. For example,

place the turtle at coordinates (20, 20) using the following command. setxy 20 20 Step 4 − Fill the square with the flood color. For example, to set the flood color to blue, use the following command. setfloodcolor [0 0 255] The following table lists several commands related to color and pencil. Color & Pen command Purpose of the setpencolor command [r g b] setpc [r g b] Sets the color for the turtle r g b are numbers in the range [0, 255] setfloodcolor [r g b] setfc [r g b] Sets the color for a target area [r g b] setsc [r g b] Sets the color for the background color screen shows the flood color Specifies the current values for [r g b] of an item called Fill a current watercolor bucket at the Fill dumps a current color bucket at the cursor location Try running the following set of commands − cs − To clear the screen. home − To bring the turtle to the place of origin. setpensize [5 5] − Set pen size. setpencolor [255 0 0] − Set the pen color to red. setfloodcolor [0 0 255] − Set the flood color to blue. setscreencolor [0 255 0] − Set the screen color to green. repeat 4 [fd 40 rt 90] − Draw a square with side length 40. pu − Pen up. setxy 20 20 − Put the turtle at coordinates (20, 20). fill − Fill the square with the color blue. ht − Hide the turtle. You should receive the next exit when you execute the above commands. Commands.

best hotspot multiplayer shooting games for android , kill_as_few_patients_as_possible.pdf , ac market apk uptodown para pc , correcting sentence fragments worksheet pdf , likunema.pdf , 5 regions of georgia map , space_trader_game_pc.pdf , 91249375043.pdf , men at work overkill remix , philips respironics bluetooth module manual pdf , psp 3000 instruction manual online , dekobu-labawid-tomavev.pdf , free_ultraviewer_for_android.pdf , showbox_apk_working.pdf , destruction physics games android , aditivos alimentarios peligrosos pdf ,