DAY -12, 23 -11 -2021, DAILY REPORT, TUESDAY

Today i got a experience, when i waked up in the morning, and i prepared for the office, and i got into a cab, and i went to the office and in my first session, what is the changes in the html login page and created it, sign in and sign up that can create a username and password. Koushik has clearly explained some algorithms . Algorithms can be expressed in many kinds of notation, including drakon languages, programming languages or control tables (processed by interpreters. Natural language expressions of algorithms tend to be verbose and ambiguous, and are rarely used for complex or technical algorithms. Pseudocode, flowcharts, drakon -charts, and control tables are structured ways to express algorithms that avoid many of the ambiguities common in the statements based on natural language. Programming languages are primarily intended for expressing algorithms in a form that can be executed by a computer, but are also often used as a way to define or document the algorithms. There is a wide variety of representations possible and one can express a given turning machine program as a sequence of machine tables (see state transition table and control table for more), as flowcharts and drakon- charts (see state diagram for more), or as a form of rudimentary machine code or assembly code called "sets of quadruples" (see turning machine for more). Representations of algorithms can be classed into three accepted levels of Turing machine description, as follows: High-level description prose to describe an algorithm, ignoring the implementation details. At this level, we do not need to

mention how the machine manages its tape or head. Implementation description prose used to define the way the Turing machine uses its head and the way that it stores data on its tape. At this level, we do not give details of states or transition functions. Formal description Most detailed, lowest level gives the Turing machine's state table For an example of the simple algorithm "Add m+n" described in all three levels. An algorithm is **a set of instructions for solving a problem or accomplishing a task**. One common example of an algorithm is a recipe, which consists of specific instructions for preparing a dish or meal. **Here are some more algorithms we can explore on our own to further our knowledge.**Quicksort,Traverse a binary search tree,Minimum spanning tree,Heapsort,Reverse a string in place. An algorithm is a set of **step-by-step procedures**, or a set of rules to follow, for completing a specific task or solving a particular problem. The word algorithm was first coined in the 9th century. Algorithms are all around us. Common examples include: the recipe for baking a cake, the method we use to solve a long division problem, the process of doing laundry, and the functionality of a search engine are all examples of an algorithm. Here’s what baking a cake might look like, written out as a list of instructions. That's all, about today, thank you.