**Some real life examples using stack and quee::**

**Example (1): By using stack:**

A stack of trays in a cafeteria. I also like the pole of rings analogy

With the trays, you can pop (take a tray) and push (return it when you're done). You can see what the top tray (first item) is (ex. what color), and see if the stack is empty. Non-stack operations such as getting a count of the trays are non-trivial. It is also impossible to insert or remove a tray from a random position.

**Example (2):By using stack:**

Other general and common example would be of fire arm magazine

It is easy to see that bullets can be inserted from the top, and only the topmost bullet is accessible. Such is a stack. A magazine works by LIFO (Last In First Out), and so does a Stack.

Furthermore, the four (five1) basic operations of a Stack are applicable:

(1) Push - adding a new (bullet) to the top.

(2) Pop - remove the top (bullet) from the stack (in the analogy, this is like shooting).

(3) Peek - looking at the first bullet of the stack (checking to make sure that the topmost bullet is correctly placed).

(4) Check if Empty - check if the stack is empty (in the analogy, this operation is checking if there are bullets left).

(5) (Initializing the stack creating a new stack of bullets

**Example (3) by using stacks:**

Here I have mentioned the daily life example of web browsing which type of problems occurred and how they can be solved by using stack

When a user visits a new web page, the current page gets pushed onto the stack. When the user clicks the back button, the last page pushed onto the stack is popped off of the stack and loaded in the browser window. When all of the pages are popped off of the stack, the back button grays out, indicating that the stack is empty.

Note that the forward button also uses a stack, but it works on a slightly different basis. Clicking the back button pushes the current page on the stack and clicking the forward button pops the top page off of the stack. Visiting a new page without clicking either button automatically empties the forward button stack.

**Example (1) by using queue.**

If I have six people willing to use a printer all at once .I will have a problem if two of the six attempt to print at the same time. What can I do? I can put up a mark on the printer--"One User At Time Please" and hope everyone sees and follows it. Hence then the users need to coordinate their use.

I can (software) lock it so that only one person can print at any time. But when the second user tries to show he or she is blocked until the printer is free. And the third user is blocked longer etc.

Or I can set up a queue. As each user prints the print job goes to the queue. The printer looks at the queue for jobs to print and as it prints them it removes them from the queue. No one is blocked and the printer doesn't fail because two people try to print at the exact same moment.this would be a fine way for all for betterment service

**Example (2) by using queue.**

Here are the some real life problems that can be solved by using queue

• Purchasing tickets to a concert (e.g. Ticketmaster)

• Grading the students' work in a first-come-first-serve basis

• Online shopping orders for Packaging/Shipping (e.g. e-commerce warehouse)

**Example (3) by using queue.**

The 3rd example is that when we go to submit our utility bills where we forced to make queue .in which we all stand in a standing scenario’s to submit our bills .the person who stand first is treated first …