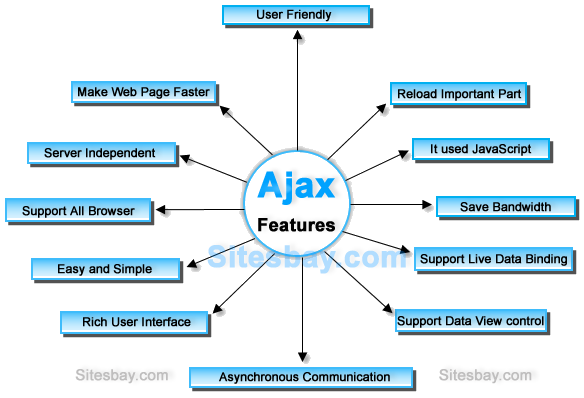
WP- topic: Ajax

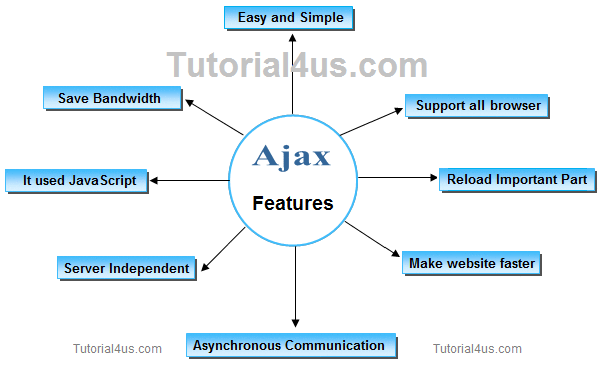
Intro- Rachel

Purpose- Rachel

Features - Shreya

Example (4) – Apurav & Prabhat





You can communicate with server without reloading. Eg if you like the post on Fb it will increase the likes n reload the page but with the help of Ajax u can like the post without reloading.

In non Ajax web application the client sends the request the server decodes it and sends the response n the whole page is reloaded. It reloads the whole page every time a request is sent. In ajax web application the client sends the request and only partial page is refreshed or updated. This saves time n the amount data transmitted between client and server.

The google chrome web browsers inspect element shows the refreshing of the content of the web page.

**User Friendly**

AJAX system enables web applications in a faster, responsive, and user-friendly way. It helps in retaining user engagement by allowing multiple updates on a webpage at a given time.

Makes web page faster:

The purpose of AJAX is to increase the performance, speed, and usability of [**web applications**](https://status200.net/developing-a-web-app/). So, the AJAX technique reduces the server traffic inside requests and lowers the time consumption on both side’s responses.

**Bandwidth Usage**

The bandwidth usage is one of the advantages of AJAX that helps to improve performance as well as speed. Rather than transmitting the whole page’s content, this technology helps to fetch partial contents so as to make the best use of the Server’s bandwidth.

AJAX is the most viable Rich Internet Application (RIA) technology so far. It is getting tremendous industry momentum and several tool kit and frameworks are emerging. But at the same time, AJAX has browser incompatibility and it is supported by JavaScript, which is hard to maintain and debug.

Data binding-AJAX helps you to load data in the background and display it on the webpage, without reloading the complete webpage. Data binding is **the process that establishes a connection between the app UI and the data it displays**.

 AJAX is compatible with asp.net, J2EE, PHP or any other language. It supports almost all popular browsers such as IE5 and above, Firefox 1.0 and above, Safari 1.2 and above, opera 7.6 and above, and RockMelt.

* AJAX is a web browser technology independent of web server software.
* A user can continue to use the application while the client program requests information from the server in the background.

AJAX, which stands for asynchronous JavaScript and XML, is **a technique that allows web pages to be updated asynchronously**, which means that the browser doesn't need to reload the entire page when only a small bit of data on the page has changed. AJAX passes only the updated information to and from the server.

<https://www.tutorialspoint.com/ajax/what_is_ajax.htm>

SE

Historical Perspective of Testing- Shreya

Definitions of Testing - Rachel

Important Features of Testing – Apurav & Prabhat

//[Glenford J. Myers](https://en.wikipedia.org/wiki/Glenford_J._Myers" \o "Glenford J. Myers) initially introduced the separation of debugging from testing in 1979.[[14]](https://en.wikipedia.org/wiki/Software_testing#cite_note-Myers_1979-14) Although his attention was on breakage testing ("A successful test case is one that detects an as-yet undiscovered error."[[14]](https://en.wikipedia.org/wiki/Software_testing#cite_note-Myers_1979-14): 16), it illustrated the desire of the software engineering community to separate fundamental development activities, such as debugging, from that of verification.

Software testing didn’t evolve in a single day; it took time and sweat to get it where it is today. Testing gurus like Hetzel and Dave Gelprin divide testing into five significant eras:

**Debugging-oriented era**: This phase was during the early 1950s, when there was no distinction between testing and debugging. The focus was on fixing bugs. Developers used to write code, and when faced with an error would analyse and debug the issues. There was no concept of testing or testers.

1. **Demonstration-oriented era**: From 1957 to 1978, the distinction between debugging and testing was made and testing was carried out as a separate activity. During this era, the major goal of software testing was to make sure that software requirements were satisfied. As an example, the requirement might have been ‘We need a web application that displays a list of 10 products only’. Testers used to make sure that only 10 products were displayed. This failed because of the probability that a software’s function decreases as testing increases, i.e. the more you test, the more likely you'll find a bug. The concept of negative testing (or breaking the application) was not practiced in this era.

Negative testing **ensures that your application can gracefully handle invalid input or unexpected user behavior**.

1. **Destruction-oriented era**: From 1979 to 1982, the focus was on breaking the code and finding the errors in it. It was Glenford J. Myers who initially introduced the separation of debugging from testing in 1979 although his attention was on breakage testing. (‘A successful test case is one that detects an as-yet-undiscovered error.’) It illustrated the software engineering community’s desire to separate fundamental development activities, such as debugging, from verification. As an example, a tester would test software in such a way that it would break (e.g. entering letters in a field that should only accept numbers). There was no defect prevention approach during this phase. However, the destruction-oriented approach also failed because software would never get released because you could find one bug after another. Also, fixing a bug could also lead to another bug.

[**Testing**](https://www.geeksforgeeks.org/software-testing-basics/)**:** Testing is the process of verifying and validating that a software or application is bug free, meets the technical requirements as guided by its design and development and meets the user requirements effectively and efficiently with handling all the exceptional and boundary cases. [**Debugging**](https://www.geeksforgeeks.org/software-engineering-debugging/)**:** Debugging is the process of fixing a bug in the software. It can defined as the identifying, analyzing and removing errors.

1. **Evaluation-oriented era**: From 1983 to 1987, the focus was on evaluating and measuring the quality of software. Testing improved the confidence index on how the software was working. Testers tested until they reached an acceptable point, where the number of bugs detected was reduced. This was mainly applicable to large software.
2. **Prevention-oriented era**: 1988 to 2000 saw a new approach, with tests focusing on demonstrating that software met its specification, detecting faults and preventing defects. Code was divided into testable and non-testable. Testable code had fewer bugs than code that was hard to test. In this era, identifying the testing techniques was the key. The last decade of the 20th Century also saw exploratory testing, where a tester explored and deeply understood the software in an attempt to find more bugs.

Exploratory testing is **an approach to software testing that is often described as simultaneous learning, test design, and execution**.

<https://salsadigital.com.au/insights/a-brief-history-of-software-testing>

DBMS

Inventory management System Database application- Apurav

Canonical Cover of FDs with algorithm - Prabhat

Organization of Records in Files in database - Rachel

Records in pl/sql – Shreya

What is inventory database? Inventory database is a centralized repository for all inventory data in an organization. Database for inventory management software allows balancing inventory costs and risks against the desired inventory performance metrics.

To maximize the benefit of an inventory database, provide a wide variety of information about the inventory that goes beyond what it is. Include the date it was created or purchased and who placed the order. Include the cost of each item, the location of the warehouses where it’s stored if you have more than one, the specific customer or distribution location for which it’s intended and its shipping cost. If you used credit to pay for the order or production, include the weekly or monthly interest the inventory costs you. When you sell inventory, do not delete it from your database. Mark it as sold and note the date, salesperson, customer, selling price and profit.

//A **record** is a data structure that can hold data items of different kinds. Records consist of different fields, similar to a row of a database table.

For example, you want to keep track of your books in a library. You might want to track the following attributes about each book, such as Title, Author, Subject, Book ID. A record containing a field for each of these items allows treating a BOOK as a logical unit and allows you to organize and represent its information in a better way.

PL/SQL can handle the following types of records −

* Table-based
* Cursor-based records
* User-defined records

AP topic: Cordova setups, plugins and example

Cordova- Prabhat

Plugins- Rachel

Example (2)- Shreya & Apurav

Maths

The complimentary Function, 1 example- Rachel

The inverse operator 1/f(D), the symbolic expiration for the particular integral 1/f(D) X, 1 example- Shreya

Real life application of complimentary & inverse- Apurav & Prabhat

Scilab (optional)- jisko mila uska

F(D)y=Qx

[ 1/f(d)\* Qx=P.i ]

Case 1:

Qx= X

1/D \*x =>