Unit 1

Note: The material to prepare this presentation has been taken from internet and are generated only for students reference and not for commercial use.

Outline

- Web Services
- EJB

Web Services

Outline- Web Services

- Web Services: Overview,
- Types of WS,
- Difference between SOAP and

Web Services: Overview

- A **Web Service** is can be defined by following ways:
  - is a client server application or application component for communication.
  - method of communication between two devices over network.
  - is a collection of standards or protocols for exchanging information between two devices or application.
  - Is available over the Internet or private (intranet) networks
  - Uses a standardized XML messaging system
  - Is not tied to any one operating system or programming language
  - Is self-describing via a common XML grammar
  - Is discoverable via a simple find mechanism

Components of Web Services

- The basic web services platform is XML + HTTP. All the standard web services work using the following components −
  - SOAP (Simple Object Access Protocol)
  - UDDI (Universal Description, Discovery and Integration)
  - WSDL (Web Services Description Language)
- **How Does a Web Service Work?**
  - A web service enables communication among various applications by using open standards such as HTML, XML, WSDL, and SOAP. A web service takes the help of −
    1. XML to tag the data
    2. SOAP to transfer a message
    3. WSDL to describe the availability of service.

Components of Web Services

- **SOAP**
  - SOAP is an XML-based protocol for exchanging information between computers.
  - SOAP is a communication protocol.
  - SOAP is for communication between applications.
  - SOAP is a format for sending messages.
  - SOAP is designed to communicate via Internet.
  - SOAP is platform independent.
  - SOAP is language independent.
  - SOAP is simple and extensible.
  - SOAP allows you to get around firewalls.
  - SOAP will be developed as a W3C standard.

Components of Web Services

- **WSDL**
  - WSDL is an XML-based language for describing web services and how to access them.
  - WSDL stands for Web Services Description Language.
  - WSDL was developed jointly by Microsoft and IBM.
  - WSDL is an XML based protocol for information exchange in decentralized and distributed environments.
  - WSDL is the standard format for describing a web service.
  - WSDL definition describes how to access a web service and what operations it will perform.
  - WSDL is a language for describing how to interface with XML-based services.
  - WSDL is an integral part of UDDI, an XML-based worldwide business registry.
  - WSDL is the language that UDDI uses.
  - WSDL is pronounced as 'wiz-dull' and spelled out as 'W-S-D-L'.

Components of Web Services

- **UDDI**
  - UDDI is an XML-based standard for describing, publishing, and finding web services.
  - UDDI stands for Universal Description, Discovery, and Integration.
  - UDDI is a specification for a distributed registry of web services.
  - UDDI is platform independent, open framework.
  - UDDI can communicate via SOAP, CORBA, and Java RMI Protocol.
  - UDDI uses WSDL to describe interfaces to web services.
  - UDDI is seen with SOAP and WSDL as one of the three foundation standards of web services.
  - UDDI is an open industry initiative enabling businesses to discover each other and define how they interact over the Internet.

Outline- Web Services

- Web Services: Overview,
- Types of WS,
- Difference between SOAP and

Types of Web Services

SOAP Web Services

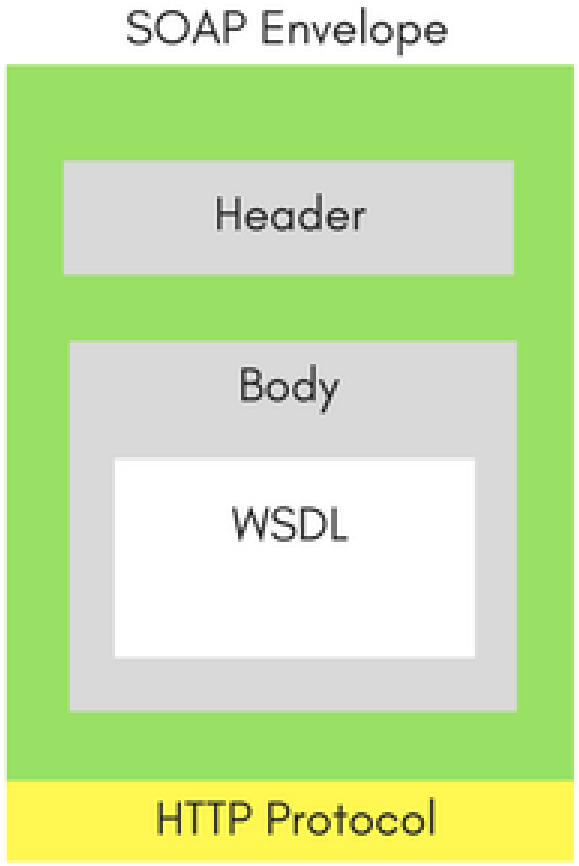
REST Web Services

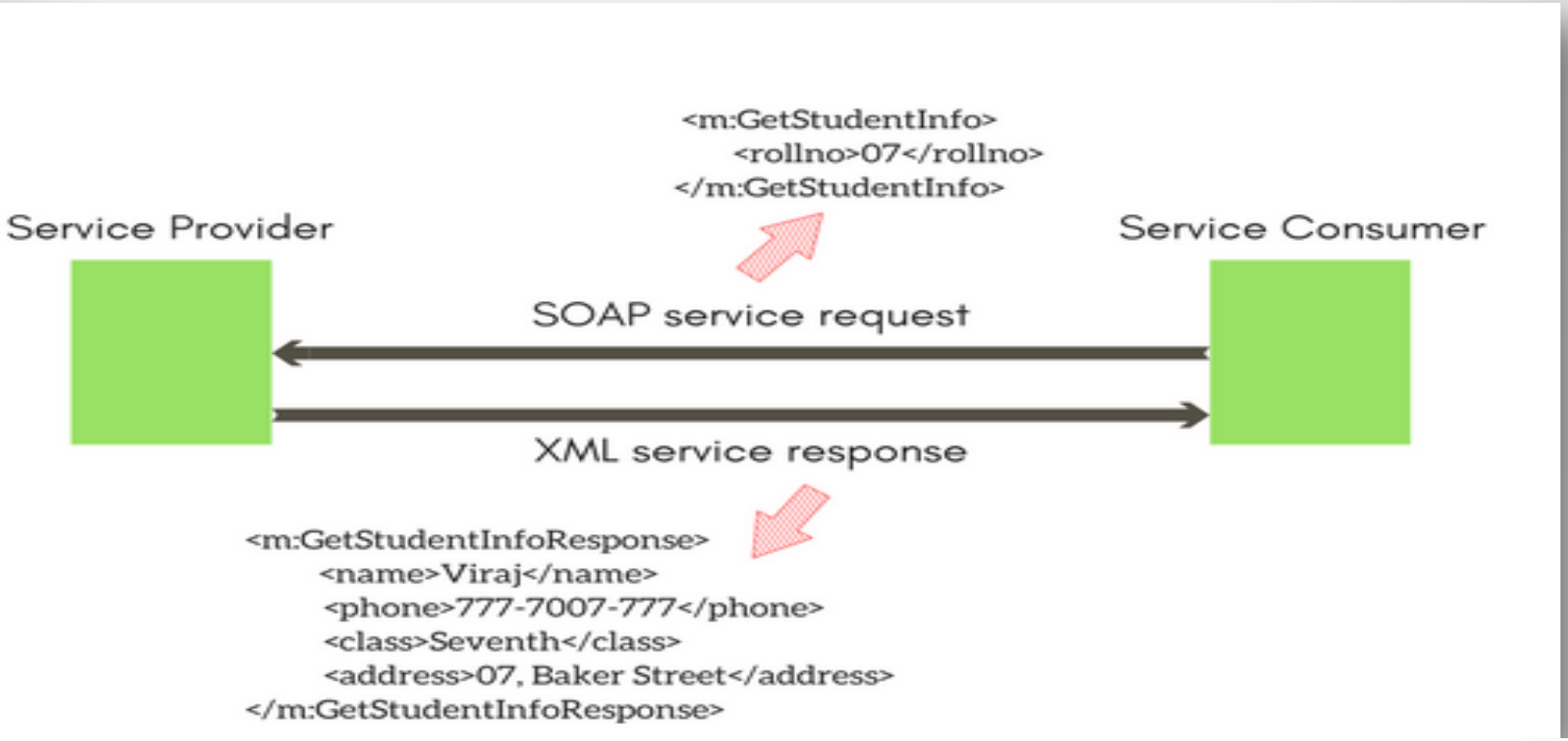
SOAP

- **SOAP Web Services**
- SOAP stands for Simple Object Access Protocol. It is a XML-based protocol for accessing web services.
- SOAP is a W3C recommendation for communication between two applications.
- SOAP is XML based protocol.
- It is platform independent and language independent.
- By using SOAP, you will be able to interact with other programming language

applications.

SOAP




<m:GetStudentInfo>
<rollno>07</rollno>
</m:GetStudentInfo>
Service Provider
Service Consumer
SOAP service request
XML service response
<m:GetStudentInfoResponse>
<name>Viraj</name>
<phone>777-7007-777</phone>
<class>Seventh</class>
<address>07, Baker Street</address>
</m:GetStudentInfoResponse>

SOAP

- Well, every application serving SOAP requests, has a **WSDL file**.
- WSDL is an XML, and it stands for Web Service Description Language.
- WSDL describes all the methods available in the web service, along with the request and response types.
- It describes the contract between service and client.
- SOAP was intended to be a way to do remote procedure calls to remote objects by sending XML over HTTP.
- Today the world is moving fast towards the RESTful Web Services.

SOAP

- **Advantages of Soap Web Services**
  - **WS Security**: SOAP defines its own security known as WS Security.
  - **Language and Platform independent**: SOAP web services can be written in any programming language and executed in any platform.
- **Disadvantages of Soap Web Services**
  - **Slow**: SOAP uses XML format that must be parsed to be read. It defines many standards that must be followed while developing the SOAP applications. So it is slow and consumes more bandwidth and resource.
  - **WSDL dependent**: SOAP uses WSDL and doesn't have any other mechanism to discover the service.

REST

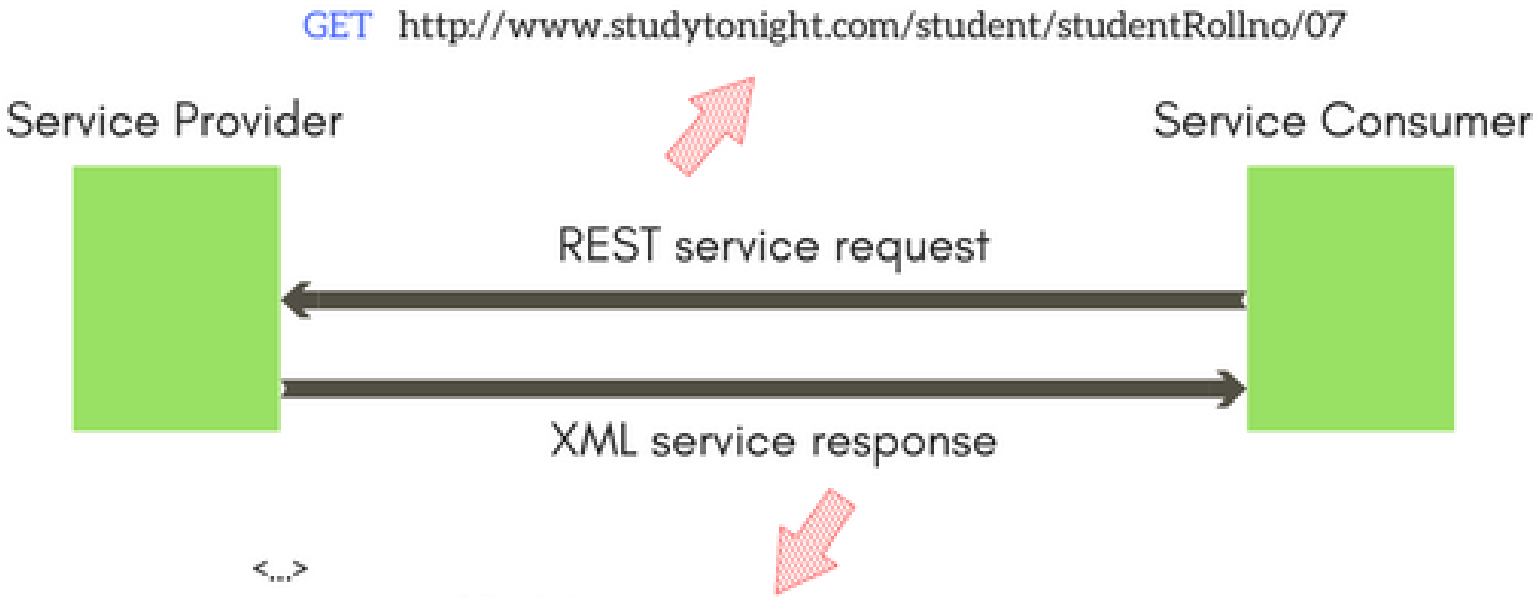
- **RESTful Web Services**
- REST stands for REpresentational State Transfer.
- REST is an architectural style not a protocol.
- **Advantages of RESTful Web Services**
  - **Fast**: RESTful Web Services are fast because there is no strict specification like SOAP. It consumes less bandwidth and resource.
  - **Language and Platform independent**: RESTful web services can be written in any programming language and executed in any platform.
  - **Can use SOAP**: RESTful web services can use SOAP web services as the implementation.
  - **Permits different data format**: RESTful web service permits different data format such as Plain Text, HTML,

XML and JSON.

REST

- The **REST** stands for **Representational State Transfer**.
- REST is not a set of standards or rules, rather it is a style of software architecture.
- The applications which follow this architecture are referred to as **RESTful**
- Unlike SOAP which targets the *actions*, REST concerns more on the resources.
- REST locates the resources by using URL and it depends on the type of **transport** protocol(with HTTP - GET, POST, PUT, DELETE,...) for the actions to be performed on the resources.
- The REST service locates the resource based on the URL and performs the action based on the transport action verb.

REST


GET http://www.studytonight.com/student/studentRollno/07
Service Provider
Service Consumer
REST service request
XML service response
<...>
<name>Viraj</name>
<phone>777-7007-777</phone>
<class>Seventh</class>
<address>07, Baker Street</address>
<...>

REST

- For Example: in a RESTful architecture, this URL *http://{serverAddress}/students/studentRollno/07* can be used to:
  - To get student information by sending a REST call of **GET** type, and the service will return information of student with roll no as 07
  - The same service can also be used to update the student data, by sending in the new values as Form data in a **PUT** request.

Outline- Web Services

- Web Services: Overview,
- Types of WS,
- Difference between SOAP and REST

SOAP v/s REST Web Services

Outline

- Web Services
- EJB
- CMS
- Advance Technology

EJB- Enterprise JavaBeans

Outline of EJB

- EJB: Overview
- types of EJB,
- benefits,
- Architecture,
- EJB technology,
- JNDI lookup

EJB: Overview

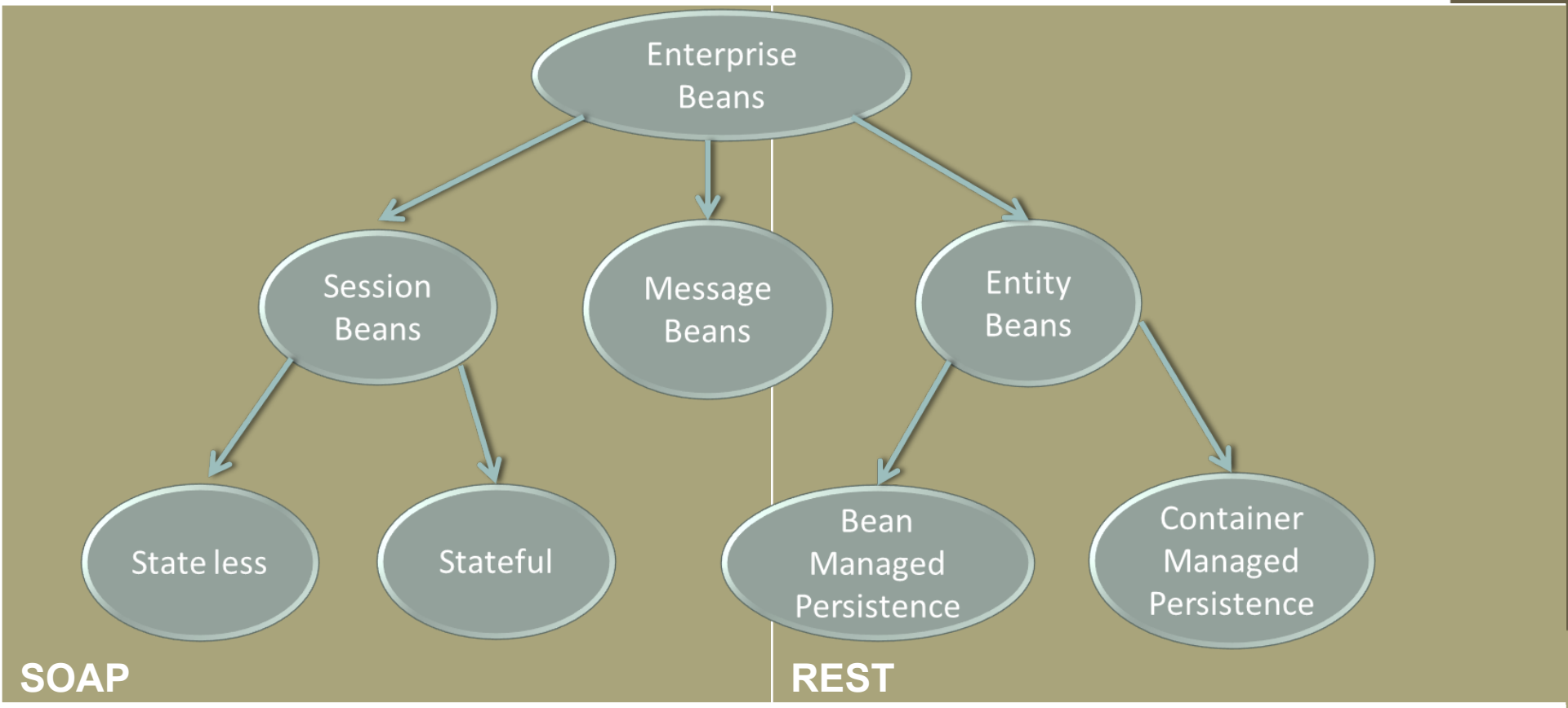
- EJB stands for **Enterprise Java Beans**.
- EJB is an essential part of a J2EE platform.
- EJB provides an architecture to develop and deploy component based enterprise applications considering robustness, high scalability, and high performance.
- **When use Enterprise Java Bean?**
  - **Application needs Remote Access**. In other words, it is distributed.
  - **Application needs to be scalable**. EJB applications supports load balancing, clustering and fail-over.

- **Application needs encapsulated business logic**. EJB application is separated from presentation and persistent layer.

Outline of EJB

- EJB: Overview
- types of EJB,
- benefits,
- Architecture,
- EJB technology,
- JNDI lookup

Types of EJB



SOAP	REST
SOAP is a **protocol**.	REST is an **architectural style**.

SOAP stands for **Simple Object Access Protocol**.	REST stands for **REpresentational State Transfer**.
SOAP **can't use REST** because it is a protocol.	REST **can use SOAP** web services because it is a concept and can use any protocol like HTTP, SOAP.
SOAP **uses services interfaces to expose the business logic**.	REST **uses URI to expose business logic**.
JAX-WS is the java API for SOAP web services.	**JAX-RS** is the java API for RESTful web services.
SOAP **defines standards** to be strictly followed.	REST does not define too much standards like SOAP.
SOAP **requires more bandwidth** and resource than REST.	REST **requires less bandwidth** and resource than SOAP.
SOAP **defines its own security**.	RESTful web services **inherits security measures** from the underlying transport.
SOAP **permits XML** data format only.	REST **permits different** data format such as Plain text, HTML, XML, JSON etc.
SOAP is **less preferred** than REST.	REST **more preferred** than SOAP.

Types of EJB

S.No	Type & Description
1	**Session Bean** Session bean stores data of a particular user for a single session. It can be **stateful** or **stateless**. It is less resource intensive as compared to entity bean. Session bean gets destroyed as soon as user session terminates.
2	**Entity Bean** **Entity beans** represent persistent data storage. User data can be saved to database via entity beans and later on can be retrieved from the database in the entity bean.

3	**Message Driven Bean** Message driven beans are used in context of JMS (Java Messaging Service). Message Driven Beans can consumes JMS messages from external entities and act accordingly.

Session beans

- **Stateless Session bean** *is a business object that represents business logic* only. It doesn't have state (data).
- In other words, *conversational state* between multiple method calls is not maintained by the container in case of stateless session bean.
- The stateless bean objects are pooled by the EJB container to service the request on demand.
- It can be accessed by one client at a time. In case of concurrent access, EJB container routes each request to different instance.
- **Annotations used in Stateless Session Bean**

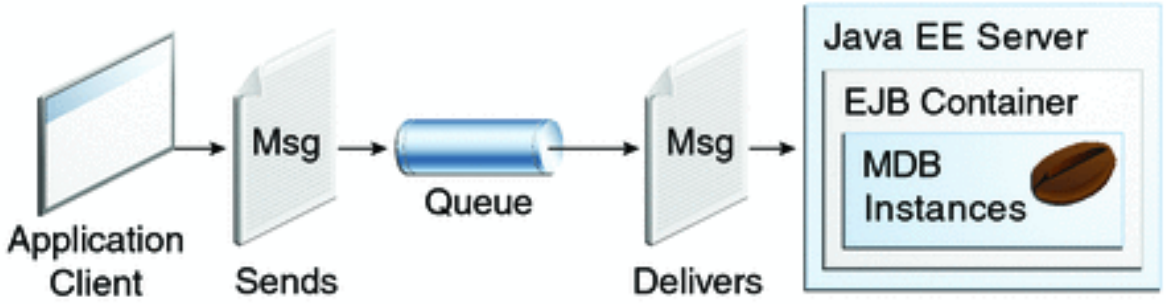
- @Stateless

Session beans

- **Stateful Session Bean**
- **Stateful Session bean** *is a business object that represents business logic* like stateless session bean. But, it maintains state (data).
- In other words, *conversational state* between multiple method calls is maintained by the container in stateful session bean.
- **Annotations used in Stateful Session Bean**
- @Stateful

Message Driven Bean

- **Message Driven Bean**
- A message driven bean (MDB) is a bean that contains business logic. But, it is invoked by passing the message. So, it is like JMS Receiver.
- MDB asynchronously receives the message and processes it.
- A message driven bean receives message from queue or topic, so you must have the knowledge of JMS API.
- A message driven bean is like stateless session bean that encapsulates the business logic and doesn't maintain state.



Entity beans

- **Container-managed persistence—**
  - The EJB container manages data by saving it to a designated resource, which is normally a database.
  - For this to occur, you must define the data that the container is to manage within the deployment descriptors.
  - The container manages the data by saving it to the database.

- **Bean-managed persistence—**
  - The bean implementation manages the data within callback methods.
  - All the logic for storing data to your persistent storage must be included in the ejbStore method and reloaded from your storage in the ejbLoad method.

- The container invokes these methods when necessary.

Outline of EJB

- EJB: Overview
- types of EJB,
- benefits,
- Architecture,
- EJB technology,
- JNDI lookup

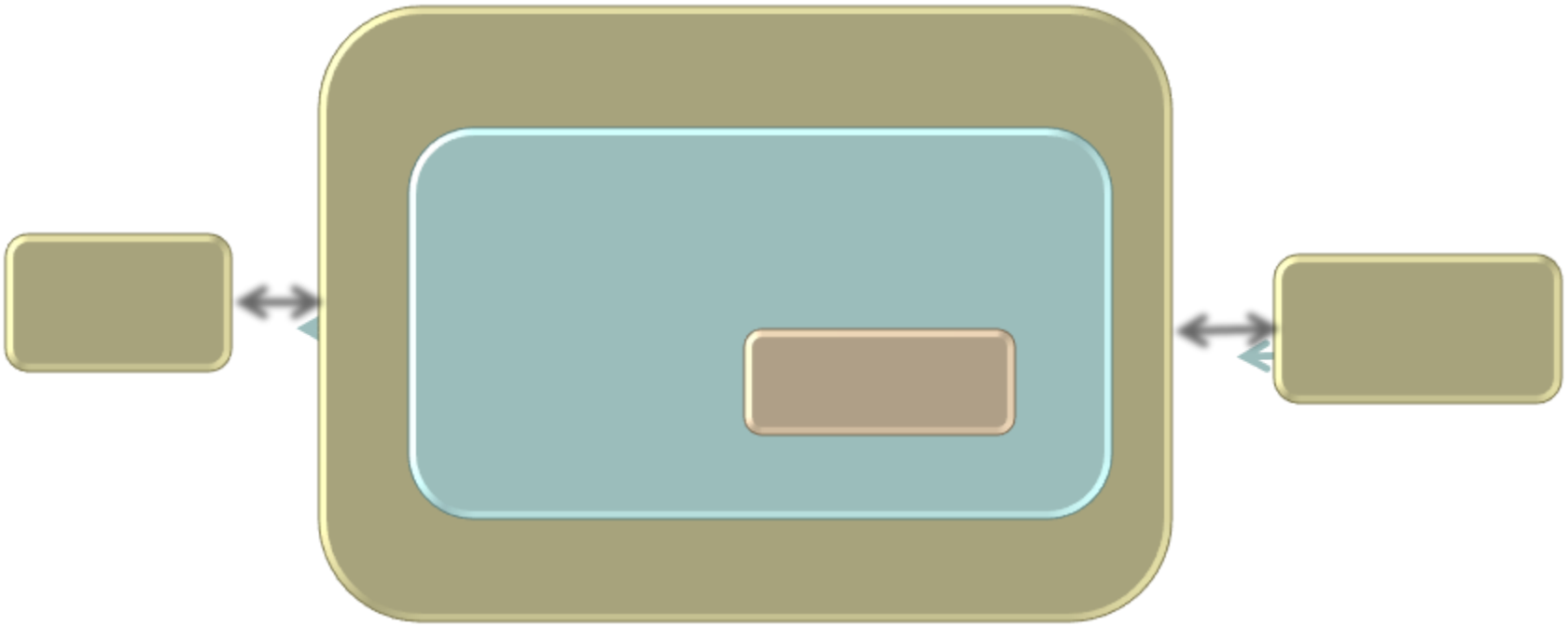
EJB- Benefits

- Simplified development **of large-scale enterprise level application.**
- Application Server/EJB container provides most of the system level services like **transaction handling, logging, load balancing**, persistence mechanism, exception handling, and so on.
- EJB developer just focus on **business logic** and on solving business problems.
- Because business logic lies in EJB, so Front end developer can focus on the presentation of client interface.
- The **client developer does not have to code** the routines that implement **business rules or access databases**. As a result, clients side has less codes which is particularly important for clients that **run on small devices**.
- Java Beans are **portable** components which enable the java application assembler to build new applications from existing java beans.

Outline of EJB

- EJB: Overview
- types of EJB,
- benefits,
- Architecture,
- EJB technology,
- JNDI lookup

EJB Architecture



EJB Architecture- Components

- EJB server(s)
- EJB container(s)
- Enterprise Beans
- EJB clients
- other auxillary systems (e.g. Java Naming and Directory Interface (JNDI) server, Java Transaction Service (JTS), ...).

EJB Architecture- Components

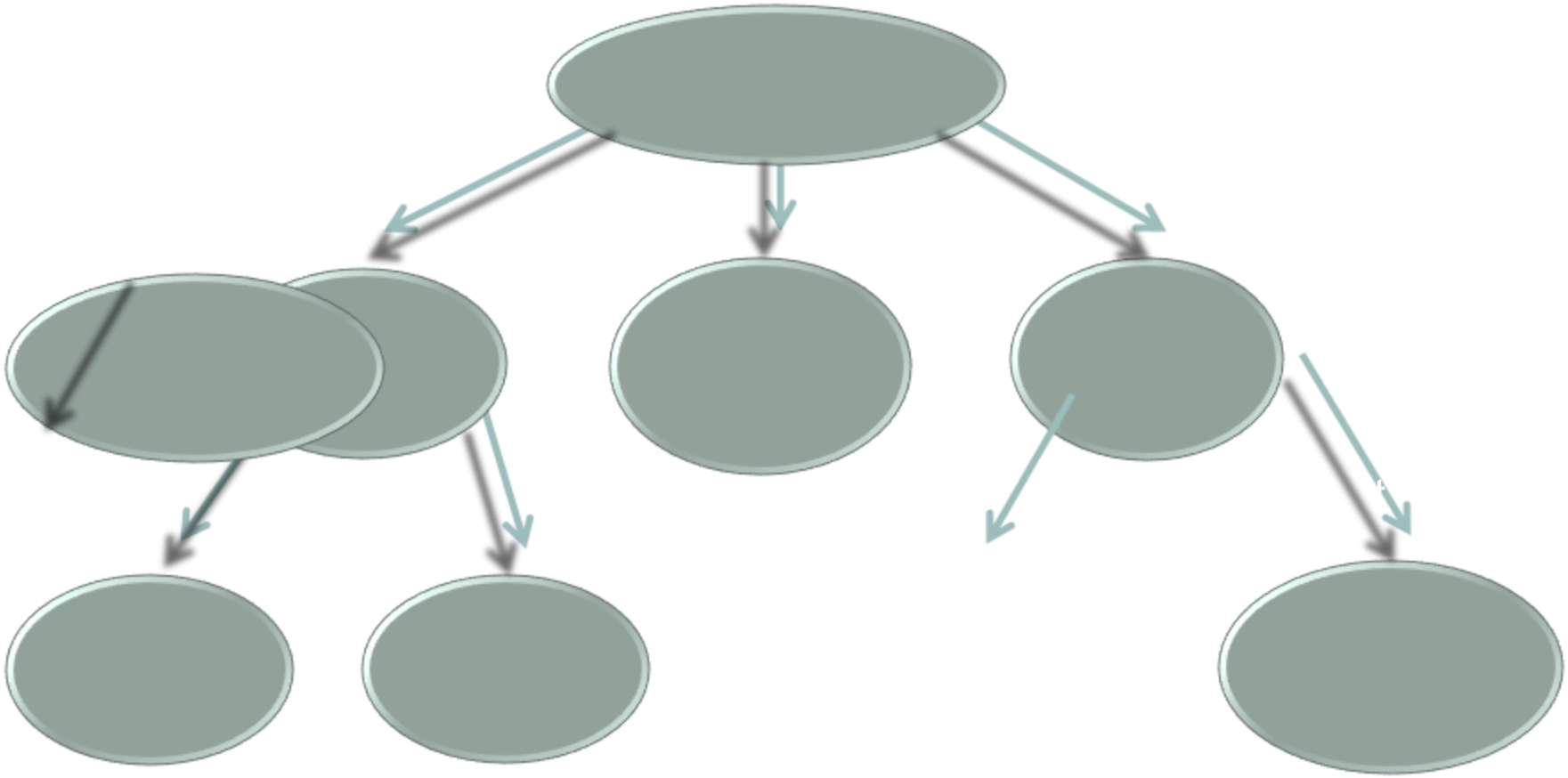
- **EJB Server:**
  - Outmost container
  - Manage EJB Container
  - Provides runtime environment
  - Services
    1. Process & thread management
    2. System recourse management
    3. Database connection pooling & catching

EJB Architecture -Components

- **EJB Container**
  - Contains Enterprise beans
  - Services
  - Naming
  - Life style management
  - Persistent
  - Transaction management
  - Security

EJB Architecture-Components

- **Enterprise Java Beans**



EJB Architecture -Components

- **EJB Clients**
  - Interacts with EJB container
  - EJB Client is a local program which can call & operate remote beans.
  - Client locates an Enterprise Java Beans through JNDI(Java Naming Directory Services).

Outline of EJB

- EJB: Overview
- types of EJB,
- benefits,
- Architecture,
- EJB technology,
- JNDI lookup

EJB technology

- Enterprise JavaBeans (EJB) technology is the server-side component architecture for Java Platform, Enterprise Edition (Java EE).
- EJB technology enables rapid and simplified development of distributed, transactional, secure and portable applications based on Java technology.

Outline of EJB

- EJB: Overview
- types of EJB,
- benefits,
- Architecture,
- EJB technology,
- JNDI lookup

JNDI lookup

- The Java Naming and Directory Interface (JNDI) is a Java API for a directory service that allows Java software clients to discover and look up data and objects via a name.
- Like all Java APIs that interface with host systems, JNDI is independent of the underlying implementation.
- Additionally, it specifies a service provider interface (SPI) that allows directory service implementations to be plugged into the framework.
- It may make use of a server, a flat file, or a database; the choice is up to the vendor.
- **Typical uses of JNDI include:**
  - Connecting a Java application to an external directory service (such as an address database or an LDAP server)
  - Allowing a Java Servlet to look up configuration information provided by the hosting web container

JNDI lookup-Architecture

- The JNDI architecture consists of an API and a service provider interface (SPI).
- Java applications use the JNDI API to access a variety of

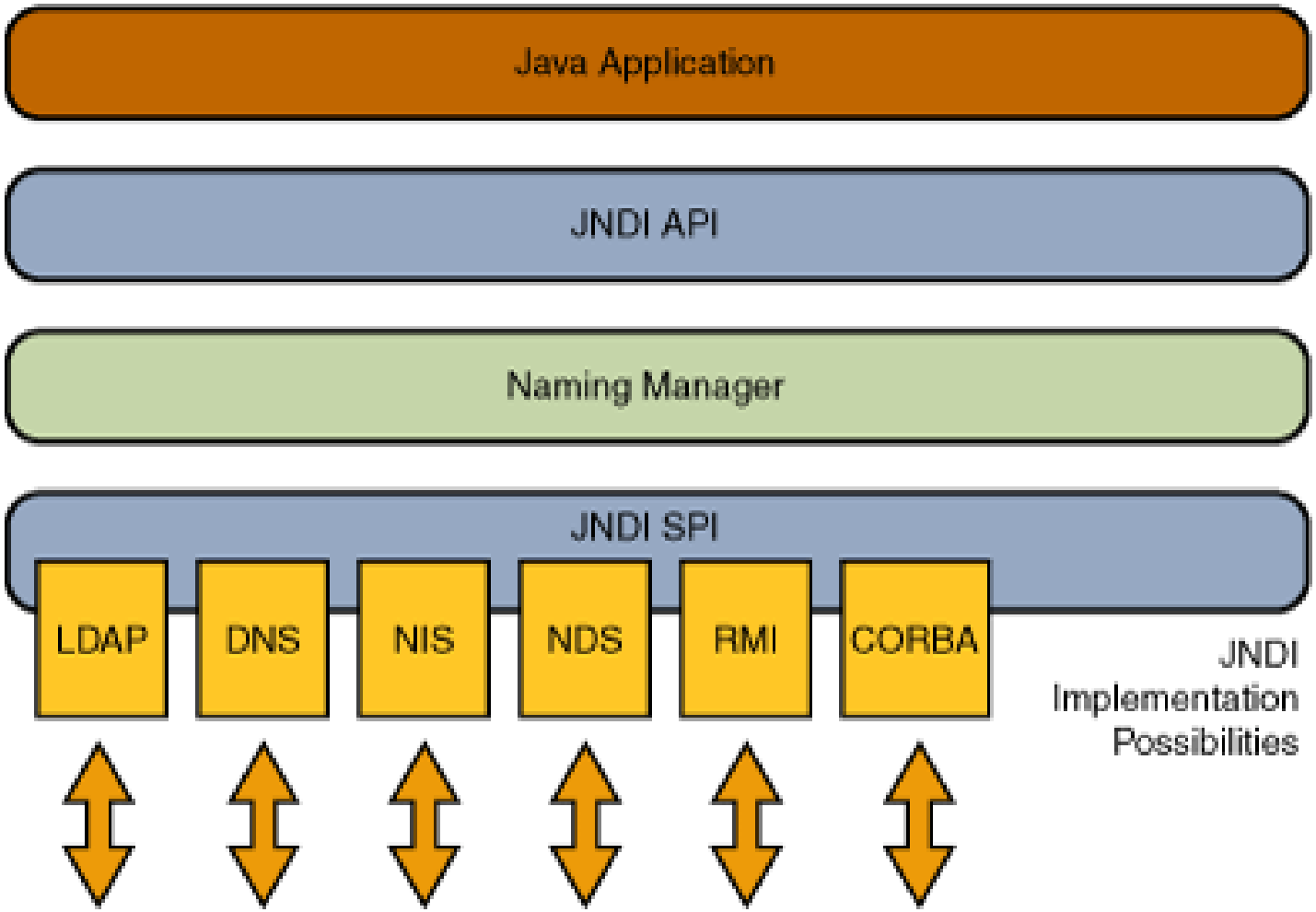
  naming and directory services.

- The SPI enables a variety of naming and directory services to be plugged in transparently, thereby allowing the Java application using the JNDI API to access their services
- To use the JNDI, you must have the JNDI classes and one or more service providers. The JDK includes service providers for the following naming/directory services:
  - Lightweight Directory Access Protocol (LDAP)
  - Common Object Request Broker Architecture (CORBA)

Common Object Services (COS) name service

- Java Remote Method Invocation (RMI) Registry
- Domain Name Service (DNS)

JNDI lookup-Architecture



Outline

- Web Services
- EJB
- CMS
- Advance Technology

CMS-

Content Management System

Outline of CMS

- Introduction to Content Management System(CMS)
- Wordpress
- Joomala

Introduction to Content Management System(CMS)

- A content management system (CMS) is a computer application that supports the creation and modification of digital content.
- It typically supports multiple users in a collaborative environment.

- CMS features vary widely. Most CMSs include Web-based publishing, format management, history editing and version control, indexing, search, and retrieval.

- CMS is designed to support the management of the content of Web pages. Web content includes text and embedded graphics, photos, video, audio, maps, and program code (e.g., for applications) that displays content or interacts with the user.

Introduction to Content Management System(CMS)

- Such a content management system (CMS) typically has two major components:

1. **A content management application (CMA)** is the front-end user interface that allows a user, to add, modify, and remove content from a website.

2. **A content delivery application (CDA)** compiles that information and updates the website.

Introduction to Content Management System(CMS)

- Examples of CMS
  - **WordPress,**
  - **Joomla**
  - **Drupal.**

Introduction to Content Management System(CMS)

- **Advantages**
  - Reduced need to code from scratch
  - Easy to create a unified look and feel
  - Version control
  - Edit permission management

- **Disadvantages**
  - Limited or no ability to create functionality not proposed in the CMS (e.g., layouts, web apps, etc.)
  - Increased need for special expertise and training for content authors

Outline of CMS

- Introduction to Content Management System(CMS)
- Wordpress
- Joomala

Word press

- WordPress is an online, open source website creation tool written in PHP.
- But in non-geek speak, it’s probably the easiest and most powerful blogging and website content management system (or CMS) in existence today.

Word press- Features

- **User Management** − It allows managing the user information such as changing the role of the users, create or delete the user, change the password and user information.
- **Media Management** − It is the tool for managing the media files and folder, in which you can easily upload, organize and manage the media files on your website.
- **Theme System** − It allows modifying the site view and functionality. It includes images, stylesheet, template files and custom pages.
- **Extend with Plugins** − Several plugins are available which provides custom functions and features according to the users need.
- **Search Engine Optimization** − It provides several search engine optimization (SEO) tools which makes on-site SEO simple.
- **Multilingual** − It allows translating the entire content

into the language preferred by the user.

- **Importers** − It allows importing data in the form of posts. It imports custom files, comments, post pages and tags.

Word press-Advantages

- Open source platform and available for free.
- CSS files can be as per users need.
- There are many plugins and templates available for free.
- It is very easy to edit the content as it uses that allows the user to directly manipulate the layout of document
- Media files can be uploaded easily and quickly.
- It offers several SEO tools which makes on-site SEO simple.
- Customization is easy according to the user's needs.
- It allows creating different roles for users for website such as admin, author, editor and contributor.

Word press-disadvantages

- Using several plugins can make the website heavy to load & run.
- PHP knowledge is required to make modifications or changes in the WordPress website.
- Sometimes software needs to be updated to keep the WordPress up-to-date with the current browsers and mobile devices.
- Updating WordPress version leads to loss of data, so a backup copy of the website is required.
- Modifying and formatting the graphic images and tables is difficult.

WordPress - Installation

- **System Requirements for WordPress**
- **Database** − MySQL 5.0 +
- **Web Server** −
  - WAMP (Windows)
  - LAMP (Linux)
  - XAMP (Multi-platform)
  - MAMP (Macintosh)
- **Operating System** − Cross-platform
- **Browser Support** − IE (Internet Explorer 8+), Firefox, Google chrome, Safari, Opera
- **PHP Compatibility** − PHP 5.2+
- **Download WordPress using** link

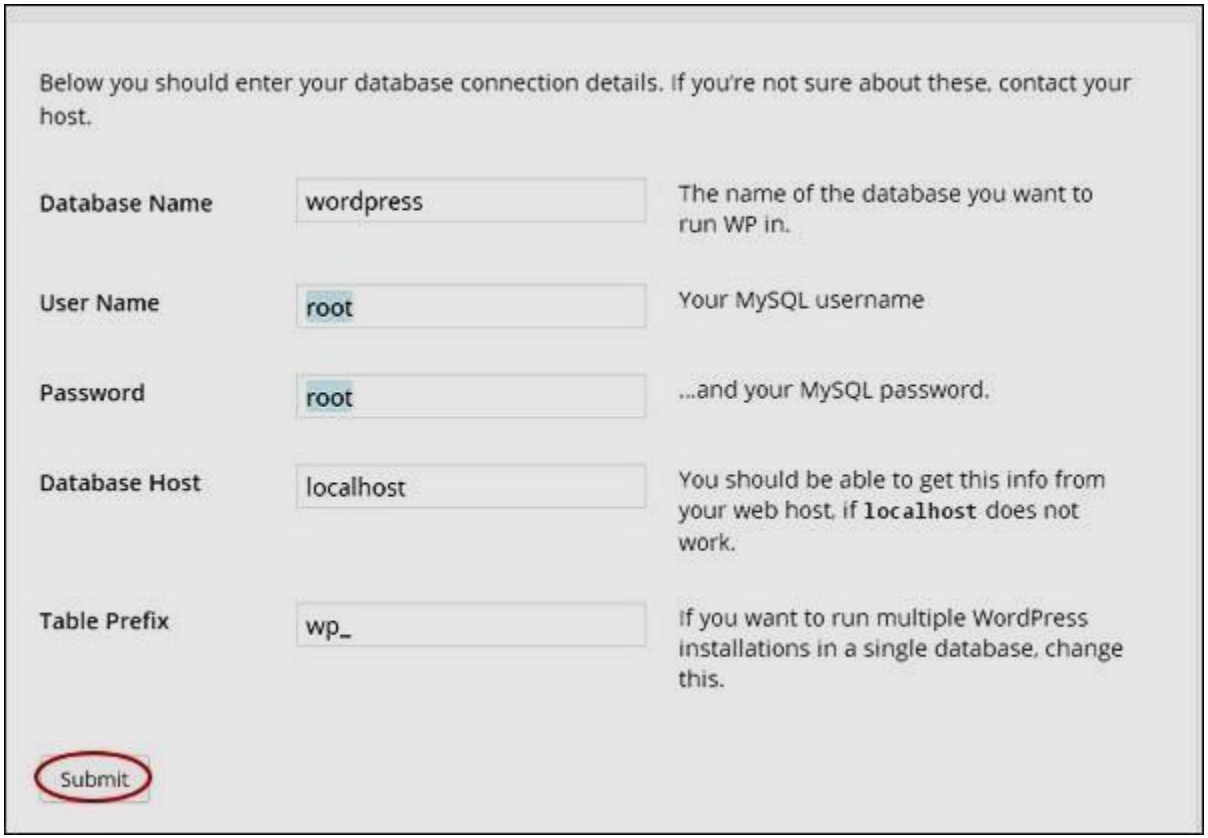
https://wordpress.org/download/

WordPress - Installation

- **Create Store Database**
- WordPress requires MySQL database. So create a new empty database with user/password
- **Set Up Wizard**
  - **Step (1)** − Extract the downloaded WordPress folder and upload it into your web server or localhost.
  - **Step (2)** − Open your browser and navigate to your WordPress file path, then you will get the first screen of the WordPress installer. Select your language for the WordPress and click on **Continue**.
  - **Step (3)** − In this step, you can view the information needed for the database before proceeding with WordPress installation. Click on **Let's go!**

WordPress - Installation

Step (4) − Here, you have to enter the information about the MySQL database as described in the following screen.

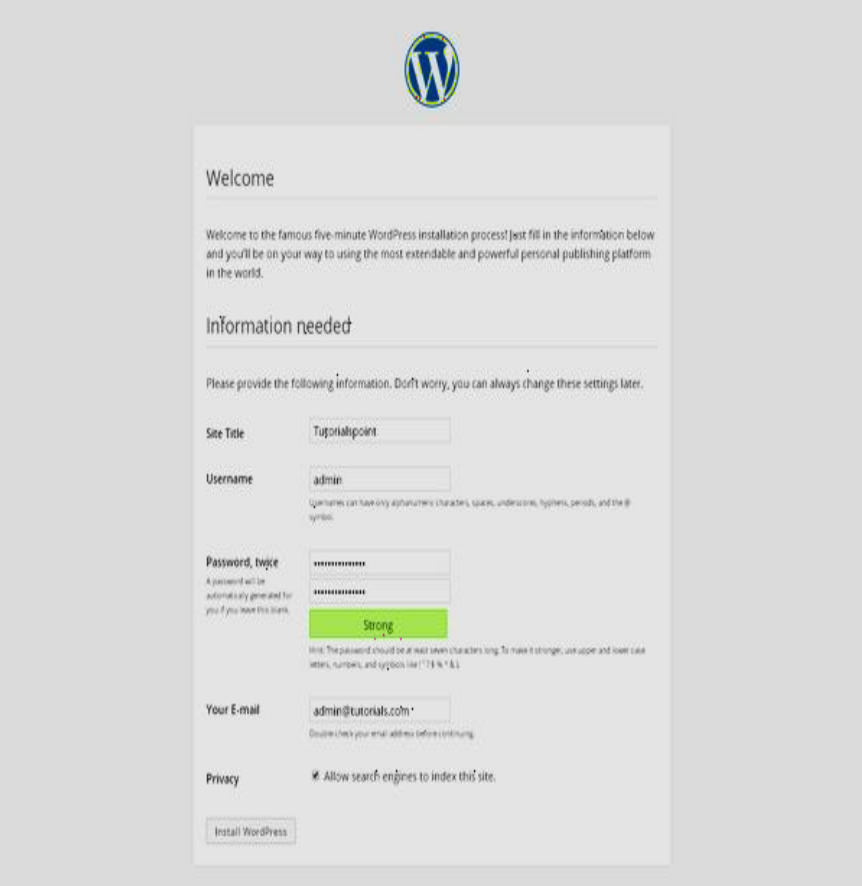


WordPress - Installation

Step (5) − WordPress checks the database setting and gives you the confirmation screen as shown in the following snapshot. Click on **Run the install**

WordPress - Installation

Step (6) − Enter administrative information.



It contains the following fields −

Site Title − Enter the name of the site which you are going to create in WordPress.

Username − Enter the username as per your choice while logging in the WordPress.

Password twice − Enter password two times to protect your site.

Your E-mail − Enter your e-mail address which helps to recover the password or any update.

Privacy − It allows the search engine to index this site after checking the checkbox.

After filling all the information, click on

*the **Install WordPress** button.*

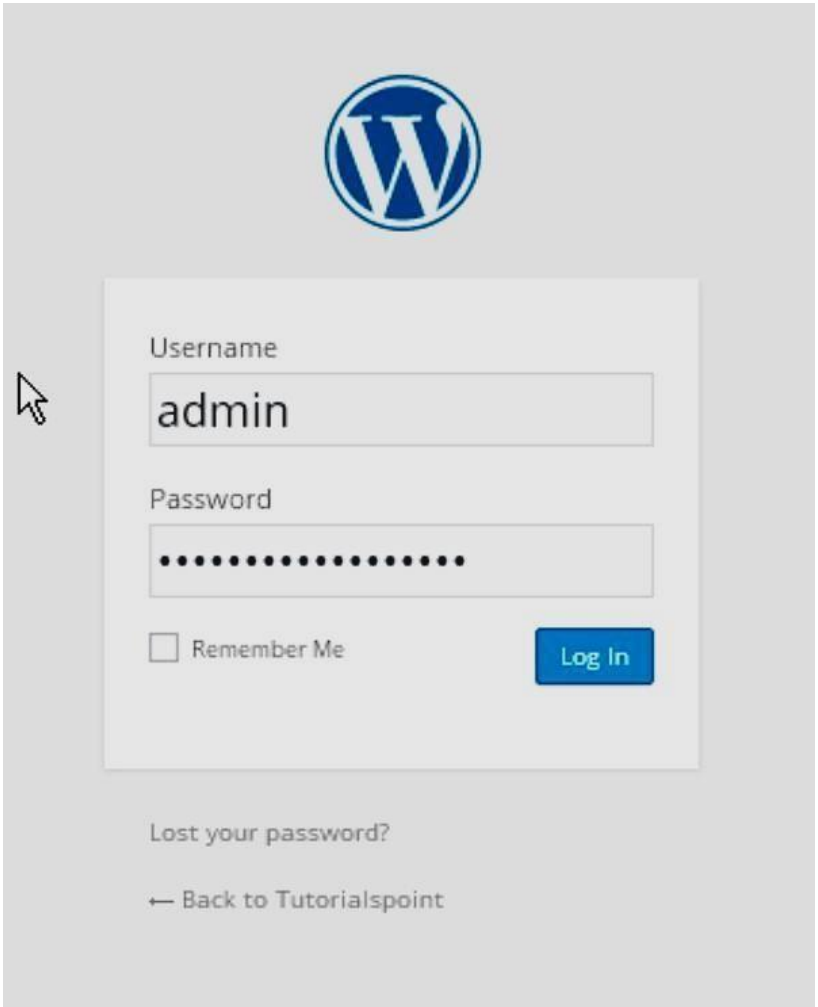
WordPress - Installation

Step (7) − After installation being successful, you will get a screen of the stating success as seen in the following screen.

You can view your username and password detail added in WordPress. Click on **Log In** button.

WordPress - Installation

Step (8) − After clicking on login, you will get a WordPress Admin Panel as depicted in the following screen.

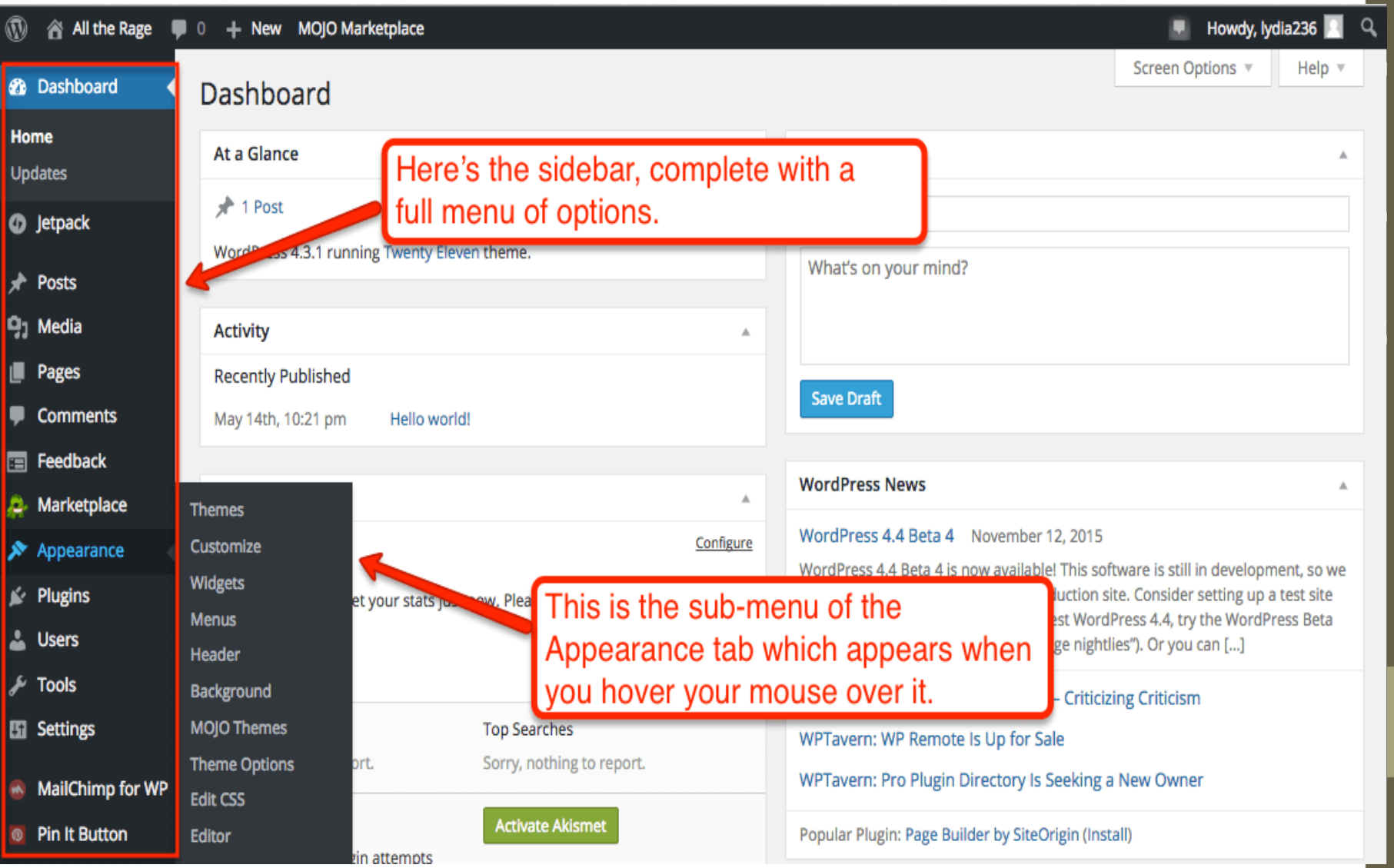


Enter the username and password which you had mentioned during installation as shown in step 6 and click on the **Log In** button.

WordPress - Dashboard

- The WordPress Dashboard is a first screen which will be seen when you log into the administration area of your blog which will display the overview of the website.
- It is a collection of gadgets that provide information and provide an overview of what's happening with your blog.
- You can customize your needs by using some quick links such as writing quick draft, replying to latest comment, etc.

WordPress - Dashboard



Outline of CMS

- Introduction to Content Management System(CMS)
- Wordpress
- Joomala

Joomala

- Joomla is an open source Content Management System (CMS), which is used to build websites and online applications.
- It is free and extendable which is separated into front-end templates and back-end templates (administrator).
- Joomla is developed using PHP, Object Oriented Programming, software design patterns and MySQL (used for storing the data).
- **History**
- Joomla is based on **Mambo** CMS which was developed by an Australian company in 2001 and initially released on *August 17, 2005*. The official version of Joomla 1.0 was released on *September 22, 2005*.

Joomala-Features

- **User Manager** − It allows managing the user information such as permission to edit, access, publish, create or delete the user, change the password and languages.
- **Content Manager** − It allows managing the content using WYSIWYG editor to create or edit the content in a very simple way.
- **Banner Manager** − It is used to add or edit the banners on the website.
- **Template Manager** − It manages the designs that are used on the website. The templates can be implemented without changing the content structure within a few seconds.
- **Media Manager** − It is the tool for managing the media files and folder in which you can easily upload, organize and manage your media files into your article editor tool.
- **Contact Manager** − It allows to add contacts, managing the contact information of the particular users.
- **Web Link Manager** − The link resource is provided for user of the site and can be sorted into categories.
- **Search** − It allows users to search the appropriate information on the site. You can use smart indexing, advanced search options, auto suggest searches
- **Menu Manager** − It allows to create menus and menu items.
- **RSS** − It stands for Really Simple syndication which helps your site contents and RSS files to be automatically updated.

Joomala- Advantages

- It is an open source platform and available for free.
- Joomla is designed to be easy to install and set up even if you're not an advanced user.
- Since Joomla is so easy to use, as a web designer or developer, you can quickly build sites for your clients.
- It is very easy to edit the content as it uses WYSIWYG editor (**W**hat **Y**ou **S**ee **I**s **W**hat **Y**ou **G**et is a user interface that allows the user to directly manipulate the layout of the document without having a layout command).
- It ensures the safety of data content and doesn't allow anyone to edit the data.
- By default, Joomla is compatible with all browsers.
- The templates are very flexible to use.
- Media files can be uploaded easily in the article editor tool.

- Provides easy menu creation tool.

Joomala- Disadvantages

- It gives compatibility problem while installing several modules, extensions and plugins simultaneously.
- Plugins and modules are not free in Joomla.
- Development is too difficult to handle when you want to change the layout.
- Joomla is not much SEO (Search Engine Optimization) friendly.
- It makes website heavy to load and run.

Real World Examples of What Joomla Can Create?

- Corporate web sites or portals
- Corporate intranets and extranets
- Online magazines, newspapers, and publications
- E-commerce and online reservations
- Government applications
- Small business web sites
- Non-profit and organizational web sites
- Community-based portals
- School and religious web sites
- Personal or family homepages

Joomla - Installation

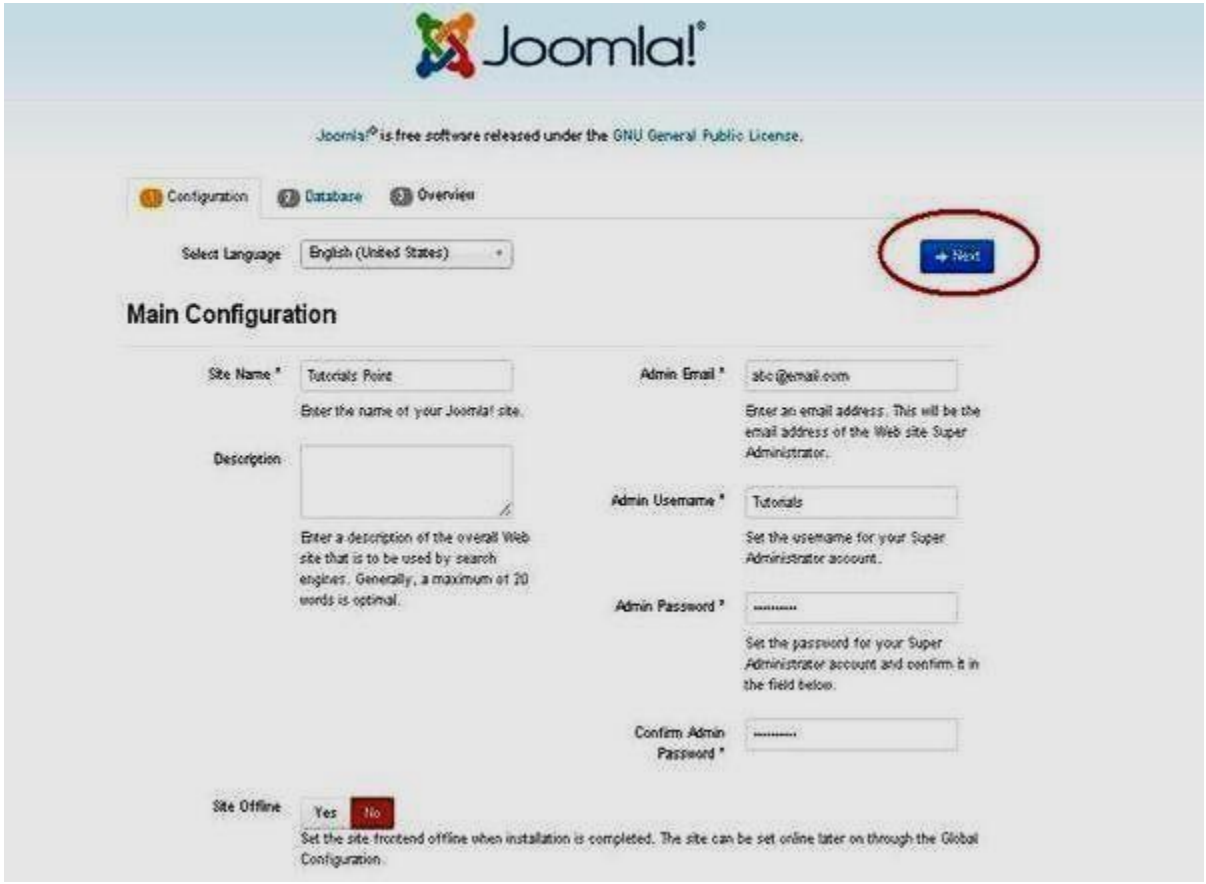
- **System Requirements for Joomla 3.x**
- **Database** − MySQL 5.1 +
- **Web Server** −
  - WAMP (Windows)
  - LAMP (Linux)
  - XAMP (Multi-platform)
  - MAMP (Macintosh)
  - Nginx
  - Microsoft IIS

- **Operating System** − Cross-platform
- **Browser Support** − IE (Internet Explorer 7), Firefox, Google chrome
- **SSL (Secure Socket Layer)** − A valid security certificate is required for HTTPS
- **PHP Compatibility** − PHP 5.4+ or PHP 5.3.10+
- **Download Joomla using** link http://www.joomla.org/download.htm

Joomla - Installation

- **Create Store Database**
  - Joomla requires MySQL database. So create a new empty database and user/password for Joomla.
  - After following the above step, you can continue with the installation process.
- **SetUp Wizard**
  - **Step (1)** − Extract the downloaded Joomla folder and upload it on your web server or localhost.
  - **Step (2)** − Open your browser and navigate to your Joomla file path, then you will get the first screen of the Joomla installer as shown in the following screen. In our case the path is **localhost/< Your_joomla_folder >**.

Joomla - Installation



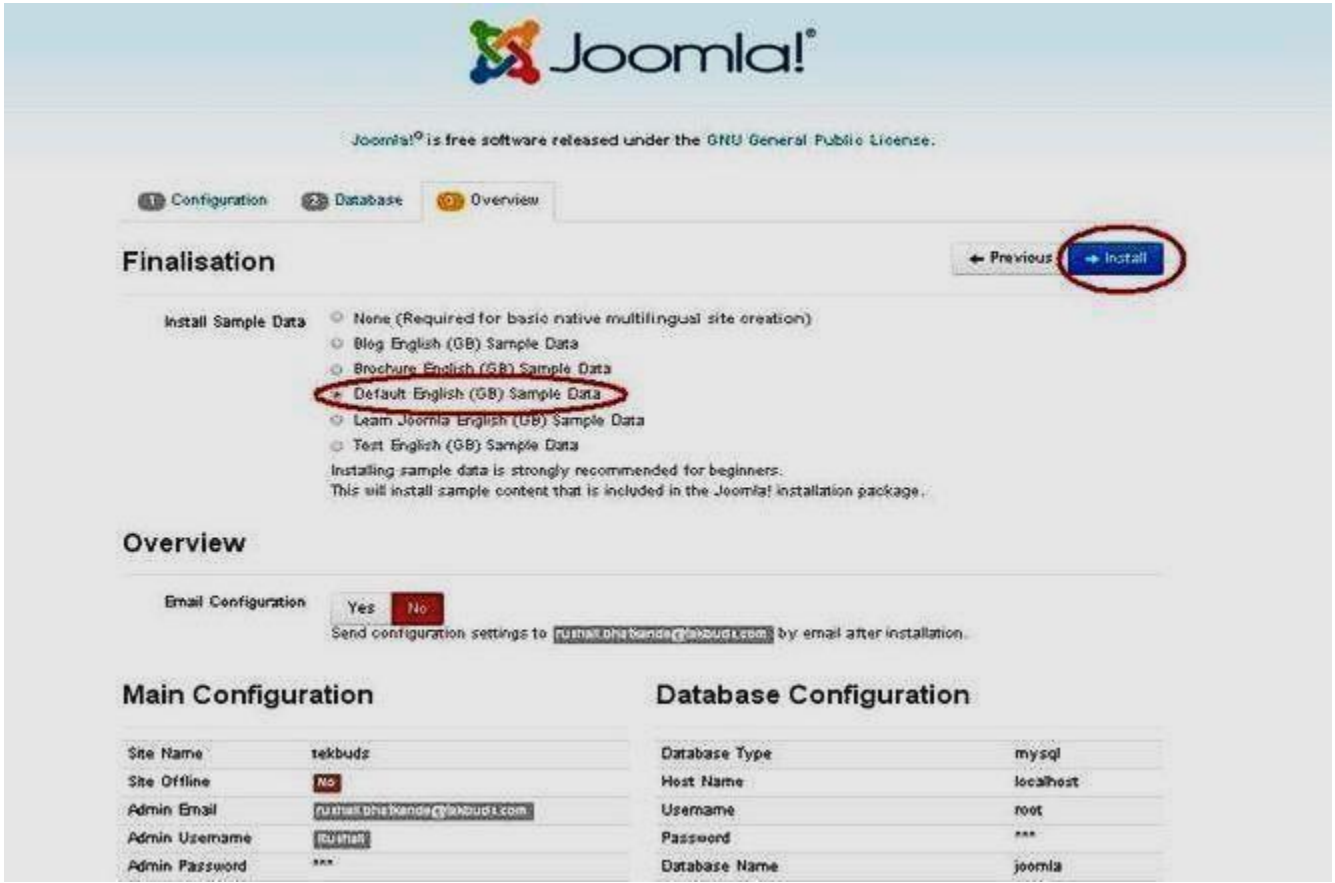
Joomla - Installation

Step (3) − Here, you have to enter the information about the MYSQL database as seen in the screen below.

Joomla - Installation

Step (4) − In this step, you will be on last page of the installation process.

Choose **Default English (GB) Sample Data** as an example to build your website and click on **Install** button.



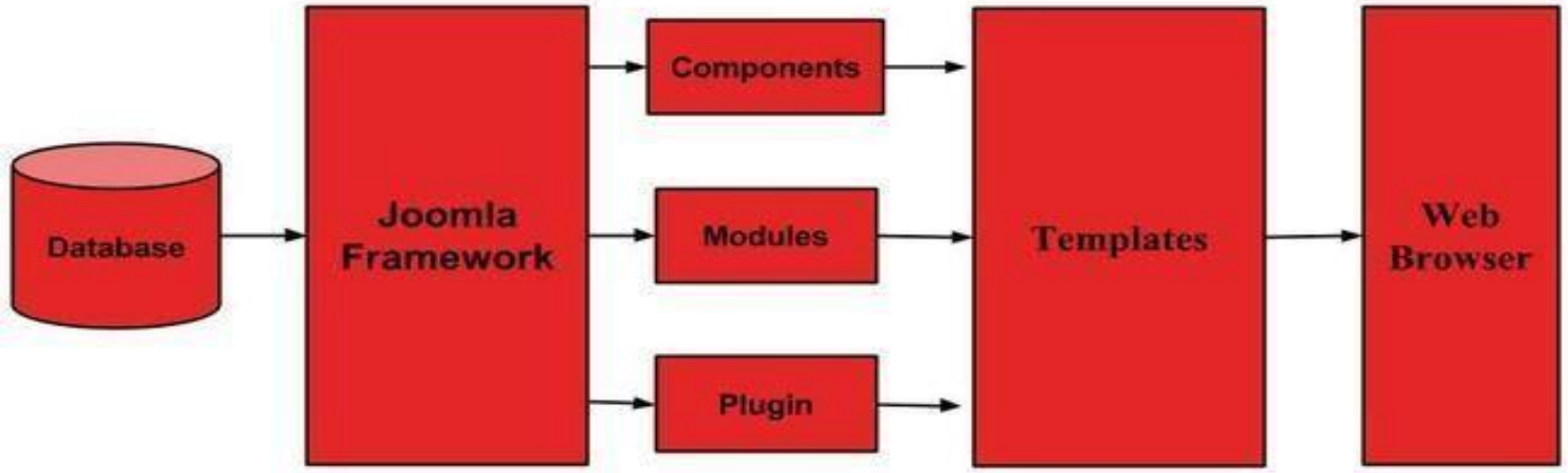
Joomla - Installation

- **Step (5)** − After a few minutes, after installation is successful and you will get a screen of the Joomla web installer as shown below.
- **Admin Panel**
- After installing Joomla you are able to login to your Joomla Admin Panel as described below −
- **Step (1)** − To access Joomla administrative panel, open the browser and type the URL as

http://localhost/ < Your_joomla_folder> /administrator/ index.php

- **Step (2)** − Enter the username and password which you had mentioned during installation as shown in the step 3 of Set Up Wizard section and click on the login button.

Joomla - Architecture



Joomla - Architecture

- Joomla is a Model-View-Controller web application.
- The architecture of Joomla contains following layers −
  - Database
  - Joomla Framework
  - Components
  - Modules
  - Plugin
  - Templates
  - Web Server

Joomla - Control Panel

- The **Control Panel** provides default features and functions of Joomla to access through clickable icons, menu bar etc. Important icons of control panel and their functions are :
- **Article Content**
  - **Add New Article** − It creates a new article page.
  - **Article Manager** − It manages all your present articles.
  - **Category Manager** − It creates new categories and helps in publishing/unpublishing the categories.
  - **Media Manager** − It manages the files by uploading various new files or deleting the existing ones on your web server.
- **Structure Format**
  - **Menu Manager** − Menu manager allows creating custom menus for your website and navigating through your website.
  - **Module Manager** − It manages the modules such as location and function of modules that are installed on site.
- **User Information**

- **User Manager** − It manages the user information, which allows creating or deleting the user, changing passwords, time and languages. You can also assign the user to *User Groups*.

Joomla - Control Panel

- **Set the configuration**
  - **Global Configuration** − This is an important part in the Joomla back- end. Any changes made in this configuration, will affect the entire website.
  - **Template Manager** − It manages the templates used in the website.
  - **Language Manager** − It manages installed language by setting the default language for your site.
- **Install Extension**
  - There are many Extensions available in Joomla. You can install different types of extensions to extend the functionality of the site.
- **Maintenance**
  - **Joomla is up-to-date** − It views the current update status of the Joomla installation.
  - **All extensions are up-to-date** − It views the current update

status of the Joomla extension.

Difference between Wordpress & Joomla

Parameter	Wordpress	Joomla
Flexible	Less Flexible	More flexible
Interface	People like Wordpress interface more	People less like Joomla interface
SEO	SEO capability of Wordpress is not as good as Joomla	SEO capability of Joomla is more better than Wordpress
Security	Thousands of plugins are available for Wordpress which may raise issue of security	Joomla has its own set of security extensions, hence security is good.
Content Management	Easy	Little bit difficult

Outline

- Web Services
- EJB
- CMS
- Advance Technology

Advance Technology

Outline of Advance Technology

- Bootstrap
- JSF
- Spring

Bootstrap

- Twitter Bootstrap is the most popular front end framework in the recent time.
- It is sleek, intuitive, and powerful mobile first front-end framework

  for faster and easier web development.

- It uses HTML, CSS and Javascript.
- Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins
- Bootstrap also gives you the ability to easily create responsive designs
- **What is Responsive Web Design?**

Responsive web design is about creating web sites which automatically adjust themselves to look good on all devices, from small phones to large desktops.

Bootstrap

- **Bootstrap History**
  - Bootstrap was developed by Mark Otto and Jacob Thornton at Twitter, and released as an open source product in August 2011 on GitHub.
  - **In June 2014 Bootstrap was the No.1 project on GitHub!**
- **Why Use Bootstrap?-**Advantages of Bootstrap:
  - **Easy to use:** Anybody with just basic knowledge of HTML and CSS can start using Bootstrap
  - **Responsive features:** Bootstrap's responsive CSS adjusts to phones, tablets, and desktops
  - **Mobile-first approach:** In Bootstrap 3, mobile-first styles are part of the core framework
  - **Browser compatibility:** Bootstrap is compatible with all modern browsers (Chrome, Firefox, Internet Explorer, Safari, and Opera)
- **Where to Get Bootstrap?**
  - Download Bootstrap from getbootstrap.com

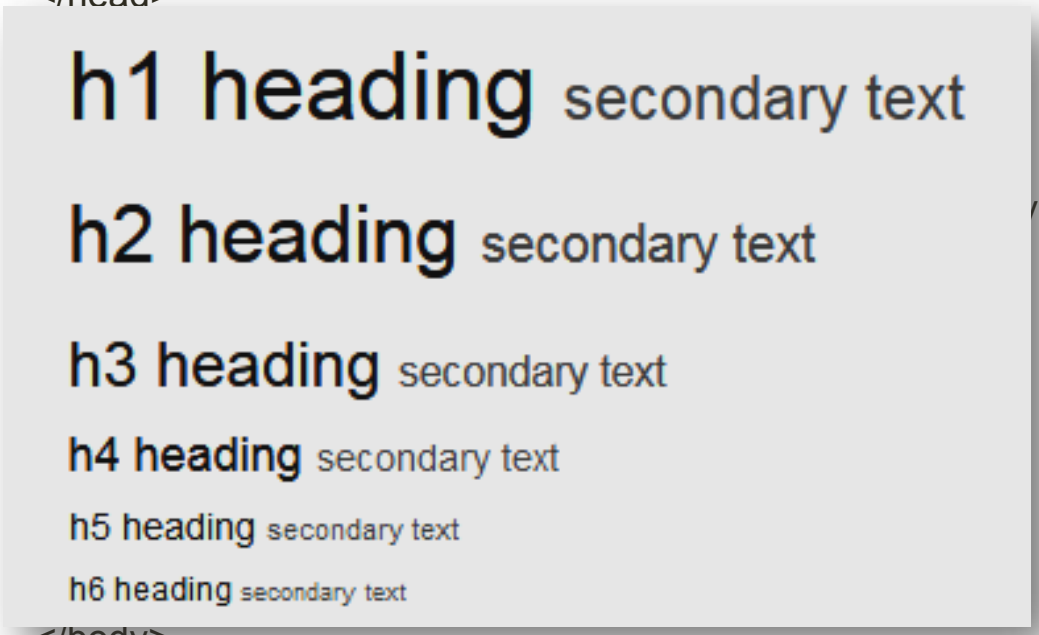
- Include Bootstrap from a CDN

Bootstrap-Example 1

```
<html >
<head>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
</head>
<body>
<div class="container">
  <h1>Lighter, Secondary Text</h1>
  <p>The small element is used to create a lighter, secondary text in any heading:</p>
  <h1>h1 heading <small>secondary  text</small></h1>
  <h2>h2 heading <small>secondary  text</small></h2>
  <h3>h3  heading  <small>secondary text</small></h3>
  <h4>h4  heading  <small>secondary text</small></h4>
  <h5>h5  heading  <small>secondary text</small></h5>
  <h6>h6  heading  <small>secondary text</small></h6>
</div>
</body>
</html>
```

Bootstrap-Example 1-O/P

```
<html >
<head>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
</head>
```




```
text in any heading:</p>
</body>
</html>
```

```
<div class="container">
  <h1>Code Snippets</h1>
  <p>The following HTML elements:
  <code>span</code>, <code>section</code>, and <code>div</code>
  defines a section in a document.</p>
</div>
```

Code Snippets

The following HTML elements: `span`, `section`, and `div` defines a section in a document.

Bootstrap-Example 3

```
<div class="container">
  <h1>Keyboard Inputs</h1>
  <p>Use <kbd>ctrl + p</kbd>
  to open the Print dialog box.</p>
</div>
```

Bootstrap-

Contextual Colors and Backgrounds

- Bootstrap also has some contextual classes that can be used to provide "meaning through colors".
- The classes for text colors are:
  - .text-muted,
  - .text-primary,
  - .text-success,
  - .text-info,
  - .text-warning, and
  - .text-danger:

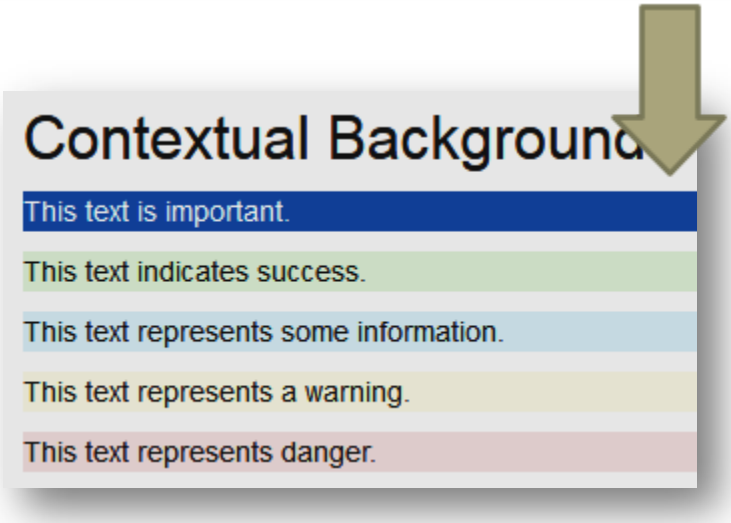
Backgrounds

```
<div class="container">
  <h2>Contextual Colors</h2>
  <p class="text-muted">This text is muted.</p>
  <p class="text-primary">This text is important.</p>
  <p class="text-success">This text indicates success.</p>
  <p class="text-info">This text represents some information.</p>
  <p class="text-warning">This text represents a warning.</p>
  <p class="text-danger">This text represents danger.</p>
</div>
```

Bootstrap- Backgrounds Example

- The classes for background colors are:.bg-primary, .bg-success, .bg-info, .bg-warning, and .bg-danger:

```
<div class="container">
  <h2>Contextual Backgrounds</h2>
  <p class="bg-primary">This text is important.</p>
  <p class="bg-success">This text indicates success.</p>
  <p class="bg-info">This text represents some information.</p>
  <p class="bg-warning">This text represents a warning.</p>
  <p class="bg-danger">This text represents danger.</p>
</div>
```



Outline of Advance Technology

- Bootstrap
- JSF
- Spring

JSF- JavaServer Faces

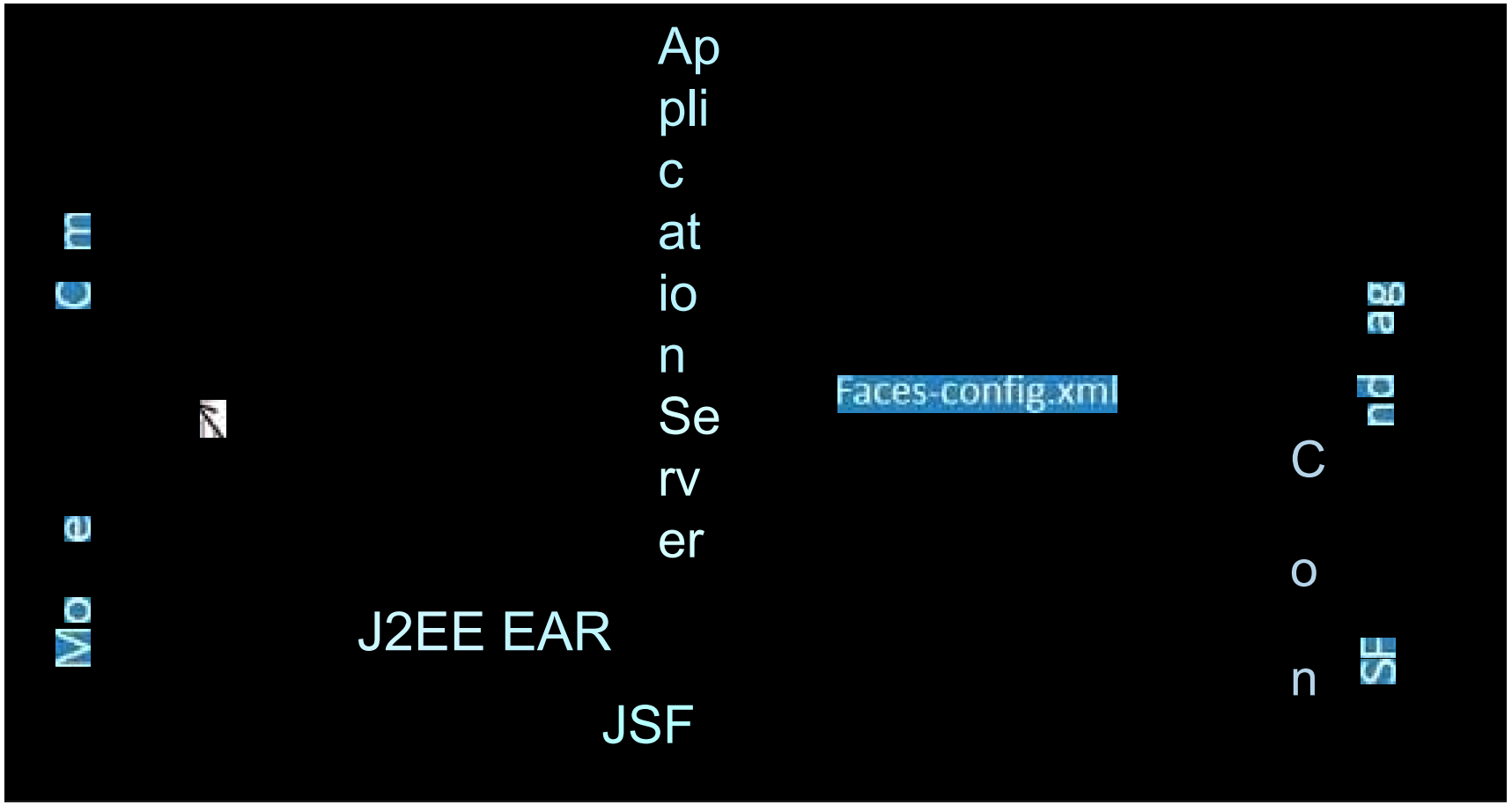
- **JavaServer Faces** (JSF) is a
- MVC web framework that simplifies the construction of User Interfaces (UI) for server-based applications using reusable UI components in a page.
- JSF provides a facility to connect UI widgets with data sources and to server-side event handlers.
- The JSF specification defines a set of standard UI components and provides an Application Programming Interface (API) for developing components.
- JSF enables the reuse and extension of the existing standard UI components.

JSF- Benefits

- JSF reduces the effort in creating and maintaining applications, JSF facilitates Web application development by −
  - Providing reusable UI components
  - Making easy data transfer between UI components
  - Managing UI state across multiple server requests
  - Enabling implementation of custom components
  - Wiring client-side event to server-side application code

JSF Architecture

- JSF contains −
  - **JavaBeans components** as models containing application-specific functionality and data
  - A **custom tag library** for representing event handlers and validators
  - A custom tag library for rendering UI components
  - UI components represented as stateful objects on the server
  - Server-side **helper classes**
  - **Validators, event handlers, and navigation handlers**
  - **Application configuration resource file** for configuring application resources


Web
pplic
ation
Ap
pli
c
at
io
n
Se
rv
er
Faces-config.xml
C
o
n
J2EE EAR
JSF

t r o l l e r

s . t « e s

s e r v l e t

Data

View

JSP Pages / I:fir
Cun1gurier›ts

Model

M‹1n.1¿eJ bean.t

- There are controllers which can be used to perform user actions. UI can be created by web page authors and the business logic can be utilized by managed beans.
- JSF provides several mechanisms for rendering an individual component. It is upto the web page designer to pick the desired representation, and the application developer doesn't need to know which mechanism was used to render a JSF UI component.

JSF - Life Cycle

- life cycle consists of six phases which are as follows −
  - Restore view phase
  - Apply request values phase; process events
  - Process validations phase; process events
  - Update model values phase; process events
  - Invoke application phase; process events
  - Render response phase

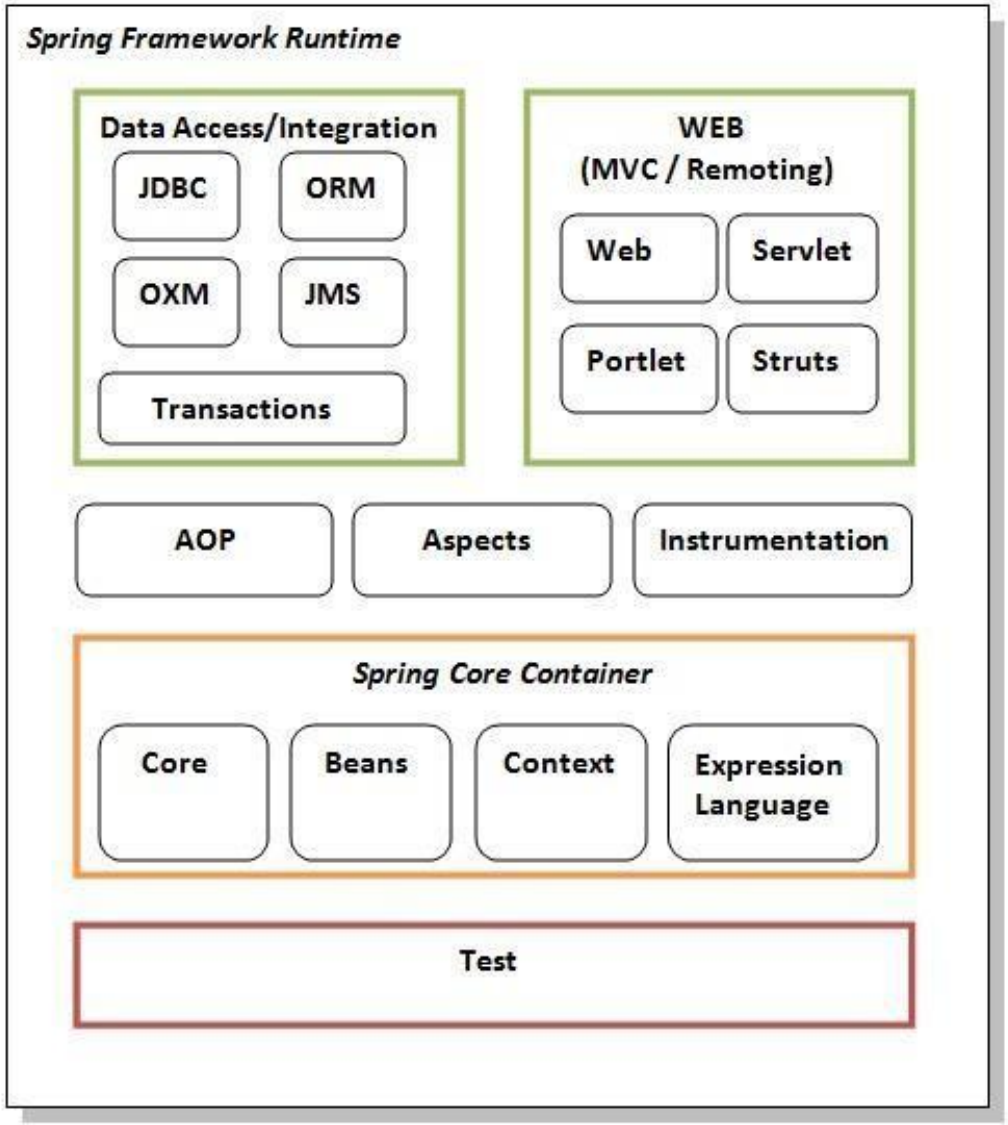
Outline of Advance Technology

- Bootstrap
- JSF
- Spring

Spring

- It was **developed by Rod Johnson in 2003**. Spring framework makes the easy development of JavaEE application.
- Spring is a ***lightweight*** framework.
- It can be thought of as a ***framework of frameworks*** because it provides support to various frameworks such as Struts, Hibernate, Tapestry, EJB, JSF etc.
- The framework, in broader sense, can be defined as a structure where we find solution of the various technical problems.
- The Spring framework comprises several modules such as IOC, AOP, DAO, Context, ORM, WEB MVC etc.

Spring Modules



Spring Modules

- **Test-** This layer provides support of testing with JUnit and TestNG.
- **Spring Core Container-** The Spring Core container contains core, beans, context and expression language (EL) modules.
- **Core and Beans-** These modules provide IOC and Dependency Injection features.
- **Context-** This module supports internationalization (I18N), EJB, JMS, Basic Remoting.
- **Expression Language-** It provides support to setting and getting property values, method invocation, accessing collections and indexers, named variables, logical and arithmetic operators, retrieval of objects by name etc.
- **AOP, Aspects and Instrumentation-**These modules support aspect oriented programming implementation where you can use Advices, Pointcuts etc. to decouple the code.
- The aspects module provides support to integration with AspectJ.
- The instrumentation module provides support to class instrumentation and classloader implementations.
- **Data Access / Integration-** This group comprises of JDBC, ORM, OXM, JMS and Transaction modules. These modules basically provide support to interact with the database.
- **Web-** This group comprises of Web, Web-Servlet, Web-Struts and Web-Portlet.

These modules provide support to create web application.

Spring-Dependency Injection (DI)

- The technology that Spring is most identified with is the **Dependency Injection (DI)** flavor of Inversion of Control.
- When writing a complex Java application, application classes should be as independent as possible of other Java classes.
- To increase the possibility to reuse these classes and to test them independently of other classes while unit testing.
- Dependency Injection helps in gluing these classes together and at the same time keeping them independent.

Spring- Aspect Oriented Programming (AOP)

- One of the key components of Spring is the **Aspect Oriented Programming (AOP)** framework.
- The functions that span multiple points of an application are called **cross-cutting concerns** and these cross-cutting concerns are conceptually separate from the application's business logic.
- There are various common good examples of aspects including logging, declarative transactions, security, caching, etc.
- The key unit of modularity in OOP is the class, whereas in AOP the unit of modularity is the aspect.
- DI helps you decouple your application objects from each other, while AOP helps you decouple cross-cutting concerns from the objects that they affect.

Spring Example

- simple steps to create the spring application
- **create the class**
- **create the xml file to provide the values**
- **create the test class**
- **Load the spring jar files**
- **Run the test class**

References

- https://www.tutorialspoint.com/webservices/what_are_web_services.htm
- https://www.javatpoint.com/what-is-web-service
- https://www.javatpoint.com/soap-web-services
- https://www.tutorialspoint.com/webservices/web_services_components.htm
- https://www.javatpoint.com/soap-web-services
- https://www.javatpoint.com/soap-vs-rest-web-services
- https://www.studytonight.com/rest-web-service/types-of-webservices
- https://www.tutorialspoint.com/ejb/ejb_overview.htm
- https://www.javatpoint.com/stateless-session-bean
- https://www.javatpoint.com/stateful-session-bean
- https://www.javatpoint.com/message-driven-bean
- https://docs.oracle.com/cd/B10002_01/generic.903/a97677/cmp.htm
- https://ejbvn.wordpress.com/category/week-1-enterprise-java-architecture/day-01-understanding-ejb-architecture/
- http://www.cs.vsb.cz/grygarek/dosys/ejb/slides/sld3.html
- http://www.oracle.com/technetwork/java/javaee/ejb/index.html
- https://en.wikipedia.org/wiki/Content_management_system
- https://www.tutorialspoint.com/wordpress/wordpress_overview.htm
- https://www.tutorialspoint.com/wordpress/wordpress_installation.htm
- https://www.tutorialspoint.com/wordpress/wordpress_dashboard.htm
- https://www.shivarweb.com/1819/how-to-use-the-wordpress-dashboard-initial-settings/
- https://www.tutorialspoint.com/joomla/joomla_overview.htm
- https://www.tutorialspoint.com/joomla/joomla_installation.htm
- https://www.tutorialspoint.com/joomla/joomla_architecture.htm
- https://www.w3schools.com/bootstrap/bootstrap_get_started.asp
- https://www.tutorialspoint.com/jsf/jsf_overview.htm
- https://www.tutorialspoint.com/jsf/jsf_life_cycle.htm
- https://www.tutorialspoint.com/jsf/jsf_architecture.htm
- https://www.javatpoint.com/spring-tutorial

- https://www.javatpoint.com/spring-modules
- https://www.tutorialspoint.com/spring/spring_overview.htm