**Application Server / App Server** (WebLogic, JBoss, WildFly, GlassFish):   
It holds big Enterprise application having big business logic. It is Heavy Weight, or it holds Heavy Weight Applications. It can handle all application operations between users and an organization's back end business applications or databases. It is frequently viewed as part of a three-tier application with: Presentation tier, logic tier, Data tier. App Server same as Web Server can also serve HTTP Content but is not limited to just HTTP. It can be provided other protocol support such as RMI/RPC  
Most of the application servers have Web Server as integral part of them, that means App Server can do whatever Web Server is capable of. Additionally, App Server have components and features to support Application level services such as Connection Pooling, Object Pooling, Transaction Support, Messaging services etc.  
As web servers are well suited for static content and app servers for dynamic content, most of the production environments have web server acting as reverse proxy to app server. That means while servicing a page request, static contents (such as images/Static HTML) are served by web server that interprets the request. Using some kind of filtering technique (mostly extension of requested resource) web server identifies dynamic content request and transparently forwards to app server  
An application server supports the whole JavaEE - EJB, JMS, CDI, JTA, the servlet API (including JSP, JSTL), etc.  
An application server runs WARs and EARs.

**Web Server / HTTP Server** (Apache HTTP Server, IIS, Nginx, LiteSpeed):   
A server which is capable of handling HTTP requests, sent by a client and respond back with a HTTP response. Web Server is designed to serve HTTP Content via HTTP protocol only. Web Server is mostly designed to serve static content, though most Web Servers have plugins to support scripting languages like Perl, PHP, ASP, JSP etc. through which these servers can generate dynamic HTTP content.

**Web Container / Servlet Container / Servlet Engine** (Apache Tomcat, Jetty):   
It is also a Web Server. It is used to manage the components like Servlets, JSP. It maintains the life cycle for Servlet Object. Calls the service method for that servlet object. pass the HttpServletRequest and HttpServletResponse Object. It is also responsible for mapping a URL to a particular servlet and ensuring that the URL requester has the correct access rights.  
A servlet-container supports only the servlet API (including JSP, JSTL).  
A web container runs only WARs.

**Web Server Programming Environment**

Apache PHP, CGI

IIS (Internet Information Server) ASP (.NET)

Tomcat Servlet

Jetty Servlet

**Application Server Programming Environment**

WAS (IBM's WebSphere Application Server) EJB

WebLogic Application Server (Oracle's) EJB

JBoss AS EJB

MTS COM+