

- Note: (i) Attempt ALL questions.
(ii) All questions carry equal marks.

Q. 1. Attempt any four parts of the following:

(5 × 4 = 20)

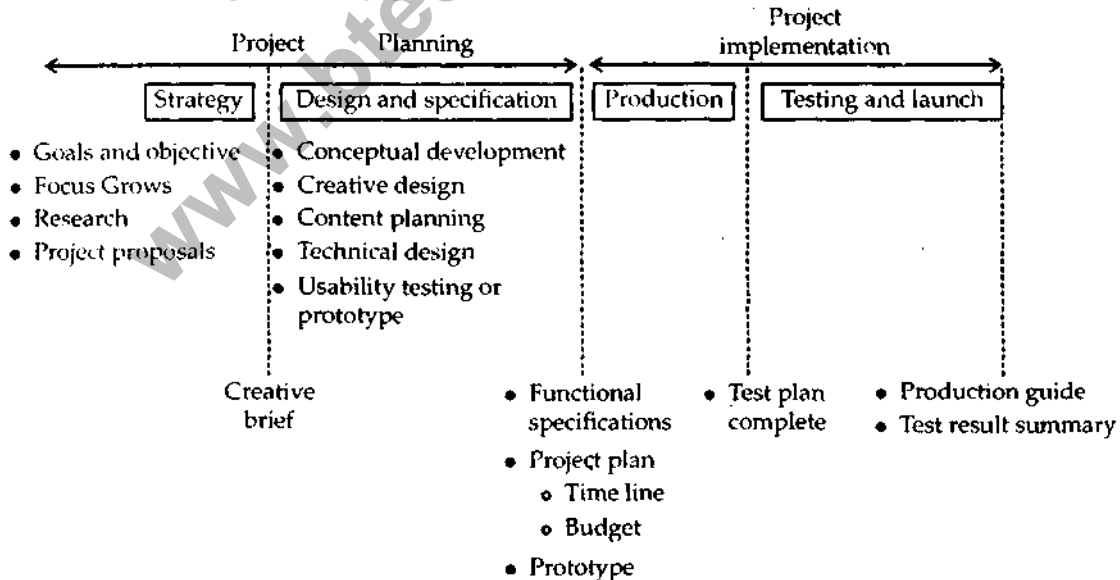
Q. (a). What do you mean by web technologies? Write down the various steps to develop a multidepartmental website.

Ans. **Web Technology:** It is a technology for developing new technologies designed to allow authors to write and deploy web applications, various technology vendors, the process rules by which each organisation produces standards varies:

STEPS TO DEVELOP MULTIDEPARTMENTAL WEB SITES:

(i) **Craft the look and feel:** The interface that a user must navigate often is called the look and feel of a web site. Users look and feel when they explore the information design of your site. They read text, make association with links, view graphics and depending on the freedom of your design, create their own part through your information.

(ii) **Make your design portable:** The web site must be portable and accessible across different browsers, operating systems and computer platforms, many designers make the mistake of testing in only one environment, assuming that their pages look the same to all of their users.



(iii) **Design for law and width:** Plan your pages so that they are accessible at a variety of connection speeds. If your pages download slowly because they contain large detailed graphics or complicated animations, your users will leave before they ever see your content.

(iv) **Plan for clear presentation and easy access to your information:** The presentation and organisation of your information is the single most important factor in determining the success of your site. Your graphics and navigation option the look and feel of your site must present a variety of option to the user without detracting from their quest for information.

Q. 1(b). Describe various protocols governing the web applications.

Ans. Protocols of Web applications:

HTTP: Hypertext Transfer Protocol (HTTP) is a method used to transfer or convey information on the World Wide Web. Its original purpose was to provide a way to publish and retrieve HTML pages.

HTTP is a request/response protocol between clients and servers. The originating client, such as a web browser, spider, or other end-user tool, is referred to as the user agent. The destination server, which stores or creates resources such as HTML files and images, is called the origin server. In between the user agent and origin server may be several intermediaries, such as proxies, gateways, and tunnels.

An HTTP client initiates a request by establishing a Transmission Control Protocol (TCP) connection to a particular port on a remote host (port 80 by default). An HTTP server listening on that port waits for the client to send a request message. Upon receiving the request, the server sends back a status line, such as "HTTP/1.1 200 OK", and a message of its own, the body of which is perhaps the requested file, an error message, or some other information.

SMTP: Simple Mail Transfer Protocol (SMTP) is the *de facto* standard for e-mail transmissions across the Internet. SMTP is a relatively simple, text-based protocol, where one or more recipients of a message are specified (and in most cases verified to exist) and then the message text is transferred. It is quite easy to test an SMTP server using the telnet program (see below).

SMTP uses TCP port 25. To determine the SMTP server for a given domain name, the MX (Mail eXchange) DNS record is typically used, falling back to a simple A record in the case of no MX (not all MTAs (Mail Transfer Agents) support fallback). Some current mail transfer agents will also use SRV records, a more general form of MX, though these are not widely adopted.

MIME: Multipurpose Internet Mail Extensions (MIME) is an Internet Standard that extends the format of e-mail to support:

- text in character sets other than US-ASCII;
- non-text attachments;
- multi-part message bodies; and
- header information in non-ASCII character sets.

Virtually all human-written Internet e-mail and a fairly large proportion of automated e-mail is transmitted via SMTP in MIME format. Internet e-mail is so closely associated with the SMTP and MIME standards that it is sometimes called SMTP/MIME e-mail.

The content types defined by MIME standards are also of importance outside of e-mail, such as in communication protocols like HTTP for the World Wide Web. HTTP requires that data be transmitted in

the context of e-mail-like messages, even though the data may not actually be e-mail.

FTP or File Transfer Protocol is used to transfer data from one computer to another over the Internet, or through a network.

Specifically, FTP is a commonly used protocol for exchanging files over any network that supports the TCP/IP protocol (such as the Internet or an intranet). There are two computers involved in an FTP transfer: a server and a client. The **FTP server**, running FTP server software, listens on the network for connection requests from other computers. The client computer, running FTP client software, initiates a connection to the server. Once connected, the client can do a number of file manipulation operations such as uploading files to the server, download files from the server, rename or delete files on the server and so on. Any software company or individual programmer is able to create FTP server or client software because the protocol is an open standard. Virtually every computer platform supports the FTP protocol. This allows any computer connected to a TCP/IP based network to manipulate files on another computer on that network regardless of which operating systems are involved. There are many existing FTP client and server programs. FTP servers can be setup anywhere between game servers, voice servers, internet hosts, and other physical servers.

TELNET (TELEtype NETwork) is a network protocol used on the Internet or local area network (LAN) connections. The term *telnet* also refers to software which implements the client part of the protocol. TELNET clients have been available on most Unix systems for many years and are available for virtually all platforms. Most network equipment and OSs with a TCP/IP stack support some kind of TELNET service server for their remote configuration

"To telnet" is also used as a verb meaning to establish or use a TELNET or other interactive TCP connection, as in, "To change your password, telnet to the server and run the passwd command".

Q. 1(c). What can be the stages and strategies required in order to develop a web project?

Ans. Stages and strategies for developing a web projects:

The most important rules of web design is: **keep your files sizes as small as possible**. There are lots of ways of doing this:

- **How long should a web page be?**

- Beware of the **toilet roll effect**: long, scrolling pages that go on forever. People don't read all that text, even if they wait around while the page loads.

- In particular, **your home page should be compact**: tell users what they're getting and provide links to the real content.

- The next level of pages can be longer and slower: by that stage, users should be confident that the information is worth downloading.

- Note: long pages *are* appropriate in some circumstances - a page like this one, for example, which you might want to print out for reference; course listings, handouts and reading lists are other examples.

- helps your pages load faster, too: succinct, direct pages take up less computer memory than long-winded, verbose pages.

- Speed up page-loading by providing layout information for images and tables.

- `` tells a computer that the page includes an image.

o `` tells the computer how much space to allocate for the image, allowing the rest of the page to load at the same time as the image. Similarly, if you specify the dimensions of a table the page will load more quickly.

o FrontPage helpfully specifies the dimensions of images automatically.

• **Big images and multimedia** elements can be very effective, but they can be very slow to load. Rather than include them on any main pages, put a link to them: anyone who follows the link will be more likely to wait for the page to load and you won't annoy other users by slowing down important pages. FrontPage includes an option for generating thumbnail links to big images to help you with this.

• Be wary of **Java applets** - some versions of Netscape lock up for tens of seconds while Java starts up and users either think their computer has crashed or get very frustrated while they're forced to wait... FrontPage's "hover button" option is an example of this: easy to add to a web page, but not a good thing.

• Choose **appropriate file formats for images**:

o **JPEGs** are great for detailed photographic images with lots of colours. Highly compressed JPEGs lose quality: try saving JPEGs at different levels of compression to find a good compromise between quality and file size (or use an optimising tool like GIFs are best for smaller images, or images with less detail and fewer colours - ideal for buttons, icons, graphs, etc. For smaller files, try reducing the number of colours in your GIFs: a three colour GIF is much smaller than a 103-colour GIF.

Use **images at their actual size!** Some web editing packages (FrontPage, for example...) make it easy to resize images on the screen: just click on the image and move the handles that appear at each corner. This specifies the width and height of the image in the web browser without affecting the file size, so you end up with a small image that takes ages to load. If you want your image to appear 100 pixels by 100 pixels, edit the image so that it is actually that size.

Q. 1(d). Explain the following terms:

Firewalls

Frames

JVM (Java Virtual Machines)

Byte Codes

Ans. Firewall: It is a security device which is configured to permit, deny or proxy data connections set and configured by the organization's security policy. Firewalls can either be hardware and/or software based.

A firewall's basic task is to control traffic between computer networks with different zones of trust. Typical examples are the Internet which is a zone with no trust and an internal network which is (and should be) a zone with high trust. The ultimate goal is to provide controlled interfaces between zones of differing trust levels through the enforcement of a security policy and connectivity model based on the least privilege principle and separation of duties.

Frames: framing means that a website can be organized into frames. Each frame displays a different HTML document. Headers and sidebar menus do not move when the content frame is scrolled up and down. For developers frames can also be convenient. For example, if an item needs to be added to the sidebar menu, only one file needs to be changed, whereas each individual page on a non-frameset

website would have to be edited if the sidebar menu appeared on all of them. However, server-side includes and scripting languages such as PHP can also be used to accomplish this aim without some of the drawbacks of frames such as confusing the operation of the address bar and back and forward buttons.

The contents of the frames may be hosted on the same server as the parent page, or it may link in code from another website server such that these external contents are automatically displayed within the frame (transclusion or *remote loading*). This may be confusing and inconvenient to the users: they can get the impression that the information belongs to the same website; also, less than the full browser window is available and the address bar is less informative. Some websites request not to be used in this way on other websites; some discourage it by including a framekiller script in its pages. The framing website runs a risk of being blamed for external content that, for example, is or becomes inaccurate or objectionable.

JVM:

Java Virtual Machine (JVM) is a virtual machine that interprets and executes Java bytecode. This code is most often generated by Java language compilers, although the JVM can also be targeted by compilers of other languages. JVMs may be developed by other companies as long as they adhere to the JVM standard published by Sun.

The JVM is a crucial component of the Java Platform. Because JVMs are available for many hardware and software platforms, Java can be both middleware and a platform in its own right — hence the expression “write once, run anywhere.” The use of the same bytecode for all platforms allows Java to be described as “compile once, run anywhere”, as opposed to “write once, compile anywhere”, which describes cross-platform compiled languages.

Byte Code:

Java bytecode is the form of instructions that the Java virtual machine executes. Each bytecode instruction is one byte in length (hence the name), thus the number of bytecodes is limited to 256. Not all 256 possible bytecode values are used. In fact, Sun Microsystems, the original creators of the Java programming language, the Java virtual machine and other components of the Java Runtime Environment, have set aside a number of values to be permanently unimplemented. The bytecodes are given below.

Q. 1(e). What are the various security threats on the web? How are they encountered?

Ans. Various security threats on the web:

Macro viruses: A macro virus, often written in the scripting languages for programs such as Word and Excel, is spread by infecting documents and spreadsheets. Since macro viruses are written in the language of the application and not in that of the operating system, they are known to be platform-independent. They can spread between Windows, Mac and any other system, so long as they are running the required application. With the ever-increasing capabilities of macro languages in applications, and the possibility of infections spreading over networks, these viruses are major threats.

Network viruses: This kind of virus is proficient in quickly spreading across a Local Area Network (LAN) or even over the Internet. Usually, it propagates through shared resources, such as shared drives and folders. Once it infects a new system, it searches for potential targets by searching the network for other vulnerable systems. Once a new vulnerable system is found, the network virus infects the other system,

and thus spreads over the network. Some of the most notorious network viruses are Nimda and SQLSlammer.

Logic bomb: A logic bomb employs code that lies inert until specific conditions are met. The resolution of the conditions will trigger a certain function (such as printing a message to the user and/or deleting files). Logic bombs may reside within standalone programs, or they may be part of worms or viruses. An example of a logic bomb would be a virus that waits to execute until it has infected a certain number of hosts. A *time bomb* is a subset of logic bomb, which is set to trigger on a particular date and/or time. An example of a time bomb is the infamous 'Friday the 13th' virus.

Cross-site scripting virus: cross-site scripting virus (XSSV) is a type of virus that utilizes cross-site scripting vulnerabilities to replicate. A XSSV is spread between vulnerable web applications and web browsers creating a symbiotic relationship

Sentinels: A sentinel is a highly advanced virus capable of empowering the creator or perpetrator of the virus with remote access control over the computers that are infected. They are used to form vast networks of zombie or slave computers which in turn can be used for malicious purposes such as a Distributed Denial-of-service attack.

Companion virus: A companion virus does not have host files per se, but exploits MS-DOS. A companion virus creates new files (typically .COM but can also use other extensions such as ".EXD") that have the same file names as legitimate .EXE files. When a user types in the name of a desired program, if a user does not type in ".EXE" but instead does not specify a file extension, DOS will assume he meant the file with the extension that comes first in alphabetical order and run the virus. For instance, if a user had "(filename).COM" (the virus) and "(filename).EXE" and the user typed "filename", he will run "(filename).COM" and run the virus. The virus will spread and do other tasks before redirecting to the legitimate file, which operates normally.

Boot sector viruses: A boot sector virus alters or hides in the boot sector, usually the 1st sector, of a bootable disk or hard drive. The boot sector is where your computer starts reading your operating system. By inserting its code into the boot sector, a virus guarantees that it loads into memory during every boot sequence. A boot virus does not affect files; instead, it affects the disks that contain them.

Multipartite viruses: Multipartite viruses are a combination of boot sector viruses and file viruses. These viruses come in through infected media and reside in memory. They then move on to the boot sector of the hard drive. From there, the virus infects executable files on the hard drive and spreads across the system.

Q. 1(f). Write down about the cyber law in India.

Ans. Cyber Laws in India:

In May 2000, both the houses of the Indian Parliament passed the Information Technology Bill. The Bill received the assent of the President in August 2000 and came to be known as the Information Technology Act, 2000. Cyber laws are contained in the IT Act, 2000.

- 1 From the perspective of e-commerce in India, the IT Act 2000 and its provisions contain many positive aspects. Firstly, the implications of these provisions for the e-businesses would be that email would now be a valid and legal form of communication in our country that can be duly produced and approved in a court of law.
 - 2 Companies shall now be able to carry out electronic commerce using the legal infrastructure provided by the Act.
 - 3 Digital signatures have been given legal validity and sanction in the Act.
 - 4 The Act throws open the doors for the entry of corporate companies in the business of being Certifying Authorities for issuing Digital Signatures Certificates.
 - 5 The Act now allows Government to issue notification on the web thus heralding e-governance.
 - 6 The Act enables the companies to file any form, application or any other document with any office, authority, body or agency owned or controlled by the appropriate Government in electronic form by means of such electronic form as may be prescribed by the appropriate Government.
 - 7 The IT Act also addresses the important issues of security, which are so critical to the success of electronic transactions. The Act has given a legal definition to the concept of secure digital signatures that would be required to have been passed through a system of a security procedure, as stipulated by the Government at a later date.
 - 8 Under the IT Act, 2000, it shall now be possible for corporates to have a statutory remedy in case if anyone breaks into their computer systems or network and causes damages or copies data. The remedy provided by the Act is in the form of monetary damages, not exceeding Rs. 1 crore.
- 1 the Act specifically stipulates that any subscriber may authenticate an electronic record by affixing his digital signature. It further states that any person can verify an electronic record by use of a public key of the subscriber.
 - 2 the Act details about Electronic Governance and provides inter alia amongst others that where any law provides that information or any other matter shall be in writing or in the typewritten or printed form, then, notwithstanding anything contained in such law, such requirement shall be deemed to have been satisfied if such information or matter is -
 - ⊗ rendered or made available in an electronic form; and
 - ⊕ accessible so as to be usable for a subsequent reference
 It also details the legal recognition of Digital Signatures.
 - 3 Act gives a scheme for Regulation of Certifying Authorities. The Act envisages a Controller of Certifying Authorities who shall perform the function of exercising supervision over the activities of the Certifying Authorities as also laying down standards and conditions governing the Certifying Authorities as also specifying the various forms and content of Digital Signature Certificates. The Act recognizes the need for recognizing foreign Certifying Authorities and it further details the various provisions for the issue of license to issue Digital Signature Certificates.
 - 4 Act details about the scheme of things relating to Digital Signature Certificates. The duties of subscribers are also enshrined in the said Act.
 - 5 Act talks about penalties and adjudication for various offences. The penalties for damage to computer, computer systems etc. has been fixed as damages by way of compensation not exceeding Rs. 1,00,00,000 to affected persons. The Act talks of appointment of any officers not below the rank of a Director to the Government of India or an equivalent officer of state government as an Adjudicating Officer who shall adjudicate whether any person has made a contravention of any of the provisions of the said Act or rules framed there under. The said Adjudicating Officer has been given the powers of a Civil Court.
 - 6 The Act talks of the establishment of the Cyber Regulations Appellate Tribunal, which shall be an appellate body where appeals against the orders passed by the Adjudicating Officers, shall be preferred.
 - 7 the Act talks about various offences and the said offences shall be investigated only by a Police Officer not below the rank of the Deputy Superintendent of Police. These offences include tampering with computer source documents, publishing of information, which is obscene in electronic form, and hacking.
 - 8 The Act also provides for the constitution of the Cyber Regulations Advisory Committee, which shall advice the government as regards any rules, or for any other purpose connected with the said act. The said Act also proposes to amend the Indian Penal Code, 1860, the Indian Evidence Act, 1872, The Bankers' Books Evidence Act, 1891, The Reserve Bank of India Act, 1934 to make them in tune with the provisions of the IT Act.

Q. 2. Attempt any four parts of the following:

(5 × 4 = 20)

Q. (a). Discuss the design of the railway reservation application using XML, HTML, Java Script and ASP in brief.

Ans. Using XML:

```
>?xml>
version = "1.0"
?>
<Passengers>
  <Person>
    <Name>
      <First>----- </First>
      <Last>----- </Last>
    </Name>
    <Address>-----</Address>
    <Phone>-----</Phone>
    <Sex>-----</Sex>
    <No. of Bearth>-----</No. of Bearth>
    <Class of journey>-----</Class of journey>
  </Person>
</People>
<HTML>
<Head>
<Script>
  Function reservation (frm)
  {
    var val = frm.txt.value;
    frm.txt.value = '';
    frm.area.value += val + '\n'
  }
</Script>
</Head>
<Body>
  <Form-name = "reservation form">
  <Input-type = 'text', Name = 'txt'>
  On click = 'add 70 Text Area (this form)'>
  <BR>
  Text Area Name = 'area' ROWS = '5' COLOS = '65'>
  </Text Area> <BR> </Form> </Body> </HTML>
```

Q. 2(b). Differentiate between the following pairs:

- (i) HTML and XML
- (ii) Constructors and finalizers

Ans. HTML And XML:

Hypertext Markup Language, is the predominant markup language for the creation of web pages. It

provides a means to describe the structure of text-based information in a document — by denoting certain text as headings, paragraphs, lists, and so on — and to supplement that text with interactive forms, embedded images, and other objects. HTML is written in the form of labels (known as tags), surrounded by less-than (<) and greater-than (>) signs. HTML can also describe, to some degree, the appearance and semantics of a document, and can include embedded scripting language code which can affect the behavior of web browsers and other HTML processors.

The **Extensible Markup Language (XML)** is a general-purpose markup language. Its primary purpose is to facilitate the sharing of data across different information systems, particularly via the Internet.

It is a simplified subset of the Standard Generalized Markup Language (SGML), and is designed to be relatively human-legible. By adding semantic constraints, application languages can be implemented in XML. These include XHTML,^[3] RSS, MathML, GraphML, Scalable Vector Graphics, MusicXML, and thousands of others. Moreover, XML is sometimes used as the specification language for such application languages.

- Constructors Vs Finalizers:

In object oriented programming, a **constructor** in a class is a special block of statements called when an object is created, either when it is declared (statically constructed on the stack, possible in C++ but not in Java and other object-oriented languages) or dynamically constructed on the heap through the keyword "new".

A constructor is similar to a class method, but it differs from a method in that it never has an explicit return type, it's not inherited, and usually has different rules for modifiers. Constructors are often distinguished by having the same name as the declaring class. Their responsibility is to pre-define the object's data members and to establish the invariant of the class, failing if the invariant isn't valid. A properly written constructor will leave the object in a 'valid' state. Immutable objects must be initialized in a constructor.

In object-oriented programming languages that use garbage collection, a **finalizer** is a special method that is executed when an object is garbage collected. It is similar in function to a destructor. In less technical terms, a finalizer is a piece of code that ensures that certain necessary actions are taken when an acquired resource (such as a file or access to a hardware device) is no longer being used. This could be closing the file or signalling to the operating system that the hardware device is no longer needed. However, as noted below, finalizers are not the preferred way to accomplish this and for the most part serve as a fail-safe.

Unlike destructors, finalizers are not deterministic. A destructor is run when the program explicitly frees an object. A finalizer, by contrast, is executed when the internal garbage collection system frees the object.

Q. 2(c). What do you mean by function overloading? Explain with examples.

Ans. **Function overloading** is a feature found in various object oriented programming languages such as C#, C++ and Java that allows the creation of several functions with the same name which differ from each other in terms of the type of the input and the type of the output of the function.

Function overloading is usually associated with statically-typed programming languages which enforce type checking in function calls. When overloading a method, you are really just making a number of different methods that happen to have the same name. It is resolved at compile time which of these methods are used.

Function overloading should not be confused with ad-hoc polymorphism or virtual functions. In those, the correct method is chosen at runtime.

Example:

```
Class a
{
public void abc()
{
    system.console.writeline("Hi");
}
}
Class b:a
{
public void abc()
{
    System.Console.WriteLine("Bye");
}
}
public static void Main()
{
    b obj = new b();
    obj.abc();
}
}
```

Q. 2(d). Describe various types of Document Type Definition (DTD) in XML. Which types of DTD is more preferable and why?

Ans. DTD: The oldest schema format for XML is the Document Type Definition (DTD), inherited from SGML. While DTD support is ubiquitous due to its inclusion in the XML 1.0 standard, it is seen as limited for the following reasons:

- It has no support for newer features of XML, most importantly namespaces.
- It lacks expressivity. Certain formal aspects of an XML document cannot be captured in a DTD.
- It uses a custom non-XML syntax, inherited from SGML, to describe the schema.

Document Type Definition (DTD), defined slightly differently within the XML and SGML specifications, is one of several SGML and XML schema languages, and is also the term used to describe a document or portion thereof that is authored in the DTD language. A DTD is primarily used for the expression of a schema via a set of declarations that conform to a particular markup syntax and that describe a class, or *type*, of SGML or XML documents, in terms of constraints on the structure of those documents. A DTD may also declare constructs that are not always required to establish document structure, but that may affect the interpretation of some documents.

DTD is native to the SGML and XML specifications, and since its introduction other specification languages such as XML Schema and RELAX NG have been released with additional functionality.

As an expression of a schema, a DTD specifies, in effect, the syntax of an "application" of SGML or XML, such as the derivative language HTML or XHTML. This syntax is usually a less general form of the syntax of SGML or XML.

In a DTD, the structure of a class of documents is described via element and attribute-list declarations. Element declarations name the allowable set of elements within the document, and specify whether and how declared elements and runs of character data may be contained within each element. Attribute-list declarations name the allowable set of attributes for each declared element, including the type of each

attribute value, if not an explicit set of valid value(s).

Q. 2(e). What are the various ways to create forms? Explain with examples.

Ans. Ways to Create Form:

Forms require two pieces: 1) a **web page** with text boxes or checkboxes where people type the information, and 2) a **separate computer program** to take that information and process it.

Creating the form that collects the information is fairly easy. Web editors like DreamWeaver or GoLive can help you create the form. Other web editors, like Netscape Composer, can't create forms, so you will have to learn a few HTML tags to create the form.

To Begin

1. Create a new web page. Include your title, page color, opening text or any non-form information.
2. Save the file.
3. Open WordPad, SimpleText or any other text editor.
4. Open the file you just created.
5. Move to the place where you want the form fields to begin.

The First Tag: The first tag of the FORM section is the FORM tag itself. It describes the ACTION and identifies the program to be used to process the form. When you use FormMail.pl, that tag looks like this:

```
<FORM action="http://scripts.uvm.edu/cgi-bin/FormMail.pl" method="POST">
```

Fields Necessary to All Forms

You can create forms with any number of fields but there are two fields that you should include. One field allows people to reset the form. The other lets them actually send the information. These are called the "Reset" and "Submit" fields. They appear as buttons in the form. The actual words that appear on those buttons can be any word or phrase you like, that is, they do not have to say submit or reset. For example: `<INPUT TYPE="submit" NAME="submit" VALUE="Register for the Conference Now!">`
`<INPUT TYPE="reset" VALUE="Reset">`

Q. 2(f). Write a JavaScript function `int findsum(f)` to find the sum of the first `n` prime integers. The number `n` is the value of an element called `number` in the form of `f`.

Ans.

```
<HTML>
<Head> </Head>
<Script language = "Java script">
Function findsum (f)
{
    document.write ("Sum is "+f)
}
Function prime (f)
{
    gr = new array ()
    if ((f%gr)! = 0)
    {
        m m = new findsum (10)
        mm.f = mm.f | this.f;
    }
}
</Script>
</Body>
</HTML>
```

Q. 3. Attempt any two parts of the following:

(10 × 2 = 20)

Q. (a). What is a thread? How a thread is created? Differentiate between process based multitasking and thread based multitasking?

Ans. Thread:

Threads are a way for a program to fork (or split) itself into two or more simultaneously (or pseudo-simultaneously) running tasks. Threads and processes differ from one operating system to another, but in general, the way that a thread is created and shares its resources is different from the way a process does. Multiple threads can be executed in parallel on many computer systems. This *multithreading* generally occurs by time slicing, wherein a single processor switches between different threads, in which case the processing is not literally simultaneous, for the single processor is only really doing one thing at a time. This switching can happen so fast as to give the illusion of simultaneity to an end user.

Part-ii

Threads are distinguished from traditional multi-tasking operating system processes in that processes are typically independent, carry considerable state information, have separate address spaces, and interact only through system-provided inter-process communication mechanisms. Multiple threads, on the other hand, typically share the state information of a single process, and share memory and other resources directly. Context switching between threads in the same process is typically faster than context switching between processes. Systems like Windows NT and OS/2 are said to have “cheap” threads and “expensive” processes; in other operating systems there is not so great a difference.

Multithreading is a popular programming and execution model that allows multiple threads to exist within the context of a single process, sharing the process’ resources but able to execute independently. The threaded programming model provides developers with a useful abstraction of concurrent execution. However, perhaps the most interesting application of the technology is when it is applied to a *single* process to enable *parallel execution* on a *multiprocessor* system.

A *process* is the “heaviest” unit of kernel scheduling. Processes own resources allocated by the operating system. Resources include memory, file handles, sockets, device handles, and windows. Processes do not share address spaces or file resources except through explicit methods such as inheriting file handles or shared memory segments, or mapping the same file in a shared way. Processes are typically pre-emptively multitasked. However, Windows 3.1 and older versions of Mac OS used co-operative or non-preemptive multitasking.

Q. 3(b). What is ESB? How does it differ with Java Beans? Write down steps to design Java Bean and Servlet API.

Ans. EJB: Enterprise Java Bean (EJB) is a managed, server-sided component for modular construction of enterprise applications.

The EJB specification is one of the several Java APIs in the Java Platform, Enterprise Edition. EJB is a server-side component that encapsulates the business logic of an application. The EJB specification was originally developed in 1997 by IBM and later adopted by Sun Microsystems (EJB 1.0 and 1.1) and enhanced under the Java Community Process as JSR 19 (EJB 2.0), JSR 153 (EJB 2.1) and JSR 220 (EJB 3.0).

The EJB specification intends to provide a standard way to implement the back-end ‘business’ code typically found in enterprise applications (as opposed to ‘front-end’ user-interface code). Such code was frequently found to reproduce the same types of problems, and it was found that solutions to these

problems are often repeatedly re-implemented by programmers. Enterprise Java Beans were intended to handle such common concerns as persistence, transactional integrity, and security in a standard way, leaving programmers free to concentrate on the particular program at hand.

Accordingly, the EJB specification details how an application server provides:

- Persistence
- Transaction processing
- Concurrency control
- Events using Java Message Service
- naming and directory services (JNDI)
- Security (JCE and JAAS)
- Deployment of software components in an application server
- Remote procedure calls using RMI-IIOP.
- Exposing business methods as Web Services.

Part-ii

JavaBeans are classes written in the Java programming language. They are used to encapsulate many objects into a single object (the bean), so that the bean can be passed around rather than the individual objects.

The specification by Sun Microsystems defines them as "reusable software components that can be manipulated visually in a builder tool". In spite of many similarities, JavaBeans should not be confused with Enterprise JavaBeans (EJB), a server-side component technology that is part of Java EE.

Java Servlet API allows a software developer to add *dynamic* content to a Web server using the Java platform. The generated content is commonly HTML, but may be other data such as XML. Servlets are the Java counterpart to non-Java dynamic Web content technologies such as PHP, CGI and ASP. NET. Servlets can maintain state across many server transactions by using HTTP cookies, session variables or URL rewriting.

The Servlet API, contained in the Java package hierarchy `javax.servlet`, defines the expected interactions of a Web container and a servlet. A Web container is essentially the component of a Web server that interacts with the servlets. The Web container is responsible for managing the lifecycle of servlets, mapping a URL to a particular servlet and ensuring that the URL requester has the correct access rights.

A Servlet is an object that receives a request (`ServletRequest`) and generates a response (`ServletResponse`) based on the request. The API package `javax.servlet.http` defines HTTP subclasses of the generic servlet (`HttpServlet`) request (`HttpServletRequest`) and response (`HttpServletResponse`) as well as a session (`HttpSession`) that tracks multiple requests and responses between the Web server and a client. Servlets may be packaged in a WAR file as a Web application.

Q. 3(c). What are Servlets? Write down the major services and advantages of servlets.

Ans. **Java Servlet** allows a software developer to add *dynamic* content to a Web server using the Java platform. The generated content is commonly HTML, but may be other data such as XML. Servlets are the Java counterpart to non-Java dynamic Web content technologies such as PHP, CGI and ASP. NET. Servlets can maintain state across many server transactions by using HTTP cookies, session variables or URL rewriting.

The Servlet, contained in the Java package hierarchy `javax.servlet`, defines the expected interactions of

a Web container and a servlet. A Web container is essentially the component of a Web server that interacts with the servlets. The Web container is responsible for managing the lifecycle of servlets, mapping a URL to a particular servlet and ensuring that the URL requester has the correct access rights.

A Servlet is an object that receives a request (ServletRequest) and generates a response (ServletResponse) based on the request. The API package `javax.servlet.http` defines HTTP subclasses of the generic servlet (HttpServlet) request (HttpServletRequest) and response (HttpServletResponse) as well as a session (HttpSession) that tracks multiple requests and responses between the Web server and a client. Servlets may be packaged in a WAR file as a Web application.

However, servlets can be generated automatically by JavaServer Pages (JSP), or alternately by template engines such as WebMacro. Often servlets are used in conjunction with JSPs in a pattern called "Model 2", which is a flavor of the model-view-controller pattern.

Part-ii The Servlet life cycle consists of the following steps:

1. The Servlet class is loaded by the container during start-up.
2. The container calls the `init()` method. This method initializes the servlet and must be called before the servlet can service any requests. In the entire life of a servlet, the `init` method is called only once.
3. After initialization, the servlet can service client-requests. Each request is serviced in its own separate thread. The container calls the `service()` method of the servlet for every request. The `service()` method determines the kind of HTTP request (GET, POST etc) and accordingly calls the methods `doGet()`, `doPost()`, `doTrace()` etc. The developer of the servlet must provide implementation for these methods. If an implementation for `doPost()` has not been provided, it means that the servlet cannot handle POST requests. In such a situation if a POST request is made, the implementation of the parent class will be invoked. By default, this will throw a BAD HTTP Request exception. A developer must never overload the `service()` method.
4. Finally, the container calls the `destroy()` method which takes the servlet out of service. The `destroy()` method like `init()` is called only once in the life-cycle of a Servlet.

ServletConfig and ServletContext

There is only one ServletContext in every application. This object can be used by all the servlets to obtain application level information or container details. Every servlet, on the other hand, gets its own ServletConfig object. This object provides initialization parameters for a servlet. A developer can obtain the reference to ServletContext using either the ServletConfig object or ServletRequest object.

Q. 4. Attempt any two parts of the following: (10 × 2 = 20)

Q. (a). What do you mean by JSP? Explain the architecture of the JSP.

Ans. JSP:

JavaServer Pages (JSP) is a Java technology that allows software developers to dynamically generate HTML, XML or other types of documents in response to a Web client request. The technology allows Java code and certain pre-defined actions to be embedded into static content.

The JSP syntax adds additional XML-like tags, called JSP actions, to be used to invoke built-in functionality. Additionally, the technology allows for the creation of JSP tag libraries that act as extensions to the standard HTML or XML tags. Tag libraries provide a platform independent way of extending the capabilities of a Web server.

JSPs are compiled into Java Servlets by a JSP compiler. A JSP compiler may generate a servlet in Java code that is then compiled by the Java compiler, or it may generate byte code for the servlet directly.

Part-ii

JSP architecture

JSPs are built on top of SUN Microsystems' servlet technology. JSPs are essentially an HTML page with special JSP tags embedded. These JSP tags can contain Java code. The JSP file extension is .jsp rather than .htm or .html. The JSP engine parses the .jsp and creates a Java servlet source file. It then compiles the source file into a class file, this is done the first time and this is why the JSP is probably slower the first time it is accessed. Any time after this the special compiled servlet is executed and is therefore returned faster.

Steps required for a JSP request:

1. The user goes to a web site made using JSP. The user goes to a JSP page (ending with .jsp). The web browser makes the request via the Internet.
2. The JSP request gets sent to the Web server.
3. The Web server recognises that the file required is special (.jsp), therefore passes the JSP file to the JSP Servlet Engine.
4. If the JSP file has been called the first time, the JSP file is parsed, otherwise go to step 7.
5. The next step is to generate a special Servlet from the JSP file. All the HTML required is converted to println statements.
6. The Servlet source code is compiled into a class.
7. The Servlet is instantiated, calling the init and service methods.
8. HTML from the Servlet output is sent via the Internet.
9. HTML results are displayed on the user's web browser.

Q. 4(b). Explain the use of Request and Response objects in the communication between the browser and the web server in JSP.

Ans. Communication Between Browser and web server in JSP:

You can get information about a client's computer

Copy the code below and place in a file named: clientinfo.jsp

Run it from your browser -clientinfo.jsp

You should see the results of your submission

```
<html>
<head>
<title>VisualBuilder.com</title>
</head>
<body>
Client computer details:
<br><br>
<b>Ip address</b>:
<br>
<%=request.getRemoteAddr()%>
<br><br>
<b>Computer name</b>:
<br>
<%=request.getRemoteHost()%>
```

```
<br><br>
</body>
</html>
```

JSP enables the dynamic generation of web pages, similar to ASP.NET and PHP.

When someone visits a JSP page, the HTML web page is generated and sent back to the visitor. Below you can see the construction of a basic web page. The HTML source code on the side shows the output "The date today is".

In order for a static web page to show today's date, it would need to be edited every day and uploaded to the web server. This is very time consuming for such a simple task. This is where the dynamic generation of web pages is used.

In the JSP above, special syntax is used to signify that the current date needs to be displayed. You have probably seen this in the JSP tutorial already. This special syntax is processed on the web server and sent back to the visitor as a normal (html) web page.

Q. 4(c). How error handling and debugging are done in Java? Explain with suitable examples.

Ans. Error Handling and Debugging In Java:

An **Error** indicates that a non-recoverable error has occurred that should not be caught. Errors usually cause the Java virtual machine to display a message and exit.

Sun says the same thing in a slightly different way:

"An Error is a subclass of Throwable that indicates serious problems that a reasonable application should not try to catch. Most such errors are abnormal conditions."

Checked by the compiler

Your code must anticipate and either handle or declare checked exceptions. Otherwise, your program won't compile. *(These are exception types that are checked by the compiler.)*

Throwable constructors and methods

As mentioned above, all errors and exceptions are subclasses of the **Throwable** class. As of JDK 1.4.0, the **Throwable** class provides four constructors and about a dozen methods. The four constructors are as follows:

```
Throwable()
Throwable(String message)
Throwable(String message,
    Throwable cause)
Throwable(Throwable cause)
```

The first two constructors have been in Java for a very long time. Basically, these two constructors allow you to construct an exception object with, or without a **String** message encapsulated in the object.

Inherited methods

All exception objects inherit the methods of the **Throwable** class, which are listed in Figure 2. Thus, any of these methods may be invoked by the code in the **catch** block in its attempt to successfully handle the exception.

For example, exceptions may have a message encapsulated in the exception object, which can be accessed using the **getMessage** method. You can use this to display a message describing the error or exception.

You can also use other methods of the **Throwable** class to:

- Display a stack trace showing where the exception or error occurred
- Produce a **String** representation of the exception object

Part-ii

Debugging Applications

In some cases, the exceptions thrown by your BDB XML application may not contain enough information to allow you to debug the source of an error. In this case, you can cause BDB XML to issue more information using the error stream.

`setLogLevel()` allows you to indicate the level of logging that you want to see (debug, info, warning, error, or all of these).

`setLogCategory()` allows you to indicate the portions of DB XML's subsystems for which you want logging messages issued (indexer, query processor, optimizer, dictionary, container, or all of these)

Q. 5. Attempt any two parts of the following:

(10 × 2 = 20)

Q. (a). What are the important issues to be considered in designing a database that is to be accessed through a web page? Explain.

Ans. Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a Data Definition Language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity.

The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data - in the relational model these are the tables and views. However, the term database design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall database application within the Database Management System or DBMS.

Design process

The process of doing database design generally consists of a number of steps which will be carried out by the database designer. Not all of these steps will be necessary in all cases, but most usually, the designer must:

- Determine the data to be stored in the database
- Determine the relationships between the different data elements
- Superimpose a logical structure upon the data on the basis of these relationships.

Within the relational model the final step can generally be broken down into two further steps, that of determining the grouping of information within the system, generally determining what are the basic objects about which information is being stored, and then determining the relationships between these group of information, or objects.

The tree structure of data may enforce hierarchical model organization, with parent-child relationship table.

Q. 5(b). Design and write a program "Hello World!" using JSP Struts Framework.

Ans. JSP Struts Framework:

Apache Struts is a free open-source framework for creating Java web applications. Web applications

differ from conventional websites in that web applications can create a dynamic response. Many websites deliver only static pages. A web application can interact with databases and business logic engines to customize a response.

Web applications based on JavaServer Pages sometimes commingle database code, page design code, and control flow code. In practice, we find that unless these concerns are separated, larger applications become difficult to maintain.

One way to separate concerns in a software application is to use a Model-View-Controller (MVC) architecture. The *Model* represents the business or database code, the *View* represents the page design code, and the *Controller* represents the navigational code. The Struts framework is designed to help developers create web applications that utilize a MVC architecture.

The framework provides three key components:

- A "request" handler provided by the application developer that is mapped to a standard URL.
- A "response" handler that transfers control to another resource which completes the response.
- A tag library that helps developers create interactive form-based applications with server pages.

The framework's architecture and tags are buzzword compliant. Struts works well with conventional REST applications and with nouveau technologies like SOAP and AJAX.

Struts for Newbies

If you are new to development with Struts, here are some pointers on how to get started. The framework "stands on the shoulders of giants". To use Struts well, it's important to have a good grasp of the fundamentals. Start by reviewing the Key Technologies primer, and studying any unfamiliar topics.

Next, branch to either the Struts 2 or Struts 1 documentation

The Struts 2 documentation is maintained as a wiki, but don't let that stop you from scrolling through. We have organized the wiki so that it can be read like a book, cover to cover. Just keep following the Next links.

For Struts 1, the most up to date book would be Struts: The Complete Reference. For newbies, Struts for Dummies is another good choice.

For Struts 2, WebWork in Action is a worthwhile resource,

Q. 5(c). In context to JDBC, explain the following terms:

- **JDBC connectivity drivers**
- **Data sources**
- **SQL statements**

Ans.

- **(i) JDBC Connectivity Drivers:**

JDBC drivers are divided into four types or levels. The four types are:

Type 1: JDBC-ODBC Bridge driver (Bridge)

Type 2: Native-API/partly Java driver (Native)

Type 3: All Java/Net-protocol driver (Middleware)

Type 4: All Java/Native-protocol driver (Pure)

Type 1: JDBC-ODBC Bridge driver

The Type 1 driver translates all JDBC calls into ODBC calls and sends them to the ODBC driver. ODBC is a generic API. The JDBC-ODBC Bridge driver is recommended only for experimental use or when no other alternative is available.

Type 1: JDBC-ODBC Bridge

Type 2: Native-API/partly Java driver

The Type 2 driver converts JDBC calls into database-specific calls i.e. this driver is specific to a particular database. Example: Oracle will have oracle native api.

Type 2: Native api/ Partly Java Driver

Type 3: All Java/Net-protocol driver

Type 3 database requests are passed through the network to the middle-tier server. The middle-tier then translates the request to the database. If the middle-tier server can in turn use Type1, Type 2 or Type 4 drivers.

Type 3: All Java/ Net-Protocol Driver

Type 4: Native-protocol/all-Java driver

The Type 4 uses java networking libraries to communicate directly with the database server.

Type 4: Native-protocol/all-Java driver

Ans. (ii) Data Sources:

It is the origin of information found in electronic media. Often when data is captured in one electronic system and then transferred to another, there is a loss of audit trail or the inherent data cannot be absolutely verified. There are systems that provide for absolute data export but then the system imported into has to allow for all available data fields to be imported. Similarly, there are transaction logs in many modern database systems. The acceptance of these transaction records into any new system could be very important for any verification of such imported data.

Ans. (iii). SQL Statements:

SQL keywords fall into several groups.

Data retrieval

The most frequently used operation in transactional databases is the data retrieval operation. When restricted to data retrieval commands, SQL acts as a declarative language:

- SELECT is used to retrieve zero or more rows from one or more tables in a database. In most applications, SELECT is the most commonly used Data Manipulation Language command. In specifying a SELECT query, the user specifies a description of the desired result set, but they do *not* specify what physical operations must be executed to produce that result set. Translating the query into an efficient query plan is left to the database system, more specifically to the query optimizer. The statements immediately following "SELECT" specify which column(s) will be part of the result. An asterisk (" *") can also be used as a "wildcard" indicator to specify that all available columns are returned.

- Commonly available keywords related to SELECT include:

- FROM is used to indicate from which tables the data is to be taken, as well as how the tables JOIN to each other.

- WHERE is used to identify which rows are to be retrieved, or applied to GROUP BY. WHERE is evaluated before the GROUP BY. It can be viewed as a filtering mechanism to limit the result to a given criteria, usually via a Boolean expression.

- GROUP BY is used to combine rows with related values into elements of a smaller set of rows.

- HAVING is used to identify which of the "combined rows" (combined rows are produced when the query has a GROUP BY keyword or when the SELECT part contains aggregates), are to be retrieved.

HAVING acts much like a WHERE, but it operates on the results of the GROUP BY and hence can use aggregate functions.

- ORDER BY is used to identify which columns are used to sort the resulting data.

Data manipulation

First, there are the standard Data Manipulation Language (DML) elements. DML is the subset of the language used to add, update and delete data:

- INSERT is used to add zero or more rows (formally tuples) to an existing table.
- UPDATE is used to modify the values of a set of existing table rows.
- MERGE is used to combine the data of multiple tables. It is something of a combination of the INSERT and UPDATE elements. It is defined in the SQL:2003 standard; prior to that, some databases provided similar functionality via different syntax, sometimes called an “upsert”.
- DELETE removes zero or more existing rows from a table.

Transaction controls

Transactions, if available, can be used to wrap around the DML operations:

- BEGIN WORK (or START TRANSACTION, depending on SQL dialect) can be used to mark the start of a database transaction, which either completes completely or not at all.
- COMMIT causes all data changes in a transaction to be made permanent.
- ROLLBACK causes all data changes since the last COMMIT or ROLLBACK to be discarded, so that the state of the data is “rolled back” to the way it was prior to those changes being requested.

Data definition

The second group of keywords is the Data Definition Language (DDL). DDL allows the user to define new tables and associated elements. Most commercial SQL databases have proprietary extensions in their DDL, which allow control over nonstandard features of the database system. The most basic items of DDL are the CREATE, ALTER, RENAME, TRUNCATE and DROP commands:

- CREATE causes an object (a table, for example) to be created within the database.
- DROP causes an existing object within the database to be deleted, usually irretrievably.
- TRUNCATE deletes all data from a table (non-standard, but common SQL command).
- ALTER command permits the user to modify an existing object in various ways — for example, adding a column to an existing table.

Data control

The third group of SQL keywords is the Data Control Language (DCL). DCL handles the authorization aspects of data and permits the user to control who has access to see or manipulate data within the database. Its two main keywords are:

- GRANT — authorizes one or more users to perform an operation or a set of operations on an object.
- REVOKE — removes or restricts the capability of a user to perform an operation or a set of operations.