WEB DESIGNING BASICS

Note : Words in { } brackets are for information only.

1. What is a Program?

Ans – It's a set of Instruction.

2. What is a programming?

Ans – It's a process to create programming.

3. Types of Programming

Ans – 4 types

   a. Procedural programming. E.g. C language
   b. Object Oriented Programming (OOPs) e.g. C++ language
   c. Logical Programming e.g. Prolog programming
   d. Functional programming e.g. Python

4. What is WWW?

Ans – Full form is World Wide Web.

It's a collection of websites/web pages stored in web server.

Its connected to local computers through the internet.

Website contains text pages, digital images, audios, videos, etc.

5. What is internet ?

Ans – It's Global network which can be used to connect between the devices.

6. Why we required different domains/sub-domains?

Ans – There is change in taste, currency, weather, limitation of servers.

{To make website you have to purchase domain, e.g. .com, .co.in, .in }

7. Client -servers communication

Ans –

- Clients "sometimes ON"

➔ Initiates request to the server when initiated
➔ Doesn't communicate directly with other clients.
➔ Need to know the servers address
➔ E.g. Web browser on your laptop/cell phone.

- Serve is "always ON"

➔ Server requests many clients
➔ Doesn't initiate contact with the clients
➔ Needs a fixed, well-known address
➔ E.g. web server www.example.com

WEB DESIGNING BASICS

8. What is Protocols?

Ans – Network protocol is a group of rules accompanied by the network.

Formalised requirements & plans composed of rules, procedures and types that describe communication among a couple of devices over the network.

Described as an approach to rules the enable a couple of entities of a communication program.

{Knowledge : session, cookies, database, deployment [AWS/GOdaddy apps]}

9. What is application security ?

Ans – Refers to security precautions used at the application level to prevent the theft of hijacking of data or code within the application.

It includes security concerns made during application development and design as well as methods & procedures for protecting applications once they've been deployed.

Purpose : to improve security practices.

10. What is software application?

Ans – It's a group of instructions, it is methodology.

5 types of software application

1. Application software
2. System software
3. Driver software
4. Middleware software
5. Programming software

11. What is SLDC (Software Development Life Cycle)

It is a structured process that enables the production of high-quality, low-cost software, in the shortest possible production time.

The goal of the SDLC is to produce superior software that meets and exceeds all customer expectations and demands.

The SDLC defines and outlines a detailed plan with stages, or phases, that each encompass their own process and deliverables. Adherence to the SDLC enhances development speed and minimizes project risks and costs associated with alternative methods of production.

{ API – Application Programming Interface e.g. Uber, Ola, Zomato }

{ Companies : TCS, Wipro, Infosys, Mindtree, Accenture}

{Basic functionalities of website}

12. Steps of SDLC.

Ans –

WEB DESIGNING BASICS

Step 1 – Planning

Step 2 – Analysis

Step 3 – Design/ look (Front End)

Step 4 – Implementation

Step 5 – Testing

Step 6- Maintenance of Website

13. What is TCP and UPD?

Ans –

Sn	TCP	UDP
1	Its not secure	Its secure
2	Its in order	not in order
3	State memory	Stateless memory
4	Retransmission is possible	Retransmission is not possible
5	Its reliable	Its not reliable
6	Error free	error packet discarded
7	Relatively slow	Relatively fast
8	Flow control	No flow control
9	E.g. FTP (File Transfer Protocol)	E.g. Live Streaming
	Web Browsing	Online games
	E-mail	VoIP, DNS (Domain name server)

14. What is Architecture ?

Ans – It's Blue print of the software structure.

It shows overall structure of software, the collection of components in it and how they interact with one another while hiding the implementation.

15. Layer in software Architecture.

Ans – 5 layers

1. Presentation – Visuals upwards to downwards.
2. Application – Applications of Functionality
3. Business / Domain – Code necessary to connect the data
4. Persistence – necessary code to connect the data (data access layer)
5. Database – Work with database.

{ CRUD – Create, Read, Update, Delete }

16. Environments in Industry
    1. The analysis & Design Environment
    2. The development Environment
    3. The common build environment

4. The testing environment
5. The production environment

{Domain + hosting purchase should be done from one platform}

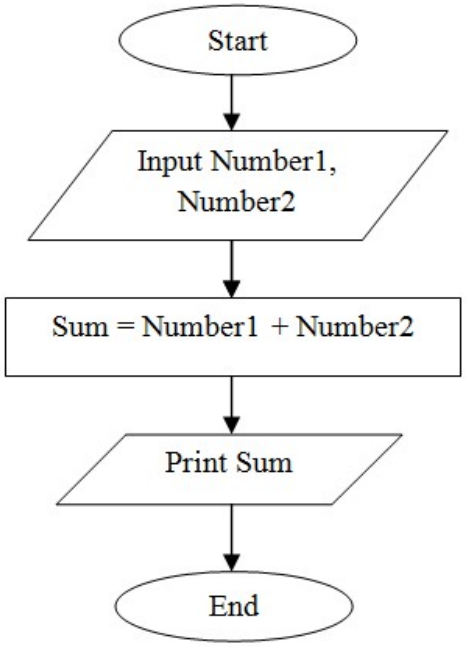
17. What is source code?

Ans – Source of Computer Program. It contains declarations, instructions, functions, loops and other statements, which act as instruction for the program on how to function. E.g. Text files, Hard disk.

18. IMP notes for source code
    - ➔ Coding must have comments
    - ➔ Coding must have a proper architecture
    - ➔ Coding must have a declaring coding versions and divide your code in packages.

19. What is Flow Chart?

Ans – Visual representation of an Algorithm using symbols and arrows to despite the steps and decision points in a process, which programming use to plan out the logic of their code before writing it, helping to visualize the flow of information & identify potential issues in the program's structure. E.g. Addition of any two numbers.



20. What are Flowchart symbols?

1. Oval - Start/End Symbol - 
2. Rectangle - Action or Process Symbol -  (single instruction)
3. Diamond - Decision Symbol - 
4. Parallelogram - Input/Output Symbol - 

WEB DESIGNING BASICS

21. Types of Flowchart

    Ans - Process flowchart

- Swimlane Flowchart
- Workflow Diagram
- Data Flow Diagram
- EPC Diagram
- SDL Diagram
- Process Map
- Process Flow Diagram

22. How to create a programming flowchart

Ans – 1. Decide the function or purpose.

2. Choose the steps and symbols.

3. Place the symbol & text.

4. Check the overall logic.

23. 5 rules of Drawing flowchart

Ans –

1. Consistency (lines, shapes, text)
2. Arrange content on a page
3. Build dataflow from left to right.
4. Set up split path instead of traditional symbols.
5. Place the return steam under the flowchart.

24. What is DFD?

Ans – It's a Data Flow Diagram.

To graphically represents the flow of the data in a business information system.

Describes the process that are involved in a system to transfer data from the input to the file storage and reports generation.

Two divisions → 1. Logical (to perform certain functionality) 2. Physical (Implementation if the logical data flow.

Two methods → 1. Yourdon & Coad 2. Gane & sarson

25. Why DFD?

Function or process which capture, manipulate, store and distribute data between components of a system and between a system & Its environment.

WEB DESIGNING BASICS

It's structure allows starting from a brand overview and expand it to a hierarchy of a detailed diagrams.

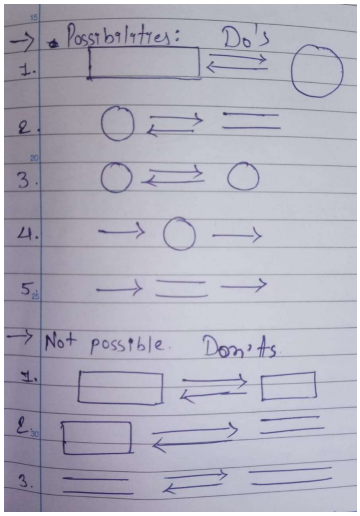
Three reasons:

1. Logical information flow of the system.
2. Determination of physical system construction requirements.
3. Simplicity of notation
4. Establishment of manual and automated system requirements.

26. DFD 4 symbols.

Ans – 1. External Entity , 2. Process, 3. Data store, 4. Data flow.

27. Dos and Don'ts of DFD



28. Use case Diagram

Ans - Unified Modeling Language. It models the behaviour of the system.

Used to illustrate the functional requirements of the system and its interaction with external agents.

It gives us high level view of the system without going into implementation details.

29. Components of Use case diagram

Ans – 4 components

1. Actors - any human or external system(e.g. organization)that interacts with the system. (Someone or something that uses our system to achieve goal)(primary actor- initiates the use of system [right side], secondary system – reactionary[left side])
2. Systems - main components of the system being modelled. (Whatever you are developing)

3. Relationships -how the actors and systems interact with each other.  (types – Association, Include [use dash line towards include use case], extend [extended to base use case dash line], generallization,
4. Use cases - specific tasks or goals that the system needs to be able to accomplish.  (represents an action that accomplishes some sort of task within the system)

---- END----