

## **SESSION 1 – ARCHITECTURE OF WEB**

(20 Easy + 20 Medium + 20 Hard MCQs)

→ Answer given just below each question

---

### **EASY LEVEL (1–20)**

**1. The Internet is best described as:**

- A) A single large computer
- B) A network of networks
- C) A collection of websites
- D) A database system

👉 Answer: B

---

**2. Which organization manages domain names globally?**

- A) W3C
- B) ICANN
- C) IEEE
- D) ISO

👉 Answer: B

---

**3. HTTP stands for:**

- A) HyperText Transfer Protocol
- B) High Transfer Text Protocol
- C) Hyper Tool Transfer Program
- D) Host Text Transfer Protocol

👉 Answer: A

---

**4. Which protocol is mainly used to access web pages?**

- A) FTP
- B) SMTP
- C) HTTP
- D) SNMP

👉 Answer: C

---

**5. HTTP belongs to which layer of TCP/IP?**

- A) Network
- B) Transport
- C) Application
- D) Data Link

👉 Answer: C

---

**6. DNS is used to:**

- A) Encrypt data
- B) Convert domain name to IP address
- C) Transfer files
- D) Authenticate users

👉 Answer: B

---

**7. Default port number of HTTP is:**

- A) 20
- B) 21
- C) 80
- D) 443

👉 Answer: C

---

**8. Default port number of HTTPS is:**

- A) 25
- B) 110
- C) 80
- D) 443

👉 Answer: D

---

**9. Which HTTP method is used to retrieve data?**

- A) POST
- B) PUT
- C) GET
- D) DELETE

👉 Answer: C

---

**10. Which HTTP method is used to submit form data?**

- A) GET
- B) POST
- C) HEAD
- D) TRACE

👉 Answer: B

---

**11. HTTPS provides:**

- A) Compression
- B) Encryption
- C) Caching
- D) Redirection

👉 Answer: B

---

**12. Which web server is developed by Microsoft?**

- A) Apache
- B) Nginx
- C) IIS
- D) Tomcat

👉 Answer: C

---

**13. Apache web server is:**

- A) Proprietary
- B) Open source
- C) Paid only
- D) Windows-only

👉 Answer: B

---

**14. Domain name is:**

- A) Machine-readable address
- B) Human-readable address
- C) MAC address
- D) Port number

👉 Answer: B

---

**15. Which is a valid domain name?**

- A) www.google
- B) google.com
- C) google@com
- D) http/google

👉 Answer: B

---

**16. HTTP is a:**

- A) Stateful protocol
- B) Stateless protocol
- C) Secure protocol

D) Connection-oriented protocol

👉 Answer: B

---

**17. Which HTTP version came first?**

- A) HTTP/2
- B) HTTP/1.1
- C) HTTP/1.0
- D) HTTPS

👉 Answer: C

---

**18. Which HTTP method deletes a resource?**

- A) PUT
- B) POST
- C) DELETE
- D) HEAD

👉 Answer: C

---

**19. Website files are stored on:**

- A) Browser
- B) Client
- C) Web server
- D) Router

👉 Answer: C

---

**20. IIS primarily runs on:**

- A) Linux
- B) Windows
- C) macOS
- D) Android

👉 Answer: B

---

## 🟡 MEDIUM LEVEL (21–40)

**21. DNS mainly uses which protocol?**

- A) TCP
- B) UDP
- C) HTTP
- D) FTP

👉 Answer: B

---

**22. Which HTTP method is idempotent?**

- A) POST
- B) GET
- C) CONNECT
- D) PATCH

👉 Answer: B

---

**23. HTTP status code 200 indicates:**

- A) Redirect
- B) Client error
- C) Success
- D) Server error

👉 Answer: C

---

**24. HTTP status code 404 means:**

- A) Forbidden
- B) Bad request
- C) Not found
- D) Server error

👉 Answer: C

---

**25. HTTP status code 500 indicates:**

- A) Client error
- B) Server error
- C) Redirect
- D) Authentication error

👉 Answer: B

---

**26. Which HTTP method fetches only headers?**

- A) GET
- B) POST
- C) HEAD
- D) OPTIONS

👉 Answer: C

---

**27. Which HTTP version introduced persistent connections by default?**

- A) HTTP/1.0
- B) HTTP/1.1
- C) HTTP/2

D) HTTPS

👉 Answer: B

---

**28. Stateless protocol means:**

- A) Server stores client info
- B) No session maintained by server
- C) Connection is encrypted
- D) Server crashes often

👉 Answer: B

---

**29. HTTPS uses which protocol for security?**

- A) SSL/TLS
- B) FTP
- C) UDP
- D) SSH

👉 Answer: A

---

**30. Which status code represents redirection?**

- A) 2xx
- B) 3xx
- C) 4xx
- D) 5xx

👉 Answer: B

---

**31. GET request data is sent via:**

- A) Request body
- B) URL
- C) Cookies
- D) Headers only

👉 Answer: B

---

**32. POST request data is sent via:**

- A) URL
- B) Header only
- C) Request body
- D) Status code

👉 Answer: C

---

**33. Which server is open-source and cross-platform?**

- A) IIS
- B) Apache
- C) WebLogic
- D) GlassFish

👉 Answer: B

---

**34. Which HTTP method updates an existing resource?**

- A) GET
- B) POST
- C) PUT
- D) HEAD

👉 Answer: C

---

**35. DNS translates:**

- A) IP to MAC
- B) MAC to IP
- C) Domain to IP
- D) Port to IP

👉 Answer: C

---

**36. Which protocol ensures data integrity and confidentiality?**

- A) HTTP
- B) FTP
- C) HTTPS
- D) SMTP

👉 Answer: C

---

**37. Which status code indicates unauthorized access?**

- A) 200
- B) 301
- C) 401
- D) 500

👉 Answer: C

---

**38. Which component handles client requests?**

- A) Browser
- B) Web server
- C) DNS
- D) Router

👉 Answer: B

---

**39. HTTP/2 mainly improves:**

- A) Security only
- B) Compression only
- C) Performance
- D) URL handling

👉 Answer: C

---

**40. Which HTTP method is NOT safe?**

- A) GET
- B) HEAD
- C) POST
- D) OPTIONS

👉 Answer: C

---

## 🔴 HARD LEVEL (41–60)

**41. Why is HTTP stateless?**

- A) For security
- B) For performance
- C) Server does not store client context
- D) Uses UDP

👉 Answer: C

---

**42. HTTP sessions are maintained using:**

- A) DNS
- B) Cookies
- C) IP only
- D) Ports

👉 Answer: B

---

**43. Which feature was added in HTTP/2?**

- A) Text-based protocol
- B) Multiple TCP connections
- C) Multiplexing
- D) No headers

👉 Answer: C

---

**44. HTTP/1.0 drawback was:**

- A) Encryption
- B) Statelessness
- C) New connection per request
- D) DNS dependency

👉 Answer: C

---

**45. Which status code indicates permanent redirection?**

- A) 302
- B) 301
- C) 307
- D) 200

👉 Answer: B

---

**46. HTTPS prevents:**

- A) Packet loss
- B) SQL injection
- C) Man-in-the-middle attack
- D) Server crash

👉 Answer: C

---

**47. Which HTTP method is both safe and idempotent?**

- A) POST
- B) PUT
- C) GET
- D) DELETE

👉 Answer: C

---

**48. DNS primarily works at which layer?**

- A) Transport
- B) Network
- C) Application
- D) Session

👉 Answer: C

---

**49. IIS stands for:**

- A) Internet Information Services
- B) Internal Internet System
- C) Integrated Internet Server
- D) Internet Interface Software

👉 Answer: A

---

**50. Which HTTP response class indicates client-side error?**

- A) 1xx
- B) 2xx
- C) 4xx
- D) 5xx

👉 Answer: C

---

**51. HTTP/2 reduces latency by:**

- A) Compression only
- B) Caching only
- C) Multiplexing & header compression
- D) Statelessness

👉 Answer: C

---

**52. Which protocol resolves domain name before HTTP request?**

- A) SMTP
- B) DNS
- C) FTP
- D) ARP

👉 Answer: B

---

**53. What happens if DNS fails?**

- A) HTTP still works
- B) Website loads slower
- C) Website cannot be accessed
- D) Only images fail

👉 Answer: C

---

**54. Which HTTP method should NOT change server state?**

- A) POST
- B) PUT
- C) GET
- D) DELETE

👉 Answer: C

---

**55. HTTPS uses certificates issued by:**

- A) ISP
- B) DNS server
- C) Certificate Authority

D) Web server

👉 Answer: C

---

**56. Which is true about HTTP/2?**

- A) Uses plain text
- B) Uses binary framing
- C) Uses UDP
- D) Replaces TCP

👉 Answer: B

---

**57. Web architecture follows which model?**

- A) Peer-to-peer
- B) Client-server
- C) Ring
- D) Mesh

👉 Answer: B

---

**58. Which server is best suited for Windows-based hosting?**

- A) Apache
- B) Nginx
- C) IIS
- D) Lighttpd

👉 Answer: C

---

**59. HTTP status code 403 means:**

- A) Not found
- B) Unauthorized
- C) Forbidden
- D) Server error

👉 Answer: C

---

**60. HTTP session tracking is required because HTTP is:**

- A) Secure
- B) Fast
- C) Stateless
- D) Text-based

👉 Answer: C



# SESSION 2 – HTML & HTML5 (WPT – CCEE)

---



## EASY LEVEL (1–20)

**1. HTML stands for:**

- A) High Text Machine Language
- B) Hyper Text Markup Language
- C) Hyper Tool Multi Language
- D) Home Tool Markup Language

👉 Answer: B

---

**2. HTML5 is:**

- A) Programming language
- B) Markup language
- C) Scripting language
- D) Query language

👉 Answer: B

---

**3. Which tag is used for the largest heading?**

- A) `<heading>`
- B) `<h6>`
- C) `<h1>`
- D) `<head>`

👉 Answer: C

---

**4. Which tag is used to create a hyperlink?**

- A) `<link>`
- B) `<a>`
- C) `<href>`
- D) `<url>`

👉 Answer: B

---

**5. Which tag is used to display an image?**

- A) `<img>`
- B) `<image>`
- C) `<src>`
- D) `<pic>`

👉 Answer: A

---

**6. Which attribute specifies image location?**

- A) href
- B) alt
- C) src
- D) path

👉 Answer: C

---

**7. Which tag is used for paragraph?**

- A) <para>
- B) <pg>
- C) <p>
- D) <paragraph>

👉 Answer: C

---

**8. Which list displays items with bullets?**

- A) Ordered list
- B) Unordered list
- C) Definition list
- D) Table list

👉 Answer: B

---

**9. Which tag creates an ordered list?**

- A) <ul>
- B) <ol>
- C) <li>
- D) <list>

👉 Answer: B

---

**10. Which tag is used for table rows?**

- A) <td>
- B) <th>
- C) <tr>
- D) <table>

👉 Answer: C

---

**11. Which tag embeds another webpage?**

- A) <frame>

B) <iframe>

C) <embed>

D) <object>

👉 Answer: B

---

**12. HTML files are saved with extension:**

A) .ht

B) .xml

C) .html

D) .css

👉 Answer: C

---

**13. Which tag defines a form?**

A) <input>

B) <form>

C) <fieldset>

D) <control>

👉 Answer: B

---

**14. Which input type is used for password?**

A) text

B) hidden

C) password

D) secure

👉 Answer: C

---

**15. Radio buttons allow:**

A) Multiple selection

B) Single selection

C) Text input

D) File upload

👉 Answer: B

---

**16. Checkbox allows:**

A) Only one option

B) Only text

C) Multiple selections

D) Hidden input

👉 Answer: C

---

**17. Which tag is used for dropdown list?**

- A) <option>
- B) <select>
- C) <dropdown>
- D) <list>

👉 Answer: B

---

**18. HTML validation checks:**

- A) Browser compatibility
- B) Code correctness
- C) Internet speed
- D) Server response

👉 Answer: B

---

**19. Which tag plays audio in HTML5?**

- A) <sound>
- B) <music>
- C) <audio>
- D) <media>

👉 Answer: C

---

**20. Which tag plays video in HTML5?**

- A) <movie>
- B) <media>
- C) <video>
- D) <film>

👉 Answer: C

---

## 🟡 MEDIUM LEVEL (21–40)

**21. Which element is new in HTML5?**

- A) <font>
- B) <center>
- C) <section>
- D) <big>

👉 Answer: C

---

**22. Which HTML5 tag is used for navigation links?**

- A) <nav>
- B) <menu>
- C) <links>
- D) <header>

👉 Answer: A

---

**23. Which tag defines footer content?**

- A) <bottom>
- B) <footer>
- C) <end>
- D) <section>

👉 Answer: B

---

**24. Which attribute enables form validation automatically?**

- A) validate
- B) required
- C) pattern
- D) check

👉 Answer: B

---

**25. Which input type is used for email validation?**

- A) text
- B) mail
- C) email
- D) validate

👉 Answer: C

---

**26. Which HTML5 attribute provides hint text?**

- A) value
- B) hint
- C) placeholder
- D) title

👉 Answer: C

---

**27. Which element is used for drawing graphics?**

- A) <svg>
- B) <canvas>
- C) <graphic>

D) <draw>

👉 Answer: B

---

**28. Which API provides location of user?**

- A) Media API
- B) Storage API
- C) Geo-location API
- D) Canvas API

👉 Answer: C

---

**29. DOM stands for:**

- A) Document Object Model
- B) Data Object Model
- C) Document Oriented Markup
- D) Digital Object Model

👉 Answer: A

---

**30. DOM represents HTML document as:**

- A) Linear text
- B) Tree structure
- C) Table
- D) Graph

👉 Answer: B

---

**31. Which tag is semantic?**

- A) <div>
- B) <span>
- C) <article>
- D) <font>

👉 Answer: C

---

**32. Which attribute links label to input?**

- A) id
- B) name
- C) for
- D) class

👉 Answer: C

---

**33. Which tag groups form elements?**

- A) <group>
- B) <fieldset>
- C) <formgroup>
- D) <legend>

👉 Answer: B

---

**34. Which tag provides caption for fieldset?**

- A) <title>
- B) <legend>
- C) <label>
- D) <caption>

👉 Answer: B

---

**35. HTML5 audio supports:**

- A) MP3 only
- B) WAV only
- C) Multiple formats
- D) No formats

👉 Answer: C

---

**36. Which attribute auto-plays video?**

- A) play
- B) start
- C) autoplay
- D) loop

👉 Answer: C

---

**37. Which HTML tag is deprecated in HTML5?**

- A) <strong>
- B) <em>
- C) <font>
- D) <section>

👉 Answer: C

---

**38. Which tag is non-semantic?**

- A) <article>
- B) <section>
- C) <aside>

D) <div>

👉 Answer: D

---

**39. HTML5 supports drag and drop using:**

- A) Java
- B) CSS
- C) JavaScript API
- D) PHP

👉 Answer: C

---

**40. DOM allows:**

- A) Only reading HTML
- B) Only writing HTML
- C) Dynamic modification
- D) Server-side scripting

👉 Answer: C

---

## 🔴 HARD LEVEL (41–60)

**41. Which HTML5 feature improves accessibility?**

- A) Canvas
- B) ARIA
- C) Video
- D) Audio

👉 Answer: B

---

**42. Microdata is used for:**

- A) Styling
- B) SEO & structured data
- C) Security
- D) Validation

👉 Answer: B

---

**43. Which tag represents independent content?**

- A) <section>
- B) <div>
- C) <article>
- D) <aside>

👉 Answer: C

---

**44. Which attribute disables form validation?**

- A) novalidate
- B) disable
- C) off
- D) ignore

👉 Answer: A

---

**45. Which input type shows date picker?**

- A) calendar
- B) date
- C) time
- D) datetime

👉 Answer: B

---

**46. Which attribute specifies multiple file upload?**

- A) many
- B) multi
- C) multiple
- D) files

👉 Answer: C

---

**47. Which tag embeds multimedia with fallback?**

- A) <iframe>
- B) <embed>
- C) <object>
- D) <media>

👉 Answer: C

---

**48. Canvas graphics are:**

- A) Vector-based
- B) Raster-based
- C) Text-based
- D) CSS-based

👉 Answer: B

---

**49. SVG differs from canvas because SVG is:**

- A) Raster
- B) Pixel-based

C) Vector-based

D) Deprecated

👉 Answer: C

---

**50. Which HTML5 API allows offline storage?**

A) Cache API

B) Web Storage

C) Session API

D) DOM API

👉 Answer: B

---

**51. DOM manipulation is mostly done using:**

A) HTML

B) CSS

C) JavaScript

D) SQL

👉 Answer: C

---

**52. Which attribute is mandatory for radio button grouping?**

A) id

B) value

C) name

D) class

👉 Answer: C

---

**53. Which tag improves SEO most?**

A) <div>

B) <span>

C) Semantic tags

D) <font>

👉 Answer: C

---

**54. Which HTML5 feature removed need of Flash?**

A) CSS3

B) Audio & Video tags

C) Canvas

D) SVG

👉 Answer: B

---

**55. Which event fires when form is submitted?**

- A) onclick
- B) onsubmit
- C) onload
- D) onchange

👉 Answer: B

---

**56. Which attribute opens link in new tab?**

- A) new
- B) blank
- C) target="\_blank"
- D) window

👉 Answer: C

---

**57. DOM treats attributes as:**

- A) Elements
- B) Objects
- C) Nodes
- D) Methods

👉 Answer: C

---

**58. Which tag defines sidebar content?**

- A) `<aside>`
- B) `<nav>`
- C) `<footer>`
- D) `<section>`

👉 Answer: A

---

**59. HTML5 validation is done using:**

- A) JavaScript only
- B) CSS
- C) Built-in browser rules
- D) Server only

👉 Answer: C

---

**60. DOM Level 0 refers to:**

- A) CSS
- B) HTML specification
- C) Original browser DOM

D) XML

👉 Answer: C

## SESSION 3 – CASCADING STYLE SHEETS (CSS)

---

### EASY LEVEL (1–20)

1. CSS stands for:

- A) Color Style Sheets
- B) Cascading Style Sheets
- C) Creative Style Sheets
- D) Computer Style Sheets

👉 Answer: B

---

2. CSS is used to:

- A) Create logic
- B) Structure web pages
- C) Style web pages
- D) Store data

👉 Answer: C

---

3. Which CSS type is written inside HTML tag?

- A) External
- B) Internal
- C) Inline
- D) Embedded

👉 Answer: C

---

4. Which CSS is written inside `<style>` tag?

- A) Inline
- B) External
- C) Internal
- D) Browser

👉 Answer: C

---

**5. External CSS is stored in:**

- A) .html file
- B) .js file
- C) .css file
- D) .xml file

👉 Answer: C

---

**6. Which tag is used to link external CSS?**

- A) <css>
- B) <script>
- C) <link>
- D) <style>

👉 Answer: C

---

**7. Which attribute is used to apply unique style?**

- A) class
- B) id
- C) name
- D) style

👉 Answer: B

---

**8. Which attribute can be reused multiple times?**

- A) id
- B) class
- C) style
- D) name

👉 Answer: B

---

**9. Which CSS property sets text color?**

- A) text-color
- B) font-color
- C) color
- D) fg-color

👉 Answer: C

---

**10. Which CSS property sets background color?**

- A) bgcolor
- B) background-color

C) back-color

D) color

👉 Answer: B

---

**11. Which CSS property controls font size?**

A) text-size

B) font-style

C) font-size

D) size

👉 Answer: C

---

**12. Which unit is relative in CSS?**

A) px

B) cm

C) em

D) pt

👉 Answer: C

---

**13. CSS box model includes:**

A) margin, border, padding, content

B) padding, margin, color, font

C) border, background, color, font

D) content only

👉 Answer: A

---

**14. Which part of box model is outermost?**

A) Content

B) Padding

C) Border

D) Margin

👉 Answer: D

---

**15. Which property sets space inside border?**

A) margin

B) padding

C) border

D) content

👉 Answer: B

---

**16. Which property sets space outside border?**

- A) padding
- B) border
- C) margin
- D) outline

👉 Answer: C

---

**17. Which CSS property sets font family?**

- A) font-style
- B) font-type
- C) font-family
- D) text-style

👉 Answer: C

---

**18. Which CSS selector selects an id?**

- A) .id
- B) #id
- C) \*id
- D) @id

👉 Answer: B

---

**19. Which selector selects all elements?**

- A) #
- B) .
- C) \*
- D) all

👉 Answer: C

---

**20. Which style has highest priority?**

- A) External
- B) Internal
- C) Inline
- D) Browser default

👉 Answer: C

---

## 🟡 MEDIUM LEVEL (21–40)

**21. CSS was developed by:**

- A) Microsoft
- B) W3C

- C) Google
- D) Netscape

👉 Answer: B

---

**22. Which CSS type is best for large websites?**

- A) Inline
- B) Internal
- C) External
- D) Embedded

👉 Answer: C

---

**23. Which property makes text bold?**

- A) font-weight
- B) text-style
- C) font-bold
- D) bold

👉 Answer: A

---

**24. Which CSS rule controls multiple styles?**

- A) Multiple style sheets
- B) Cascading
- C) Inline rules
- D) Browser rules

👉 Answer: B

---

**25. Which property aligns text horizontally?**

- A) vertical-align
- B) align
- C) text-align
- D) position

👉 Answer: C

---

**26. Which CSS font style makes text italic?**

- A) font-weight
- B) font-style
- C) font-variant
- D) text-style

👉 Answer: B

---

**27. Which CSS property removes underline from link?**

- A) link-style
- B) text-decoration
- C) font-decoration
- D) remove-line

👉 Answer: B

---

**28. Which CSS selector selects class `box`?**

- A) #box
- B) .box
- C) box
- D) \*box

👉 Answer: B

---

**29. Which property sets border thickness?**

- A) border-color
- B) border-style
- C) border-width
- D) border-size

👉 Answer: C

---

**30. Which property is shorthand for border?**

- A) border-all
- B) border
- C) outline
- D) frame

👉 Answer: B

---

**31. Which CSS unit is relative to viewport width?**

- A) px
- B) em
- C) vw
- D) pt

👉 Answer: C

---

**32. Which HTML tag holds internal CSS?**

- A) <css>
- B) <script>
- C) <style>

D) <link>

👉 Answer: C

---

**33. Multiple classes can be applied using:**

- A) commas
- B) space separated values
- C) plus sign
- D) dot

👉 Answer: B

---

**34. Which selector selects all <p> tags?**

- A) #p
- B) .p
- C) p
- D) \*p

👉 Answer: C

---

**35. Which property changes mouse cursor?**

- A) pointer
- B) cursor
- C) mouse
- D) hover

👉 Answer: B

---

**36. Which CSS property makes element invisible but occupies space?**

- A) display:none
- B) visibility:hidden
- C) opacity:0
- D) hidden

👉 Answer: B

---

**37. Which CSS property controls element width?**

- A) size
- B) width
- C) box
- D) dimension

👉 Answer: B

---

**38. Which CSS property controls height?**

- A) size
- B) depth
- C) height
- D) box-height

👉 Answer: C

---

**39. Which CSS property defines background image?**

- A) bg-image
- B) image
- C) background-image
- D) image-background

👉 Answer: C

---

**40. Inline CSS uses which attribute?**

- A) class
- B) id
- C) style
- D) css

👉 Answer: C

---

## HARD LEVEL (41–60)

**41. CSS specificity order (highest to lowest):**

- A) class > id > element
- B) element > class > id
- C) id > class > element
- D) inline > class > id

👉 Answer: C

---

**42. If inline and external styles conflict, which applies?**

- A) External
- B) Browser
- C) Inline
- D) Internal

👉 Answer: C

---

**43. \* { margin:0; } refers to:**

- A) id selector
- B) class selector

C) universal selector

D) group selector

👉 Answer: C

---

**44. Which CSS property affects box size calculation?**

A) display

B) position

C) box-sizing

D) float

👉 Answer: C

---

**45. `box-sizing: border-box;` means:**

A) Border excluded from width

B) Padding excluded from width

C) Border and padding included in width

D) Only content counted

👉 Answer: C

---

**46. Which property sets font thickness?**

A) font-style

B) font-weight

C) text-weight

D) weight

👉 Answer: B

---

**47. Which property controls spacing between letters?**

A) word-spacing

B) line-height

C) letter-spacing

D) text-indent

👉 Answer: C

---

**48. Which property controls spacing between lines?**

A) letter-spacing

B) word-spacing

C) line-height

D) margin

👉 Answer: C

---

**49. Which CSS rule is applied last?**

- A) Browser default
- B) External
- C) Internal
- D) Inline

👉 Answer: D

---

**50. Which selector has highest specificity?**

- A) `.box`
- B) `div`
- C) `#box`
- D) `*`

👉 Answer: C

---

**51. Which CSS property controls element visibility?**

- A) `display`
- B) `opacity`
- C) `visibility`
- D) `show`

👉 Answer: C

---

**52. CSS fonts are used to:**

- A) Change page layout
- B) Improve readability
- C) Add logic
- D) Store data

👉 Answer: B

---

**53. Which CSS property centers text horizontally?**

- A) `center`
- B) `text-align`
- C) `align`
- D) `position`

👉 Answer: B

---

**54. Which property prevents content overflow?**

- A) `overflow`
- B) `float`
- C) `clear`

D) wrap

👉 Answer: A

---

**55. Which CSS selector selects elements inside another element?**

- A) Adjacent selector
- B) Descendant selector
- C) Child selector
- D) Universal selector

👉 Answer: B

---

**56. Which syntax links external CSS?**

- A) `<style src="style.css">`
- B) `<link rel="stylesheet" href="style.css">`
- C) `<css href="style.css">`
- D) `<script src="style.css">`

👉 Answer: B

---

**57. Which CSS property removes bullets from list?**

- A) list-style
- B) list-style-type
- C) bullet-style
- D) text-decoration

👉 Answer: B

---

**58. Which CSS property sets page layout spacing?**

- A) padding
- B) margin
- C) border
- D) outline

👉 Answer: B

---

**59. Which CSS property controls text capitalization?**

- A) text-style
- B) text-transform
- C) font-style
- D) capitalize

👉 Answer: B

---

**60. CSS cascade means:**

- A) Random rule application
- B) Rules applied by priority
- C) Only inline rules
- D) Only external rules

👉 **Answer: B**

---

## SESSION 4 – RESPONSIVE WEB DESIGN & BOOTSTRAP

---

### EASY LEVEL (1–20)

**1. Responsive Web Design means:**

- A) Same design for desktop only
- B) Fixed-width web pages
- C) Web pages adapt to different screen sizes
- D) Faster loading pages

👉 **Answer: C**

---

**2. UI scripting mainly deals with:**

- A) Server-side logic
- B) Database handling
- C) User interface behavior
- D) Network protocols

👉 **Answer: C**

---

**3. Best user experience should support:**

- A) Desktop only
- B) Mobile only
- C) Tablet only
- D) Desktop, Tablet, and Mobile

👉 **Answer: D**

---

**4. Bootstrap is:**

- A) Programming language
- B) Database

C) Front-end framework

D) Web server

👉 Answer: C

---

**5. Bootstrap is mainly used for:**

A) Backend development

B) Responsive UI design

C) Database design

D) Security

👉 Answer: B

---

**6. Bootstrap is written in:**

A) Java only

B) HTML & Java

C) HTML, CSS, and JavaScript

D) Python

👉 Answer: C

---

**7. Bootstrap grid system is based on:**

A) 10 columns

B) 8 columns

C) 12 columns

D) 16 columns

👉 Answer: C

---

**8. Which class defines a row in Bootstrap?**

A) `.grid`

B) `.row`

C) `.column`

D) `.container`

👉 Answer: B

---

**9. Which class creates a fixed-width container?**

A) `.container-fluid`

B) `.wrapper`

C) `.container`

D) `.box`

👉 Answer: C

---

**10. Which class creates full-width container?**

- A) `.container`
- B) `.row`
- C) `.container-fluid`
- D) `.fluid`

👉 Answer: C

---

**11. Bootstrap typography mainly controls:**

- A) Images
- B) Tables
- C) Text styles
- D) Forms

👉 Answer: C

---

**12. Which component displays warning messages?**

- A) Badge
- B) Alert
- C) Panel
- D) Well

👉 Answer: B

---

**13. Which Bootstrap component is used for navigation bar?**

- A) Tabs
- B) Pills
- C) Navbar
- D) Menu

👉 Answer: C

---

**14. Which component shows progress visually?**

- A) Badge
- B) Alert
- C) Progress bar
- D) Label

👉 Answer: C

---

**15. Which component groups buttons together?**

- A) Button toolbar
- B) Button group
- C) Panel

D) Badge

👉 Answer: B

---

**16. Which Bootstrap component displays small count values?**

- A) Panel
- B) Badge
- C) Progress bar
- D) Alert

👉 Answer: B

---

**17. Which component collapses content?**

- A) Tabs
- B) Dropdown
- C) Collapse
- D) Navbar

👉 Answer: C

---

**18. Which component is used for dropdown menus?**

- A) Collapse
- B) Dropdown
- C) Navbar
- D) List group

👉 Answer: B

---

**19. Which Bootstrap feature ensures consistency in design?**

- A) JavaScript
- B) Templates & themes
- C) Database
- D) Routing

👉 Answer: B

---

**20. Bootstrap forms help in:**

- A) Server validation
- B) UI styling of inputs
- C) Database connection
- D) Security

👉 Answer: B

---

## MEDIUM LEVEL (21–40)

21. Responsive design uses which CSS concept heavily?

- A) Floats
- B) Media queries
- C) Tables
- D) Frames

👉 Answer: B

---

22. Which Bootstrap class is used for small devices?

- A) .col-lg-\*
- B) .col-md-\*
- C) .col-sm-\*
- D) .col-xl-\*

👉 Answer: C

---

23. Which class is used for extra small devices?

- A) .col-xs-\*
- B) .col-sm-\*
- C) .col-md-\*
- D) .col-lg-\*

👉 Answer: A

---

24. Bootstrap grid classes are based on:

- A) Device width
- B) Device height
- C) Screen resolution
- D) Browser type

👉 Answer: A

---

25. .col-md-6 means:

- A) 6 rows
- B) 6 columns out of 12 on medium devices
- C) 6px width
- D) 6 containers

👉 Answer: B

---

26. Which component replaces HTML tables with styling?

- A) Panels

- A) Styled tables
- B) Grid system
- C) Forms

👉 Answer: B

---

**27. Which component displays list of items vertically?**

- A) Panels
- B) List groups
- C) Badges
- D) Pagination

👉 Answer: B

---

**28. Which Bootstrap class styles images responsively?**

- A) `.img-style`
- B) `.img-fluid`
- C) `.img-responsive`
- D) `.image-auto`

👉 Answer: B

---

**29. Jumbotron is used for:**

- A) Navigation
- B) Highlighting important content
- C) Footer
- D) Sidebar

👉 Answer: B

---

**30. Which component breaks content into pages?**

- A) Pagination
- B) Tabs
- C) Pills
- D) Navbar

👉 Answer: A

---

**31. Which component creates tabbed navigation?**

- A) Alerts
- B) Panels
- C) Tabs
- D) Collapse

👉 Answer: C

---

**32. Pills differ from tabs mainly by:**

- A) Functionality
- B) Shape and appearance
- C) JavaScript usage
- D) Grid size

👉 **Answer: B**

---

**33. Which component is used to show grouped content blocks?**

- A) Panels
- B) Alerts
- C) Badges
- D) Buttons

👉 **Answer: A**

---

**34. Bootstrap inputs include:**

- A) Text, password, checkbox
- B) Only text
- C) Only buttons
- D) Only dropdowns

👉 **Answer: A**

---

**35. Which class styles buttons in Bootstrap?**

- A) `.btn`
- B) `.button`
- C) `.click`
- D) `.press`

👉 **Answer: A**

---

**36. Which class makes button blue by default?**

- A) `.btn-success`
- B) `.btn-danger`
- C) `.btn-primary`
- D) `.btn-info`

👉 **Answer: C**

---

**37. Which Bootstrap feature reduces custom CSS need?**

- A) JavaScript
- B) Predefined components

C) HTML tags

D) Database

👉 Answer: B

---

### 38. Which layout approach does Bootstrap follow?

A) Fixed layout

B) Fluid layout

C) Both fixed and fluid

D) Absolute layout

👉 Answer: C

---

### 39. Bootstrap themes mainly change:

A) Backend logic

B) Database schema

C) Look and feel

D) Network speed

👉 Answer: C

---

### 40. Bootstrap templates are:

A) Server scripts

B) Ready-made page layouts

C) APIs

D) Databases

👉 Answer: B

---

## 🔴 HARD LEVEL (41–60)

### 41. Why is Bootstrap preferred for responsive design?

A) It uses Java

B) Built-in grid & components

C) Faster servers

D) Database support

👉 Answer: B

---

### 42. Bootstrap grid rows must be placed inside:

A) `.col`

B) `.container` or `.container-fluid`

C) `.panel`

D) `.body`

👉 Answer: B

---

**43. Total columns in a Bootstrap row should not exceed:**

- A) 10
- B) 11
- C) 12
- D) 16

👉 Answer: C

---

**44. What happens if columns exceed 12?**

- A) Error
- B) Columns overflow
- C) Columns wrap to next line
- D) Browser crashes

👉 Answer: C

---

**45. Which Bootstrap class hides content on small screens?**

- A) `.hidden-sm`
- B) `.hide-small`
- C) `.no-sm`
- D) `.invisible-sm`

👉 Answer: A

---

**46. Which component uses JavaScript dependency?**

- A) Grid
- B) Typography
- C) Dropdown
- D) Containers

👉 Answer: C

---

**47. Collapse component is useful for:**

- A) Fixed content
- B) Mobile navigation
- C) Tables
- D) Images

👉 Answer: B

---

**48. Navbar becomes responsive using:**

- A) Grid system
- B) Collapse plugin

C) Pagination

D) Panels

👉 Answer: B

---

**49. Which feature ensures same UI across browsers?**

A) Media queries

B) Bootstrap CSS reset

C) JavaScript

D) DOM

👉 Answer: B

---

**50. Bootstrap form layout is based on:**

A) Tables

B) Grid system

C) Frames

D) Flexbox only

👉 Answer: B

---

**51. Which Bootstrap component shows status or count?**

A) Panel

B) Badge

C) Alert

D) Jumbotron

👉 Answer: B

---

**52. Progress bar width is controlled by:**

A) Java only

B) CSS width

C) HTML height

D) Grid columns

👉 Answer: B

---

**53. Dropdown menus require:**

A) CSS only

B) JavaScript

C) HTML only

D) Server code

👉 Answer: B

---

**54. Bootstrap supports mobile-first design means:**

- A) Desktop first
- B) Tablet first
- C) Mobile optimized first
- D) Server optimized

👉 **Answer: C**

---

**55. Which class makes table responsive?**

- A) `.table`
- B) `.table-responsive`
- C) `.responsive-table`
- D) `.table-fluid`

👉 **Answer: B**

---

**56. Which component organizes links vertically?**

- A) Tabs
- B) Pills
- C) List groups
- D) Panels

👉 **Answer: C**

---

**57. Bootstrap reduces development time because:**

- A) No CSS needed
- B) Ready-made components
- C) No HTML needed
- D) No JS needed

👉 **Answer: B**

---

**58. Which Bootstrap utility controls spacing?**

- A) Color utilities
- B) Spacing utilities (m-, p-)
- C) Display utilities
- D) Text utilities

👉 **Answer: B**

---

**59. Which Bootstrap feature helps reuse designs?**

- A) Grid
- B) Themes
- C) Alerts

D) Badges

👉 Answer: B

---

**60. Responsive design improves:**

- A) SEO only
- B) User experience across devices
- C) Database performance
- D) Server security

👉 Answer: B

---

## SESSION 5 – JAVASCRIPT (WPT – CCEE)

---

### EASY LEVEL (1–20)

**1. JavaScript is a:**

- A) Compiled language
- B) Markup language
- C) Scripting language
- D) Assembly language

👉 Answer: C

---

**2. JavaScript runs mainly on:**

- A) Server only
- B) Browser only
- C) Both client and server
- D) Database

👉 Answer: C

---

**3. Which keyword is used to declare variable (ES6)?**

- A) var
- B) let
- C) const
- D) All of the above

👉 Answer: D

---

**4. JavaScript is:**

- A) Case-insensitive
- B) Case-sensitive
- C) Not sensitive
- D) Only lowercase

👉 Answer: B

---

**5. Single-line comment in JavaScript is:**

- A) `<!-- -->`
- B) `/* */`
- C) `//`
- D) `#`

👉 Answer: C

---

**6. Which operator is used for assignment?**

- A) `==`
- B) `====`
- C) `=`
- D) `:=`

👉 Answer: C

---

**7. Which data type is NOT primitive?**

- A) Number
- B) String
- C) Boolean
- D) Array

👉 Answer: D

---

**8. Which keyword defines constant variable?**

- A) `let`
- B) `var`
- C) `const`
- D) `static`

👉 Answer: C

---

**9. JavaScript arrays are:**

- A) Fixed size
- B) Homogeneous
- C) Dynamic

D) Two-dimensional only

👉 Answer: C

---

**10. Index of first element in array is:**

- A) 0
- B) 1
- C) -1
- D) Depends

👉 Answer: A

---

**11. Which function converts string to number?**

- A) `toString()`
- B) `parseInt()`
- C) `stringify()`
- D) `NumberToString()`

👉 Answer: B

---

**12. Which Boolean value represents false?**

- A) 0
- B) `false`
- C) `null`
- D) `undefined`

👉 Answer: B

---

**13. JavaScript uses which symbol for block?**

- A) `()`
- B) `[]`
- C) `{}`
- D) `<>`

👉 Answer: C

---

**14. Which loop executes at least once?**

- A) `for`
- B) `while`
- C) `do-while`
- D) `foreach`

👉 Answer: C

---

**15. Which operator checks value and type?**

- A) ==
- B) =
- C) ===
- D) !==

👉 Answer: C

---

**16. Which method gives string length?**

- A) size()
- B) count()
- C) length
- D) len()

👉 Answer: C

---

**17. Which method adds element at end of array?**

- A) pop()
- B) shift()
- C) push()
- D) unshift()

👉 Answer: C

---

**18. Which object represents date and time?**

- A) Time
- B) Clock
- C) Date
- D) Calendar

👉 Answer: C

---

**19. Which keyword is used for conditional statement?**

- A) switch
- B) if
- C) else
- D) All of the above

👉 Answer: D

---

**20. JavaScript code inside HTML is written using:**

- A) <js>
- B) <javascript>
- C) <script>

D) <code>

👉 Answer: C

---

## 🟡 MEDIUM LEVEL (21–40) — *With Code Snippets*

21. Output of the following code?

```
let x = 10;  
let y = "10";  
console.log(x == y);
```

- A) true
- B) false
- C) error
- D) undefined

👉 Answer: A

---

22. Output?

```
console.log(typeof null);
```

- A) null
- B) object
- C) undefined
- D) number

👉 Answer: B

---

23. Output?

```
let a;  
console.log(a);
```

- A) null
- B) 0
- C) undefined
- D) error

👉 Answer: C

---

24. Output?

```
console.log(5 + "5");
```

- A) 10
- B) 55
- C) error
- D) NaN

👉 Answer: B

---

## 25. Output?

```
console.log("Hello".toUpperCase());
```

- A) hello
- B) HELLO
- C) Error
- D) undefined

👉 Answer: B

---

## 26. Output?

```
let arr = [1,2,3];
arr.pop();
console.log(arr);
```

- A) [1,2,3]
- B) [1,2]
- C) [2,3]
- D) [1,3]

👉 Answer: B

---

## 27. Output?

```
console.log(Boolean(0));
```

- A) true
- B) false
- C) error
- D) undefined

👉 Answer: B

---

## 28. Output?

```
let x = 5;  
{  
    let x = 10;  
}  
console.log(x);
```

- A) 10
- B) 5
- C) error
- D) undefined

👉 Answer: B

---

## 29. Output?

```
console.log("abc".length);
```

- A) 2
- B) 3
- C) 4
- D) error

👉 Answer: B

---

## 30. Output?

```
let d = new Date();  
console.log(typeof d);
```

- A) date
- B) string
- C) object
- D) number

👉 Answer: C

---

## 31. Which method removes first array element?

- A) pop()
- B) shift()
- C) unshift()
- D) slice()

👉 Answer: B

---

**32. Which scope does `var` have?**

- A) Block
- B) Function
- C) Global only
- D) Module

👉 Answer: B

---

**33. Which method converts array to string?**

- A) `join()`
- B) `split()`
- C) `concat()`
- D) `slice()`

👉 Answer: A

---

**34. Output?**

```
console.log(10 > 5 && 5 > 1);
```

- A) true
- B) false
- C) error
- D) undefined

👉 Answer: A

---

**35. Which loop is best when count is known?**

- A) `while`
- B) `do-while`
- C) `for`
- D) `foreach`

👉 Answer: C

---

**36. Output?**

```
console.log(parseInt("10px"));
```

- A) 10
- B) NaN
- C) error

D) px10

👉 Answer: A

---

**37. Which Date method returns year?**

- A) getYear()
- B) getFullYear()
- C) year()
- D) getDate()

👉 Answer: B

---

**38. Which operator is logical OR?**

- A) &&
- B) ||
- C) !
- D) ??

👉 Answer: B

---

**39. Which keyword creates block scope?**

- A) var
- B) let
- C) static
- D) global

👉 Answer: B

---

**40. Output?**

```
console.log("5" - 2);
```

- A) 3
- B) 52
- C) NaN
- D) error

👉 Answer: A

---

🔴 **HARD LEVEL (41–60) — Tricky & CCEE Favorite**

**41. Output?**

```
console.log([] == false);
```

- A) true
  - B) false
  - C) error
  - D) undefined
- 👉 Answer: A
- 

#### 42. Output?

```
console.log(" " == false);
```

- A) true
  - B) false
  - C) error
  - D) undefined
- 👉 Answer: A
- 

#### 43. Output?

```
let x = 10;
if(true){
    var x = 20;
}
console.log(x);
```

- A) 10
  - B) 20
  - C) error
  - D) undefined
- 👉 Answer: B
- 

#### 44. Output?

```
let x = 10;
if(true){
    let x = 20;
}
console.log(x);
```

- A) 10
  - B) 20
  - C) error
  - D) undefined
- 👉 Answer: A
- 

#### 45. Output?

```
console.log(typeof NaN);
```

- A) NaN
- B) number
- C) undefined
- D) object

👉 Answer: B

---

#### 46. Output?

```
console.log(0.1 + 0.2 == 0.3);
```

- A) true
- B) false
- C) error
- D) undefined

👉 Answer: B

---

#### 47. Which value is falsy?

- A) "0"
- B) []
- C) {}
- D) 0

👉 Answer: D

---

#### 48. Output?

```
console.log(typeof typeof 10);
```

- A) number
- B) string
- C) undefined

D) object

👉 Answer: B

---

**49. Which array method does NOT change original array?**

A) push()

B) pop()

C) slice()

D) shift()

👉 Answer: C

---

**50. Output?**

```
let a = [1,2,3];
let b = a;
b.push(4);
console.log(a);
```

A) [1,2,3]

B) [1,2,3,4]

C) [4]

D) error

👉 Answer: B

---

**51. JavaScript uses which type system?**

A) Static

B) Strong

C) Dynamic

D) Strict

👉 Answer: C

---

**52. Output?**

```
console.log(new Date("2024-02-30"));
```

A) error

B) Invalid Date

C) null

D) undefined

👉 Answer: B

---

**53. Which statement skips current iteration?**

- A) stop
- B) break
- C) continue
- D) skip

👉 Answer: C

---

**54. Output?**

```
console.log (!! "hello");
```

- A) true
- B) false
- C) error
- D) undefined

👉 Answer: A

---

**55. Which operator is ternary?**

- A) ?:
- B) ??
- C) &&
- D) ||

👉 Answer: A

---

**56. Which string method extracts substring?**

- A) cut()
- B) substr()
- C) split()
- D) trim()

👉 Answer: B

---

**57. Which Date method returns day of month?**

- A) getDay()
- B) getDate()
- C) getMonth()
- D) getTime()

👉 Answer: B

---

**58. Output?**

```
console.log([1,2] + [3,4]);
```

- A) [1,2,3,4]
- B) "1,23,4"
- C) error
- D) NaN

👉 Answer: B

---

#### 59. Which is NOT a loop in JavaScript?

- A) for
- B) while
- C) repeat
- D) do-while

👉 Answer: C

---

#### 60. JavaScript arrays can store:

- A) Same type only
- B) Different data types
- C) Numbers only
- D) Strings only

👉 Answer: B

## SESSIONS 6 & 7 – JAVASCRIPT (OBJECTS, FUNCTIONS & OOP)

---

### EASY LEVEL (1–20)

#### 1. In JavaScript, objects are:

- A) Primitive types
- B) Collections of key–value pairs
- C) Arrays only
- D) Functions only

👉 Answer: B

---

**2. Which symbol is used to access object properties?**

- A) :
- B) #
- C) .
- D) @

👉 Answer: C

---

**3. How do you define an object in JavaScript?**

- A) {}
- B) []
- C) ()
- D) <>

👉 Answer: A

---

**4. Which keyword is used to create object instance?**

- A) new
- B) this
- C) create
- D) object

👉 Answer: A

---

**5. Object properties are:**

- A) Only values
- B) Only functions
- C) Variables inside objects
- D) Classes

👉 Answer: C

---

**6. Object methods are:**

- A) Variables
- B) Functions inside objects
- C) Classes
- D) Arrays

👉 Answer: B

---

**7. Which keyword refers to current object?**

- A) object
- B) self
- C) this

D) current

👉 Answer: C

---

**8. Functions in JavaScript are:**

A) Data types

B) Objects

C) Keywords

D) Operators

👉 Answer: B

---

**9. Which keyword defines a function?**

A) func

B) method

C) function

D) def

👉 Answer: C

---

**10. Function parameters are:**

A) Values passed to function

B) Variables defined in function definition

C) Return values

D) Objects

👉 Answer: B

---

**11. Arguments are:**

A) Variables in function

B) Values passed during call

C) Objects

D) Return values

👉 Answer: B

---

**12. JavaScript supports:**

A) Class-based OOP only

B) Prototype-based OOP

C) No OOP

D) Interface-based OOP

👉 Answer: B

---

**13. Prototype is used for:**

- A) Styling
- B) Inheritance
- C) Encapsulation
- D) Compilation

👉 **Answer: B**

---

**14. Constructor function is used to:**

- A) Destroy objects
- B) Create objects
- C) Inherit methods
- D) Validate data

👉 **Answer: B**

---

**15. Which keyword is used to define constructor?**

- A) constructor
- B) function
- C) new
- D) class

👉 **Answer: B**

---

**16. Which OOP concept hides internal details?**

- A) Inheritance
- B) Polymorphism
- C) Abstraction
- D) Encapsulation

👉 **Answer: D**

---

**17. Which OOP concept allows method overriding?**

- A) Encapsulation
- B) Abstraction
- C) Polymorphism
- D) Constructor

👉 **Answer: C**

---

**18. Which OOP concept reuses code?**

- A) Encapsulation
- B) Inheritance
- C) Abstraction
- D) Polymorphism

👉 **Answer: B**

---

**19. Closures are related to:**

- A) Loops
- B) Scope
- C) Arrays
- D) Objects only

👉 **Answer: B**

---

**20. JavaScript is:**

- A) Class-based
- B) Prototype-based
- C) Object-free
- D) Static

👉 **Answer: B**

---

## 🟡 MEDIUM LEVEL (21–40) — *With Code Snippets*

**21. Output?**

```
let obj = { a: 10 };
console.log(obj.a);
```

- A) 10
- B) undefined
- C) error
- D) null

👉 **Answer: A**

---

**22. Output?**

```
let user = { name: "Amol", age: 23 };
console.log(user["age"]);
```

- A) Amol
- B) 23
- C) undefined
- D) error

👉 **Answer: B**

---

### 23. Output?

```
let obj = {};
obj.x = 5;
console.log(obj.x);
```

- A) undefined
- B) error
- C) 5
- D) null

👉 Answer: C

---

### 24. Output?

```
function add(a, b) {
  return a + b;
}
console.log(add(2, 3));
```

- A) 23
- B) 5
- C) undefined
- D) error

👉 Answer: B

---

### 25. Output?

```
function test() {
  return;
}
console.log(test());
```

- A) null
- B) 0
- C) undefined
- D) error

👉 Answer: C

---

### 26. Output?

```
let person = {
```

```
name: "Ram",
greet: function() {
  return "Hello " + this.name;
};
console.log(person.greet());
```

- A) Hello
- B) Hello Ram
- C) Ram
- D) undefined

👉 Answer: B

---

## 27. Output?

```
function Person(name) {
  this.name = name;
}
let p = new Person("Shyam");
console.log(p.name);
```

- A) undefined
- B) error
- C) Shyam
- D) Person

👉 Answer: C

---

## 28. Which statement creates an object?

- A) let o = Object;
- B) let o = new Object();
- C) let o = Object[];
- D) let o = {};

👉 Answer: B & D ✓ (both valid)

---

## 29. Output?

```
let a = function() {
  return 10;
}
console.log(a());
```

- A) function
  - B) 10
  - C) error
  - D) undefined
- 👉 Answer: B
- 

### 30. Output?

```
let x = 10;
function test() {
  let x = 20;
}
test();
console.log(x);
```

- A) 10
- B) 20
- C) undefined
- D) error

👉 Answer: A

---

### 31. Functions in JavaScript are:

- A) First-class citizens
- B) Second-class
- C) Not objects
- D) Deprecated

👉 Answer: A

---

### 32. Which method adds property to prototype?

- A) object.add
- B) Object.prototype
- C) constructor
- D) this.prototype

👉 Answer: B

---

### 33. Output?

```
function outer() {
  let x = 10;
```

```
function inner() {  
    return x;  
}  
return inner;  
}  
let f = outer();  
console.log(f());
```

- A) 10
- B) undefined
- C) error
- D) null

👉 Answer: A

---

#### 34. Closure allows access to:

- A) Global scope only
- B) Parent function scope
- C) Block scope only
- D) Local scope only

👉 Answer: B

---

#### 35. Output?

```
console.log(typeof function(){});
```

- A) object
- B) function
- C) undefined
- D) method

👉 Answer: B

---

#### 36. Which keyword is used to inherit prototype?

- A) extends
- B) prototype
- C) inherits
- D) super

👉 Answer: B

---

#### 37. Output?

```
let obj = { a: 1 };
console.log("a" in obj);
```

- A) true
- B) false
- C) error
- D) undefined

👉 Answer: A

---

### 38. Which OOP concept binds data and methods?

- A) Abstraction
- B) Polymorphism
- C) Encapsulation
- D) Inheritance

👉 Answer: C

---

### 39. Which function is anonymous?

- A) function test() {}
- B) let f = function(){};
- C) function call(){};
- D) constructor(){};

👉 Answer: B

---

### 40. Prototype chain is used for:

- A) Validation
- B) Styling
- C) Property lookup
- D) Compilation

👉 Answer: C

---

## 🔴 HARD LEVEL (41–60) — *Tricky & CCEE Favorites*

### 41. Output?

```
console.log(typeof {});
```

- A) object
- B) function
- C) undefined

D) class

👉 Answer: A

---

#### 42. Output?

```
function A(){}
A.prototype.x = 10;
let a = new A();
console.log(a.x);
```

A) undefined

B) error

C) 10

D) null

👉 Answer: C

---

#### 43. Output?

```
let obj = { x: 10 };
let ref = obj;
ref.x = 20;
console.log(obj.x);
```

A) 10

B) 20

C) undefined

D) error

👉 Answer: B

---

#### 44. Which feature allows multiple forms of same function?

A) Inheritance

B) Encapsulation

C) Polymorphism

D) Abstraction

👉 Answer: C

---

#### 45. Output?

```
console.log(this);
```

(in browser global scope)

- A) null
- B) window
- C) document
- D) undefined

👉 Answer: B

---

#### 46. Output?

```
function test() {  
    console.log(this);  
}  
test();
```

- A) window
- B) undefined
- C) test
- D) error

👉 Answer: A (*non-strict mode*)

---

#### 47. JavaScript does NOT support:

- A) Encapsulation
- B) Inheritance
- C) Interfaces
- D) Polymorphism

👉 Answer: C

---

#### 48. Output?

```
let obj = {  
    x: 10,  
    getX() {  
        return this.x;  
    }  
};  
let f = obj.getX;  
console.log(f());
```

- A) 10
- B) undefined
- C) error

D) null

👉 Answer: B

---

**49. Closures are created when:**

- A) Function is executed
- B) Function is returned
- C) Variable is global
- D) Loop ends

👉 Answer: B

---

**50. Which keyword enables strict mode?**

- A) use strict
- B) "strict"
- C) "use strict"
- D) strict

👉 Answer: C

---

**51. Output?**

```
"use strict";
function test() {
    console.log(this);
}
test();
```

- A) window
- B) undefined
- C) error
- D) test

👉 Answer: B

---

**52. Which method creates object with prototype?**

- A) Object.new()
- B) Object.create()
- C) new Object()
- D) create()

👉 Answer: B

---

**53. Prototype-based inheritance means:**

- A) Class inheritance
- B) Object inherits from object
- C) Interface inheritance
- D) Method inheritance

👉 Answer: B

---

**54. Which OOP concept focuses on WHAT not HOW?**

- A) Encapsulation
- B) Inheritance
- C) Abstraction
- D) Polymorphism

👉 Answer: C

---

**55. Output?**

```
let x = function() {};
console.log(x instanceof Object);
```

- A) true
- B) false
- C) error
- D) undefined

👉 Answer: A

---

**56. Which keyword calls parent constructor?**

- A) parent
- B) this
- C) super
- D) prototype

👉 Answer: C

---

**57. Functions passed as arguments are called:**

- A) Closures
- B) Callbacks
- C) Prototypes
- D) Methods

👉 Answer: B

---

**58. Output?**

```
console.log(typeof class A {});
```

- A) class
- B) function
- C) object
- D) undefined

👉 Answer: B

---

#### 59. Which is true about JavaScript OOP?

- A) Fully class-based
- B) No inheritance
- C) Prototype-based
- D) No objects

👉 Answer: C

---

#### 60. Closures help in:

- A) Data hiding
- B) Memory leak
- C) Faster execution
- D) Compilation

👉 Answer: A

## SESSIONS 8 & 9 – JAVASCRIPT (DOM, FORMS, REGEX, ASYNC, DEBUGGING)

### EASY LEVEL (1–30)

#### 1. DOM stands for:

- A) Data Object Model
- B) Document Object Model
- C) Digital Object Method
- D) Display Object Model

👉 Answer: B

---

#### 2. DOM represents an HTML document as:

- A) Table
- B) Array
- C) Tree structure
- D) Stack

👉 Answer: C

---

**3. Which object represents the entire web page?**

- A) window
- B) screen
- C) document
- D) navigator

👉 Answer: C

---

**4. Which DOM method selects element by ID?**

- A) getElement()
- B) getElementById()
- C) querySelectorAll()
- D) getNodeById()

👉 Answer: B

---

**5. Which method selects ALL matching elements?**

- A) querySelector()
- B) getElementById()
- C) getElementsByClassName()
- D) getElement()

👉 Answer: C

---

**6. DOM events are:**

- A) Errors
- B) CSS rules
- C) User/browser actions
- D) HTML attributes

👉 Answer: C

---

**7. Which event fires when button is clicked?**

- A) onload
- B) onchange
- C) onclick
- D) onsubmit

👉 Answer: C

---

**8. DOM manipulation means:**

- A) Styling HTML
- B) Changing HTML dynamically
- C) Database update

D) Server execution

👉 Answer: B

---

**9. Which property changes element content?**

- A) value
- B) innerHTML
- C) style
- D) className

👉 Answer: B

---

**10. Which method attaches an event handler?**

- A) attach()
- B) addEvent()
- C) addEventListener()
- D) bind()

👉 Answer: C

---

**11. HTML forms are mainly used to:**

- A) Display data
- B) Collect user input
- C) Style page
- D) Navigate pages

👉 Answer: B

---

**12. Which attribute makes input mandatory?**

- A) validate
- B) required
- C) compulsory
- D) must

👉 Answer: B

---

**13. Which input type validates email automatically?**

- A) text
- B) mail
- C) email
- D) validate

👉 Answer: C

---

**14. Regular expressions are used for:**

- A) Animation
- B) Pattern matching
- C) Styling
- D) Compilation

👉 Answer: B

---

**15. Regex literal starts and ends with:**

- A) #
- B) //
- C) /
- D) <>

👉 Answer: C

---

**16. Which regex matches digits?**

- A) \w
- B) \d
- C) \s
- D) \D

👉 Answer: B

---

**17. Which block catches runtime errors?**

- A) try
- B) finally
- C) catch
- D) throw

👉 Answer: C

---

**18. Which keyword throws custom error?**

- A) error
- B) throw
- C) catch
- D) stop

👉 Answer: B

---

**19. Browser DevTools are used for:**

- A) Coding
- B) Debugging
- C) Compilation
- D) Hosting

👉 Answer: B

---

**20. `console.log()` is used for:**

- A) Styling
- B) Debugging
- C) Validation
- D) Networking

👉 **Answer: B**

---

**21. `JSLint` is used to:**

- A) Compile JS
- B) Check code quality
- C) Debug runtime
- D) Minify code

👉 **Answer: B**

---

**22. JavaScript is asynchronous using:**

- A) Threads
- B) Event loop
- C) Compiler
- D) Stack only

👉 **Answer: B**

---

**23. Callback is:**

- A) Returned function
- B) Function passed as argument
- C) Promise
- D) Event

👉 **Answer: B**

---

**24. Promise represents:**

- A) Immediate value
- B) Future value
- C) Past value
- D) Constant value

👉 **Answer: B**

---

**25. Promise states include:**

- A) start, end
- B) open, close
- C) pending, fulfilled, rejected

D) init, done

👉 Answer: C

---

**26. Which keyword defines async function?**

- A) await
- B) async
- C) defer
- D) promise

👉 Answer: B

---

**27. await pauses execution until:**

- A) Function ends
- B) Promise resolves
- C) Error occurs
- D) Loop ends

👉 Answer: B

---

**28. Which improves UI responsiveness?**

- A) Synchronous JS
- B) Asynchronous JS
- C) Blocking calls
- D) Infinite loops

👉 Answer: B

---

**29. Which event fires when page loads completely?**

- A) onclick
- B) DOMContentLoaded
- C) onload
- D) onchange

👉 Answer: C

---

**30. Which object represents browser window?**

- A) document
- B) window
- C) navigator
- D) screen

👉 Answer: B

---

## MEDIUM LEVEL (31–60) — SNIPPETS INCLUDED

### 31. Output?

```
console.log(document.nodeName);
```

- A) html
- B) #document
- C) document
- D) body

👉 Answer: B

---

### 32. What does this do?

```
document.getElementById("msg").innerHTML = "Hi";
```

- A) Reads content
- B) Deletes element
- C) Updates content
- D) Creates element

👉 Answer: C

---

### 33. Output?

```
document.querySelector("p");
```

- A) All `<p>` elements
- B) First `<p>` element
- C) Last `<p>` element
- D) Error

👉 Answer: B

---

### 34. Output?

```
document.querySelectorAll("div").length;
```

- A) 0
- B) 1
- C) Number of `<div>` elements

D) Error

👉 Answer: C

---

35. In this code, `show` is:

```
btn.addEventListener("click", show);
```

A) Event

B) Callback function

C) Error

D) Object

👉 Answer: B

---

36. Effect?

```
<form onsubmit="return false">
```

A) Submit form

B) Reload page

C) Prevent submit

D) Validate form

👉 Answer: C

---

37. Output?

```
/abc/.test("abcdef");
```

A) true

B) false

C) Error

D) undefined

👉 Answer: A

---

38. Output?

```
/\d+/.test("abc123");
```

A) true

B) false

- C) Error
  - D) undefined
- 👉 Answer: A
- 

### 39. Output?

```
try { throw "err"; } catch(e) { console.log(e); }
```

- A) undefined
- B) error
- C) err
- D) catch

👉 Answer: C

---

### 40. Purpose of this code?

```
console.error("Fail");
```

- A) Stop execution
- B) Show error in console
- C) Throw exception
- D) Compile error

👉 Answer: B

---

### 41. Output?

```
console.log("A");
setTimeout(()=>console.log("B"), 0);
console.log("C");
```

- A) A B C
- B) B A C
- C) A C B
- D) C A B

👉 Answer: C

---

### 42. Output?

```
async function f(){ return 5; }
f().then(console.log);
```

- A) Promise
- B) undefined
- C) 5
- D) Error

👉 Answer: C

---

#### 43. Where is this valid?

```
await fetch(url);
```

- A) Any function
- B) async function
- C) Loop
- D) Global only

👉 Answer: B

---

#### 44. Output?

```
Promise.resolve(2).then(x => x*2);
```

- A) 2
- B) 4
- C) Promise
- D) Error

👉 Answer: C

---

#### 45. Output?

```
Promise.reject("err").catch(console.log);
```

- A) Nothing
- B) err
- C) rejected
- D) Error

👉 Answer: B

---

#### 46. Purpose?

```
document.createElement( "p" );
```

- A) Display paragraph
- B) Create node
- C) Delete node
- D) Clone node

👉 Answer: B

---

#### 47. Effect?

```
document.body.appendChild(el);
```

- A) Remove element
- B) Append element
- C) Replace element
- D) Hide element

👉 Answer: B

---

#### 48. Effect?

```
event.preventDefault();
```

- A) Stop event
- B) Stop default behavior
- C) Stop propagation
- D) Remove handler

👉 Answer: B

---

#### 49. Output?

```
input.value = "";
```

- A) Disable input
- B) Clear input
- C) Delete input
- D) Hide input

👉 Answer: B

---

#### 50. Output?

```
/^[a-z]+$/ .test("abc123");
```

- A) true
- B) false
- C) Error
- D) undefined

👉 Answer: B

---

**51. Network tab in DevTools shows:**

- A) DOM tree
- B) CSS rules
- C) HTTP requests
- D) JS errors

👉 Answer: C

---

**52. Sources tab is used for:**

- A) Styling
- B) Debugging JS
- C) Validation
- D) Deployment

👉 Answer: B

---

**53. Callback hell means:**

- A) Infinite loop
- B) Nested callbacks
- C) Async error
- D) Memory leak

👉 Answer: B

---

**54. Which avoids callback hell?**

- A) Loops
- B) Promises / async-await
- C) CSS
- D) DOM

👉 Answer: B

---

**55. fetch() returns:**

- A) Data
- B) Response
- C) Promise

D) JSON

👉 Answer: C

---

**56. Convert response to JSON using:**

- A) parse()
- B) stringify()
- C) json()
- D) object()

👉 Answer: C

---

**57. Event bubbling means:**

- A) Parent → Child
- B) Child → Parent
- C) Stop event
- D) Ignore event

👉 Answer: B

---

**58. stopPropagation() does:**

- A) Stop default
- B) Stop bubbling
- C) Remove event
- D) Throw error

👉 Answer: B

---

**59. DOMContentLoaded fires when:**

- A) Page fully loaded
- B) DOM ready
- C) CSS loaded
- D) Images loaded

👉 Answer: B

---

**60. JSLint checks:**

- A) Runtime errors
- B) Syntax & style
- C) Network issues
- D) Memory leaks

👉 Answer: B

---

## HARD LEVEL (61–90) — ASYNC + TRICKY

### 61. Output?

```
console.log("1");
setTimeout(()=>console.log("2"));
console.log("3");
```

- A) 1 2 3
- B) 1 3 2
- C) 2 1 3
- D) 3 1 2

 **Answer:** B

---

### 62. Output?

```
async function f(){ return 10; }
console.log(f());
```

- A) 10
- B) undefined
- C) Promise
- D) Error

 **Answer:** C

---

### 63. Output?

```
Promise.all([Promise.resolve(1), Promise.resolve(2)])
.then(console.log);
```

- A) 1
- B) 2
- C) [1,2]
- D) Promise

 **Answer:** C

---

### 64. Output?

```
Promise.race([
  new Promise(r=>setTimeout(()=>r(1), 100)),
  new Promise(r=>setTimeout(()=>r(2), 50))
```

```
]).then(console.log);
```

- A) 1
- B) 2
- C) [1,2]
- D) Error

👉 Answer: B

---

#### 65. Output?

```
try { JSON.parse("{x:1}"); }
catch(e){ console.log("error"); }
```

- A) x
- B) 1
- C) error
- D) undefined

👉 Answer: C

---

#### 66. Output?

```
document.querySelectorAll("div") instanceof NodeList;
```

- A) true
- B) false
- C) Error
- D) undefined

👉 Answer: A

---

#### 67. Which is asynchronous?

- A) for loop
- B) fetch()
- C) assignment
- D) if statement

👉 Answer: B

---

#### 68. Async/await internally uses:

- A) Threads
- B) Promises
- C) Timers

D) Loops

👉 Answer: B

---

**69. Which handles async errors best?**

- A) if-else
- B) try-catch
- C) finally only
- D) switch

👉 Answer: B

---

**70. Which prevents form submission?**

- A) stopPropagation()
- B) preventDefault()
- C) return true
- D) submit()

👉 Answer: B

---

**71. Which method removes event listener?**

- A) detach()
- B) removeEventListener()
- C) deleteEvent()
- D) stopEvent()

👉 Answer: B

---

**72. Which method clones DOM node?**

- A) copyNode()
- B) cloneNode()
- C) duplicate()
- D) replicate()

👉 Answer: B

---

**73. Which object controls browser history?**

- A) document
- B) window
- C) history
- D) navigator

👉 Answer: C

---

**74. Regex flag for case-insensitive match:**

- A) g
- B) m
- C) i
- D) s

👉 Answer: C

---

**75. Which improves JS code quality?**

- A) JSLint
- B) Compiler
- C) Browser
- D) Server

👉 Answer: A

---

**76. Which prevents UI blocking?**

- A) Synchronous JS
- B) Async JS
- C) Loops
- D) Recursion

👉 Answer: B

---

**77. Which event fires first?**

- A) onload
- B) DOMContentLoaded
- C) onclick
- D) onchange

👉 Answer: B

---

**78. Which async syntax is cleanest?**

- A) Callback
- B) Promise
- C) async/await
- D) setTimeout

👉 Answer: C

---

**79. Which handles multiple async tasks together?**

- A) Promise.all()
- B) Promise.then()
- C) Promise.catch()
- D) Promise.resolve()

👉 Answer: A

---

**80. Event loop manages:**

- A) Memory
- B) Async execution
- C) DOM tree
- D) CSS

👉 Answer: B

---

**81. Which DOM node represents text?**

- A) Element node
- B) Attribute node
- C) Text node
- D) Comment node

👉 Answer: C

---

**82. Which method submits form via JS?**

- A) send()
- B) submit()
- C) post()
- D) push()

👉 Answer: B

---

**83. Which tool shows JS errors?**

- A) Network tab
- B) Elements tab
- C) Console tab
- D) Sources tab

👉 Answer: C

---

**84. Which async method waits for ALL promises?**

- A) Promise.any()
- B) Promise.race()
- C) Promise.all()
- D) Promise.resolve()

👉 Answer: C

---

**85. Which avoids callback nesting?**

- A) Loops
- B) Promises
- C) CSS

D) DOM

👉 Answer: B

---

**86. Which object stores form elements?**

- A) window
- B) document.forms
- C) history
- D) navigator

👉 Answer: B

---

**87. Which method stops default anchor navigation?**

- A) stopPropagation()
- B) preventDefault()
- C) break
- D) return

👉 Answer: B

---

**88. Which async feature improves readability?**

- A) Callbacks
- B) Promises
- C) async/await
- D) Timers

👉 Answer: C

---

**89. Which error type occurs at runtime?**

- A) Syntax error
- B) Logical error
- C) Runtime error
- D) Compile error

👉 Answer: C

---

**90. Main benefit of async programming:**

- A) Faster CPU
- B) Non-blocking execution
- C) Smaller code
- D) Compilation

👉 Answer: B



# SESSION 10 – JSON & jQuery (WPT – CCEE)

---



## EASY LEVEL (1–20)

**1. JSON stands for:**

- A) Java Source Object Notation
- B) JavaScript Object Notation
- C) Java Serialized Object Notation
- D) Java Syntax Object Notation

👉 Answer: B

---

**2. JSON is mainly used for:**

- A) Styling web pages
- B) Data interchange
- C) Database design
- D) UI rendering

👉 Answer: B

---

**3. JSON data is written in:**

- A) XML format
- B) Plain text format
- C) Binary format
- D) HTML format

👉 Answer: B

---

**4. JSON syntax is derived from:**

- A) Java
- B) C++
- C) JavaScript
- D) Python

👉 Answer: C

---

**5. JSON supports which data type?**

- A) Function
- B) Undefined
- C) String
- D) Date

👉 Answer: C

---

**6. JSON object is written using:**

- A) ()
- B) []
- C) {}
- D) <>

👉 Answer: C

---

**7. JSON array is written using:**

- A) {}
- B) ()
- C) []
- D) <>

👉 Answer: C

---

**8. JSON keys must be:**

- A) Numbers
- B) Boolean
- C) Strings
- D) Functions

👉 Answer: C

---

**9. Which is a valid JSON value?**

- A) undefined
- B) function(){}  
C) null
- D) NaN

👉 Answer: C

---

**10. JSON files usually have extension:**

- A) .js
- B) .json
- C) .xml
- D) .txt

👉 Answer: B

---

**11. jQuery is a:**

- A) Programming language
- B) JavaScript library
- C) Framework
- D) Database

👉 Answer: B

---

**12. jQuery is used to:**

- A) Replace JavaScript
- B) Simplify JavaScript coding
- C) Compile JS
- D) Create database

👉 Answer: B

---

**13. jQuery code starts with:**

- A) jQuery()
- B) \$()
- C) jquery{}
- D) @()

👉 Answer: B

---

**14. Which selector selects element by id in jQuery?**

- A) .id
- B) #id
- C) id
- D) \*id

👉 Answer: B

---

**15. Which selector selects elements by class?**

- A) #class
- B) .class
- C) class
- D) \*class

👉 Answer: B

---

**16. Which event fires on mouse click?**

- A) hover()
- B) click()
- C) change()
- D) submit()

👉 Answer: B

---

**17. jQuery animations are used for:**

- A) Validation
- B) Effects and transitions
- C) Networking

D) Security

👉 Answer: B

---

**18. Which jQuery method hides an element?**

A) remove()

B) hide()

C) delete()

D) invisible()

👉 Answer: B

---

**19. jQuery DOM manipulation means:**

A) Creating server

B) Modifying HTML elements

C) Database update

D) Styling CSS only

👉 Answer: B

---

**20. jQuery plugins are:**

A) Server tools

B) External libraries extending jQuery

C) CSS files

D) Databases

👉 Answer: B

---

## 🟡 MEDIUM LEVEL (21–40) — *With Snippets*

**21. Which JSON is valid?**

A) `{name: "Amol"}`

B) `{"name": "Amol"}`

C) `{name = "Amol"}`

D) `('name': 'Amol')`

👉 Answer: B

---

**22. Output?**

```
JSON.parse(' {"a":10} ') .a;
```

A) "10"

B) 10

C) undefined

D) error

👉 Answer: B

---

### 23. Output?

```
JSON.stringify({x:5});
```

A) {x:5}

B) "x:5"

C) '{"x":5}'

D) error

👉 Answer: C

---

### 24. JSON.parse() is used to:

A) Convert object to JSON

B) Convert JSON to object

C) Validate HTML

D) Parse XML

👉 Answer: B

---

### 25. JSON.stringify() is used to:

A) Convert JSON to object

B) Convert object to JSON string

C) Parse HTML

D) Compress data

👉 Answer: B

---

### 26. Output?

```
let j = '{"a":[1,2,3]}';  
JSON.parse(j).a.length;
```

A) 1

B) 2

C) 3

D) error

👉 Answer: C

---

**27. Which JSON value represents empty object?**

- A) []
- B) {}
- C) null
- D) ""

👉 Answer: B

---

**28. jQuery code executes after DOM ready using:**

- A) \$(load)
- B) \$(document).ready()
- C) ready()
- D) onload()

👉 Answer: B

---

**29. Output?**

```
$( "#box" ).hide();
```

What does it do?

- A) Deletes element
- B) Hides element
- C) Disables element
- D) Removes CSS

👉 Answer: B

---

**30. Which method changes HTML content?**

- A) text()
- B) html()
- C) val()
- D) attr()

👉 Answer: B

---

**31. Which jQuery method gets input value?**

- A) text()
- B) html()
- C) val()
- D) data()

👉 Answer: C

---

**32. Output?**

```
$(".item").length;
```

- A) First element
- B) Total matched elements
- C) Error
- D) Last element

👉 Answer: B

---

### 33. Which jQuery event triggers on input change?

- A) click()
- B) keypress()
- C) change()
- D) hover()

👉 Answer: C

---

### 34. Which animation fades element out?

- A) slideUp()
- B) fadeOut()
- C) hide()
- D) remove()

👉 Answer: B

---

### 35. Which animation slides element down?

- A) slideDown()
- B) fadeIn()
- C) show()
- D) animate()

👉 Answer: A

---

### 36. Output?

```
$("#p").css("color", "red");
```

Effect?

- A) Delete element
- B) Change text color
- C) Add class
- D) Hide element

👉 Answer: B

---

**37. jQuery DOM traversal means:**

- A) Styling elements
- B) Navigating DOM tree
- C) Creating server
- D) Validating form

👉 Answer: B

---

**38. Which method selects parent element?**

- A) children()
- B) parent()
- C) next()
- D) siblings()

👉 Answer: B

---

**39. Which method selects child elements?**

- A) parent()
- B) find()
- C) children()
- D) next()

👉 Answer: C

---

**40. jQuery data attributes are accessed using:**

- A) attr()
- B) data()
- C) val()
- D) html()

👉 Answer: B

---

## 🔴 HARD LEVEL (41–60) — *Tricky & Snippet-Based*

**41. Which JSON is INVALID?**

- A) {"x":10}
- B) {"x":null}
- C) {"x":undefined}
- D) {"x":[1,2]}

👉 Answer: C

---

**42. Output?**

```
JSON.parse('{"a":10, "b":20}'.b);
```

- A) "20"
- B) 20
- C) undefined
- D) error

👉 Answer: B

---

#### 43. Output?

```
JSON.stringify([1,2,3]);
```

- A) [1,2,3]
- B) "1,2,3"
- C) "[1,2,3]"
- D) error

👉 Answer: C

---

#### 44. Which JSON value is NOT allowed?

- A) true
- B) false
- C) NaN
- D) null

👉 Answer: C

---

#### 45. jQuery chaining means:

- A) Calling multiple methods together
- B) Looping elements
- C) Callback nesting
- D) DOM traversal

👉 Answer: A

---

#### 46. Output?

```
$( "#box" ).addClass("red");
```

Effect?

- A) Adds CSS class
- B) Deletes element
- C) Hides element

D) Creates element

👉 Answer: A

---

#### 47. Output?

```
$( "#box" ).remove();
```

Effect?

- A) Hide element
- B) Remove from DOM
- C) Disable element
- D) Clear content

👉 Answer: B

---

#### 48. Which method attaches event handler?

- A) on()
- B) bind()
- C) click()
- D) All of the above

👉 Answer: D

---

#### 49. Output?

```
$( "p" ).eq(0).text();
```

Returns?

- A) All paragraph text
- B) First paragraph text
- C) Last paragraph text
- D) Error

👉 Answer: B

---

#### 50. Which method performs custom animation?

- A) animate()
- B) fadeIn()
- C) slideUp()
- D) hide()

👉 Answer: A

---

**51. Which utility function iterates array/object?**

- A) `$.map()`
- B) `$.each()`
- C) `$.loop()`
- D) `$.for()`

👉 Answer: B

---

**52. Which jQuery method makes AJAX call?**

- A) `$.ajax()`
- B) `$.call()`
- C) `$.http()`
- D) `$.fetch()`

👉 Answer: A

---

**53. Which jQuery method loads data into element?**

- A) `load()`
- B) `fetch()`
- C) `ajax()`
- D) `parse()`

👉 Answer: A

---

**54. jQuery plugin is:**

- A) Built-in method
- B) User-defined extension
- C) CSS file
- D) HTML tag

👉 Answer: B

---

**55. Which JSON feature makes it language-independent?**

- A) Syntax
- B) Text format
- C) Objects
- D) Arrays

👉 Answer: B

---

**56. Output?**

```
$.trim(" hi ");
```

A) " hi "

B) "hi"

C) " hi"

D) error

👉 Answer: B

---

**57. Which jQuery method clones element?**

A) copy()

B) clone()

C) duplicate()

D) repeat()

👉 Answer: B

---

**58. Which JSON structure supports key–value pairs?**

A) Array

B) Object

C) String

D) Number

👉 Answer: B

---

**59. Which jQuery feature improves reusability?**

A) Plugins

B) Selectors

C) Events

D) Animations

👉 Answer: A

---

**60. JSON is preferred over XML because it is:**

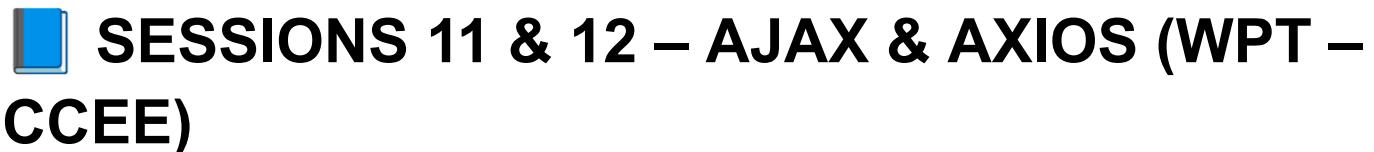
A) More secure

B) Lighter and faster

C) Binary

D) Compiled

👉 Answer: B



---

## EASY LEVEL (1–20)

### 1. AJAX stands for:

- A) Asynchronous Java and XML
- B) Asynchronous JavaScript and XML
- C) Advanced JavaScript and XML
- D) Asynchronous JSON and XML

 Answer: B

---

### 2. AJAX is mainly used to:

- A) Reload entire web page
- B) Load data asynchronously
- C) Style HTML
- D) Compile JavaScript

 Answer: B

---

### 3. AJAX allows web pages to:

- A) Refresh automatically
- B) Communicate with server without page reload
- C) Store data locally
- D) Run faster JavaScript

 Answer: B

---

### 4. Which data format is commonly used with AJAX today?

- A) XML
- B) HTML
- C) JSON
- D) CSV

 Answer: C

---

### 5. AJAX works on which side?

- A) Server-side only
- B) Client-side
- C) Database-side
- D) Network-side

 Answer: B

---

**6. Which object is used in classic AJAX?**

- A) fetch
- B) Promise
- C) XMLHttpRequest
- D) Axios

👉 Answer: C

---

**7. AJAX improves user experience by:**

- A) Blocking UI
- B) Reducing server load
- C) Partial page updates
- D) Increasing page size

👉 Answer: C

---

**8. Which library simplifies AJAX calls?**

- A) Bootstrap
- B) jQuery
- C) CSS
- D) HTML

👉 Answer: B

---

**9. jQuery AJAX method is:**

- A) \$.call()
- B) \$.request()
- C) \$.ajax()
- D) \$.http()

👉 Answer: C

---

**10. AJAX communicates with server using:**

- A) FTP
- B) HTTP
- C) SMTP
- D) TCP only

👉 Answer: B

---

**11. Axios is:**

- A) JavaScript framework
- B) Promise-based HTTP client
- C) Database
- D) Server

👉 Answer: B

---

**12. Axios is mainly used to:**

- A) Style UI
- B) Make HTTP requests
- C) Create server
- D) Validate forms

👉 Answer: B

---

**13. Axios returns:**

- A) Data directly
- B) Object
- C) Promise
- D) String

👉 Answer: C

---

**14. Which HTTP method retrieves data?**

- A) POST
- B) PUT
- C) GET
- D) DELETE

👉 Answer: C

---

**15. Which HTTP method sends data to server?**

- A) GET
- B) POST
- C) HEAD
- D) OPTIONS

👉 Answer: B

---

**16. AJAX requests are:**

- A) Blocking
- B) Synchronous only
- C) Asynchronous
- D) Sequential

👉 Answer: C

---

**17. Which event fires when AJAX request completes successfully?**

- A) onload
- B) onreadystatechange
- C) onerror

D) onsuccess

👉 Answer: B

---

**18. Axios automatically converts response to:**

- A) XML
- B) JSON
- C) HTML
- D) Text

👉 Answer: B

---

**19. Which HTTP status code indicates success?**

- A) 301
- B) 400
- C) 200
- D) 500

👉 Answer: C

---

**20. AJAX is used in live scoreboard because it:**

- A) Reloads page
- B) Updates data dynamically
- C) Blocks UI
- D) Uses database

👉 Answer: B

---

## 🟡 MEDIUM LEVEL (21–40) — WITH SNIPPETS

**21. Output?**

```
let xhr = new XMLHttpRequest();
xhr.open("GET", "data.json", true);
xhr.send();
```

What does this code do?

- A) Sends synchronous request
- B) Sends asynchronous GET request
- C) Sends POST request
- D) Error

👉 Answer: B

---

## 22. Output?

```
xhr.readyState === 4
```

Means:

- A) Request not started
- B) Loading
- C) Request completed
- D) Error

👉 Answer: C

---

## 23. Output?

```
xhr.status === 200
```

Indicates:

- A) Redirect
- B) Success
- C) Client error
- D) Server error

👉 Answer: B

---

## 24. Which method sends AJAX request?

- A) open()
- B) send()
- C) readyState
- D) status

👉 Answer: B

---

## 25. Which property handles AJAX response text?

- A) responseXML
- B) responseText
- C) result
- D) data

👉 Answer: B

---

## 26. Output?

```
$.ajax({  
  url: "data.json",  
  success: function(res) { console.log(res); }  
})
```

```
});
```

Purpose?

- A) Send POST request
- B) Perform AJAX GET request
- C) Parse XML
- D) Validate form

👉 Answer: B

---

**27. Which jQuery method is shorthand for AJAX GET?**

- A) \$.post()
- B) \$.load()
- C) \$.get()
- D) \$.ajaxGet()

👉 Answer: C

---

**28. Which jQuery method is shorthand for AJAX POST?**

- A) \$.send()
- B) \$.post()
- C) \$.submit()
- D) \$.ajaxPost()

👉 Answer: B

---

**29. Output?**

```
JSON.parse(xhr.responseText);
```

Does what?

- A) Convert object to JSON
- B) Convert JSON to JS object
- C) Validate JSON
- D) Compress data

👉 Answer: B

---

**30. Which AJAX architecture component handles UI?**

- A) Server
- B) Browser
- C) AJAX Engine
- D) Database

👉 Answer: C

---

**31. AJAX engine works:**

- A) On server
- B) Between browser and server
- C) Inside database
- D) Inside HTML

👉 Answer: B

---

**32. Axios GET request syntax:**

- A) axios.fetch(url)
- B) axios.request(url)
- C) axios.get(url)
- D) axios.call(url)

👉 Answer: C

---

**33. Output?**

```
axios.get("/data")
.then(res => console.log(res.data));
```

What is `res.data`?

- A) Full response
- B) JSON response body
- C) Headers
- D) Status code

👉 Answer: B

---

**34. Axios POST request syntax:**

- A) axios.send()
- B) axios.post(url,data)
- C) axios.push()
- D) axios.submit()

👉 Answer: B

---

**35. Axios error handling uses:**

- A) then()
- B) finally()
- C) catch()
- D) error()

👉 Answer: C

---

### 36. Output?

```
axios.get("/x")
  .catch(err => console.log(err));
```

Executes when:

- A) Request success
- B) Server error
- C) Network/error occurs
- D) Always

👉 Answer: C

---

### 37. Axios instance is used to:

- A) Replace Axios
- B) Create reusable configuration
- C) Compile JS
- D) Handle DOM

👉 Answer: B

---

### 38. Output?

```
const api = axios.create({ baseURL: "/api" });
```

Purpose?

- A) Create server
- B) Create Axios instance
- C) Send request
- D) Handle error

👉 Answer: B

---

### 39. Which Axios config sets headers?

- A) headers
- B) config
- C) options
- D) meta

👉 Answer: A

---

### 40. Live news app needs AJAX because:

- A) Page reloads every time

B) Data updates dynamically

C) Database changes

D) CSS updates

👉 Answer: B

---

## 🔴 HARD LEVEL (41–60) — TRICKY + LAB-ORIENTED

### 41. Output?

```
xhr.open("GET", url, false);
```

Request type?

A) Asynchronous

B) Synchronous

C) Parallel

D) Invalid

👉 Answer: B

---

### 42. Why synchronous AJAX is discouraged?

A) Faster

B) Blocks UI

C) Secure

D) Lightweight

👉 Answer: B

---

### 43. Output?

```
xhr.onreadystatechange = function() {
  if(xhr.readyState === 4 && xhr.status === 200)
    console.log("Done");
}
```

Purpose?

A) Send request

B) Handle successful response

C) Parse JSON

D) Create request

👉 Answer: B

---

**44. Which HTTP status indicates client error?**

- A) 2xx
- B) 3xx
- C) 4xx
- D) 5xx

👉 Answer: C

---

**45. Which Axios feature simplifies async code?**

- A) Callbacks
- B) Promises
- C) Threads
- D) Timers

👉 Answer: B

---

**46. Output?**

```
await axios.get("/data");
```

Valid only inside:

- A) Any function
- B) async function
- C) Loop
- D) Global scope

👉 Answer: B

---

**47. Which returns multiple AJAX results together?**

- A) axios.all()
- B) Promise.all()
- C) \$.ajaxAll()
- D) axios.chain()

👉 Answer: B

---

**48. Output?**

```
Promise.all([
  axios.get("/a"),
  axios.get("/b")
]);
```

Returns:

- A) Single response
- B) Array of responses
- C) First response
- D) Error

👉 Answer: B

---

**49. Axios automatically rejects promise when:**

- A) Status is 200
- B) Status is 2xx
- C) Status is not 2xx
- D) JSON invalid

👉 Answer: C

---

**50. Difference between AJAX & Axios:**

- A) Axios uses callbacks
- B) AJAX uses promises
- C) Axios is promise-based
- D) AJAX cannot handle JSON

👉 Answer: C

---

**51. Which is NOT part of AJAX architecture?**

- A) Browser
- B) AJAX engine
- C) Web server
- D) Compiler

👉 Answer: D

---

**52. Live scoreboard should use which HTTP method?**

- A) POST
- B) PUT
- C) GET
- D) DELETE

👉 Answer: C

---

**53. JSON from server is converted to object using:**

- A) JSON.stringify()
- B) JSON.parse()
- C) parseJSON()
- D) convert()

👉 Answer: B

---

**54. Axios interceptor is used for:**

- A) DOM manipulation
- B) Pre/post request processing
- C) Styling
- D) Validation

👉 Answer: B

---

**55. Which is true about Axios vs XMLHttpRequest?**

- A) Axios is older
- B) Axios simplifies syntax
- C) XMLHttpRequest uses promises
- D) Axios blocks UI

👉 Answer: B

---

**56. Output?**

```
axios.get("/x")
  .finally(()=>console.log("done"));
```

finally runs:

- A) Only on success
- B) Only on error
- C) Always
- D) Never

👉 Answer: C

---

**57. Which jQuery AJAX method loads HTML into element?**

- A) \$.ajax()
- B) \$.get()
- C) load()
- D) \$.html()

👉 Answer: C

---

**58. Which header tells server data type is JSON?**

- A) Accept
- B) Content-Type: application/json
- C) Cache-Control
- D) Host

👉 Answer: B

---

**59. Which improves performance in live updates?**

- A) Page reload
- B) Polling via AJAX
- C) Blocking calls
- D) Synchronous JS

👉 Answer: B

---

**60. Main benefit of Axios over jQuery AJAX:**

- A) Larger size
- B) Promise-based & cleaner syntax
- C) Needs jQuery
- D) Only works with XML

👉 Answer: B

---

## SESSION 13 – INTRODUCTION TO NODE.JS (WPT – CCEE)

---

### EASY LEVEL (1–10)

**1. Node.js is:**

- A) A programming language
- B) A JavaScript runtime environment
- C) A database
- D) A browser

👉 Answer: B

---

**2. Node.js is built on:**

- A) Java Virtual Machine
- B) Python runtime
- C) Chrome V8 engine
- D) Mozilla SpiderMonkey

👉 Answer: C

---

**3. Node.js is mainly used for:**

- A) UI design

B) Server-side programming

C) Database creation

D) Styling web pages

👉 Answer: B

---

#### 4. Which JavaScript engine does Node.js use?

A) Chakra

B) Nashorn

C) V8

D) Rhino

👉 Answer: C

---

#### 5. Node.js executes JavaScript on:

A) Browser only

B) Client side

C) Server side

D) Database side

👉 Answer: C

---

#### 6. REPL stands for:

A) Read Execute Print Loop

B) Read Evaluate Print Loop

C) Run Execute Program Loop

D) Read Execute Program List

👉 Answer: B

---

#### 7. Which command starts Node.js REPL?

A) node start

B) node repl

C) node

D) npm start

👉 Answer: C

---

#### 8. Which file extension is used for Node.js files?

A) .node

B) .java

C) .js

D) .json

👉 Answer: C

---

**9. Node.js follows which execution model?**

- A) Multi-threaded blocking
- B) Single-threaded blocking
- C) Single-threaded non-blocking
- D) Multi-threaded non-blocking

👉 Answer: C

---

**10. Node.js is best suited for:**

- A) CPU-intensive tasks
- B) Real-time applications
- C) Desktop UI
- D) Machine learning only

👉 Answer: B

---

## 🟡 MEDIUM LEVEL (11–20) — WITH SNIPPETS

**11. Difference between Browser JS and Node.js:**

- A) Browser JS has no JS engine
- B) Node.js has no DOM
- C) Node.js cannot access files
- D) Browser JS runs on server

👉 Answer: B

---

**12. Which global object is NOT available in Node.js?**

- A) process
- B) console
- C) window
- D) global

👉 Answer: C

---

**13. Output?**

```
console.log(typeof require);
```

- A) object
- B) function
- C) undefined
- D) module

👉 Answer: B

---

**14. Which ES6 feature is supported in Node.js?**

- A) let / const
- B) Arrow functions
- C) Classes
- D) All of the above

👉 Answer: D

---

**15. Output?**

```
let x = () => 5;  
console.log(x());
```

- A) x
- B) undefined
- C) 5
- D) error

👉 Answer: C

---

**16. Which command checks Node.js version?**

- A) node --version
- B) node -v
- C) Both A and B
- D) npm -v

👉 Answer: C

---

**17. Output?**

```
console.log(process.version);
```

- A) Node.js version
- B) npm version
- C) OS version
- D) Error

👉 Answer: A

---

**18. Which ES6 keyword declares constant?**

- A) let
- B) var
- C) const
- D) static

👉 Answer: C

---

### 19. Output?

```
let arr = [1,2,3];
arr.forEach(x => console.log(x));
```

- A) Prints array
- B) Prints 1 2 3
- C) Error
- D) Prints index

👉 Answer: B

---

### 20. Node.js uses which module system?

- A) AMD
- B) ES Modules only
- C) CommonJS
- D) UMD

👉 Answer: C

---

## 🔴 HARD LEVEL (21–30) — LAB + LOGIC BASED

### 21. Output?

```
function fact(n){
  if(n === 1) return 1;
  return n * fact(n-1);
}
console.log(fact(4));
```

- A) 6
- B) 12
- C) 24
- D) Error

👉 Answer: C

---

### 22. Which type of function is used above?

- A) Callback
- B) Arrow function
- C) Recursive function

D) Anonymous function

👉 Answer: C

---

### 23. Output?

```
for(let i=1;i<=5;i++){
  console.log(i);
}
```

- A) 1 to 5
- B) 0 to 4
- C) Infinite loop
- D) Error

👉 Answer: A

---

### 24. Output?

```
console.log(3 % 3 === 0 ? "foo" : "bar");
```

- A) foo
- B) bar
- C) foobar
- D) Error

👉 Answer: A

---

### 25. Output?

```
console.log(5 % 5 === 0 ? "bar" : "foo");
```

- A) foo
- B) bar
- C) foobar
- D) Error

👉 Answer: B

---

### 26. Output?

```
let n = 15;
if(n % 3 === 0 && n % 5 === 0)
  console.log("foobar");
```

- A) foo
- B) bar
- C) foobar
- D) Nothing

👉 Answer: C

---

**27. Which output is correct for number 7?**

- A) foo
- B) bar
- C) foobar
- D) 7

👉 Answer: D

---

**28. Node.js is NOT suitable for:**

- A) Chat applications
- B) Real-time APIs
- C) File streaming
- D) Heavy CPU computation

👉 Answer: D

---

**29. Which ES6 feature improves readability in Node.js async code?**

- A) Callbacks
- B) Promises
- C) async/await
- D) setTimeout

👉 Answer: C

---

**30. Main advantage of Node.js is:**

- A) Multithreading
- B) Blocking I/O
- C) Non-blocking I/O
- D) Heavy computation

👉 Answer: C



# SESSIONS 14 & 15 – NODE.JS ASYNCHRONOUS PROGRAMMING

---

## EASY LEVEL (1–20)

### 1. Asynchronous programming means:

- A) Code executes line by line only
- B) Code waits for each task to finish
- C) Non-blocking execution of tasks
- D) Multithreaded execution only

 Answer: C

---

### 2. Node.js supports asynchronous programming using:

- A) Threads
- B) Event loop
- C) Compiler
- D) JVM

 Answer: B

---

### 3. Callback is:

- A) Returned value
- B) Function passed as argument
- C) Promise
- D) Object

 Answer: B

---

### 4. Which function executes after async task completes?

- A) Main function
- B) Callback function
- C) Loop
- D) Constructor

 Answer: B

---

### 5. Asynchronous code helps to:

- A) Block execution
- B) Improve responsiveness
- C) Increase CPU usage

D) Reduce memory

👉 Answer: B

---

**6. Which function schedules delayed execution?**

A) setInterval()

B) setTimeout()

C) delay()

D) wait()

👉 Answer: B

---

**7. setTimeout() executes:**

A) Immediately

B) After given delay

C) Before delay

D) Randomly

👉 Answer: B

---

**8. setInterval() is used to:**

A) Execute once

B) Execute repeatedly

C) Stop execution

D) Delay promise

👉 Answer: B

---

**9. Promise represents:**

A) Past value

B) Immediate value

C) Future value

D) Constant value

👉 Answer: C

---

**10. Promise states are:**

A) start, stop

B) open, close

C) pending, fulfilled, rejected

D) init, done

👉 Answer: C

---

**11. Which method handles promise success?**

- A) catch()
- B) then()
- C) finally()
- D) resolve()

👉 Answer: B

---

**12. Which method handles promise failure?**

- A) then()
- B) resolve()
- C) catch()
- D) success()

👉 Answer: C

---

**13. async keyword is used to:**

- A) Block code
- B) Define asynchronous function
- C) Create promise manually
- D) Stop execution

👉 Answer: B

---

**14. await keyword is used to:**

- A) Define function
- B) Pause execution until promise resolves
- C) Reject promise
- D) Loop execution

👉 Answer: B

---

**15. await can be used only inside:**

- A) Any function
- B) Global scope
- C) async function
- D) Loop

👉 Answer: C

---

**16. Callback hell refers to:**

- A) Infinite loop
- B) Nested callbacks
- C) Syntax error
- D) Runtime error

👉 Answer: B

---

**17. Which avoids callback hell?**

- A) Loops
- B) Promises
- C) CSS
- D) HTML

👉 Answer: B

---

**18. Event loop is responsible for:**

- A) DOM rendering
- B) Handling async callbacks
- C) Compiling JS
- D) Memory allocation

👉 Answer: B

---

**19. Timers in Node.js are:**

- A) Blocking
- B) Synchronous
- C) Asynchronous
- D) Sequential

👉 Answer: C

---

**20. Node.js asynchronous programming is:**

- A) Multithreaded
- B) Blocking
- C) Non-blocking
- D) Sequential

👉 Answer: C

---

## 🟡 MEDIUM LEVEL (21–40) — WITH SNIPPETS

**21. Output?**

```
console.log("A");
setTimeout(() => console.log("B"), 1000);
console.log("C");
```

- A) A B C
- B) B A C
- C) A C B

D) C A B

👉 Answer: C

---

## 22. Output?

```
setTimeout(() => console.log("Hello"), 0);
console.log("World");
```

- A) Hello World
- B) World Hello
- C) Hello only
- D) Error

👉 Answer: B

---

## 23. Output?

```
function test(cb){
  cb();
}
test(() => console.log("Done"));
```

- A) Nothing
- B) Error
- C) Done
- D) Undefined

👉 Answer: C

---

## 24. Which function is callback in above code?

- A) test
- B) cb
- C) console.log
- D) arrow function

👉 Answer: D

---

## 25. Output?

```
Promise.resolve(10).then(x => console.log(x));
```

- A) Promise
- B) undefined

C) 10

D) Error

👉 Answer: C

---

## 26. Output?

```
Promise.reject("err").catch(e => console.log(e));
```

A) Nothing

B) err

C) rejected

D) Error

👉 Answer: B

---

## 27. Output?

```
Promise.resolve(1)
.then(x => x + 1)
.then(x => console.log(x));
```

A) 1

B) 2

C) Promise

D) Error

👉 Answer: B

---

## 28. Output?

```
async function f(){
  return 5;
}
f().then(console.log);
```

A) Promise

B) undefined

C) 5

D) Error

👉 Answer: C

---

### 29. Output?

```
async function f(){
  let x = await Promise.resolve(10);
  console.log(x);
}
f();
```

- A) undefined
- B) Promise
- C) 10
- D) Error

👉 Answer: C

---

### 30. Which keyword makes function return promise automatically?

- A) await
- B) async
- C) return
- D) resolve

👉 Answer: B

---

### 31. Output?

```
setInterval(() => console.log("Hi"), 1000);
```

- A) Prints once
- B) Prints repeatedly
- C) Error
- D) Blocks execution

👉 Answer: B

---

### 32. How to stop setInterval()?

- A) stopInterval()
- B) clearInterval()
- C) stop()
- D) clearTimeout()

👉 Answer: B

---

### 33. Output?

```
let id = setTimeout(() => console.log("X"), 1000);
clearTimeout(id);
```

- A) X printed
- B) Error
- C) Nothing printed
- D) Undefined

👉 Answer: C

---

#### 34. Which phase handles timers in event loop?

- A) Poll
- B) Check
- C) Timers
- D) Close

👉 Answer: C

---

#### 35. Promise.all() resolves when:

- A) First promise resolves
- B) All promises resolve
- C) Any promise resolves
- D) One promise rejects

👉 Answer: B

---

#### 36. Output?

```
Promise.all([
  Promise.resolve(1),
  Promise.resolve(2)
]).then(console.log);
```

- A) 1
- B) 2
- C) [1,2]
- D) Promise

👉 Answer: C

---

#### 37. Promise.race() resolves when:

- A) All promises resolve
- B) First promise resolves/rejects
- C) Last promise resolves

D) Any promise rejects

👉 Answer: B

---

### 38. Output?

```
Promise.race([
  new Promise(r => setTimeout(()=>r(1),100)),
  new Promise(r => setTimeout(()=>r(2),50))
]).then(console.log);
```

A) 1

B) 2

C) [1,2]

D) Error

👉 Answer: B

---

### 39. `async/await` internally uses:

A) Callbacks

B) Threads

C) Promises

D) Timers

👉 Answer: C

---

### 40. Which is NOT asynchronous?

A) `setTimeout`

B) `Promise`

C) `console.log`

D) `setInterval`

👉 Answer: C

---

## 🔴 HARD LEVEL (41–60) — TRICKY & LAB-ORIENTED

### 41. Output?

```
console.log("1");
Promise.resolve().then(()=>console.log("2"));
console.log("3");
```

- A) 1 2 3
- B) 1 3 2
- C) 2 1 3
- D) 3 1 2

👉 Answer: B

---

#### 42. Output?

```
setTimeout(() => console.log("A"), 0);
Promise.resolve().then(()=>console.log("B"));
console.log("C");
```

- A) A B C
- B) C B A
- C) B A C
- D) C A B

👉 Answer: B

---

#### 43. Which executes first?

- A) setTimeout callback
- B) Promise.then
- C) setInterval
- D) File I/O

👉 Answer: B

---

#### 44. Why promise microtasks run before timers?

- A) Higher priority queue
- B) Faster CPU
- C) Threading
- D) Blocking I/O

👉 Answer: A

---

#### 45. Output?

```
async function f(){
  throw "err";
}
f().catch(console.log);
```

A) Nothing

B) err

C) Promise

D) Error

👉 Answer: B

---

#### 46. Output?

```
async function f(){
  try{
    await Promise.reject("x");
  }catch(e){
    console.log(e);
  }
}
f();
```

A) Nothing

B) Promise

C) x

D) Error

👉 Answer: C

---

#### 47. Which handles async errors best?

A) if-else

B) try-catch

C) switch

D) loop

👉 Answer: B

---

#### 48. Which creates callback hell?

A) Nested promises

B) Nested callbacks

C) async-await

D) Event loop

👉 Answer: B

---

#### 49. Which avoids deeply nested async code?

A) Callbacks

B) Promises

C) async/await

D) Both B and C

👉 Answer: D

---

#### 50. Output?

```
Promise.resolve(1)
.then(x => { throw x; })
.catch(x => console.log(x));
```

A) 1

B) Error

C) Promise

D) undefined

👉 Answer: A

---

#### 51. Event loop manages:

A) Memory

B) Async tasks execution

C) Compilation

D) DOM

👉 Answer: B

---

#### 52. Which timer executes repeatedly?

A) setTimeout

B) setInterval

C) setImmediate

D) process.nextTick

👉 Answer: B

---

#### 53. process.nextTick() executes:

A) After timers

B) After I/O

C) Before event loop continues

D) After promises

👉 Answer: C

---

#### 54. Which has highest priority?

A) setTimeout

B) setImmediate

C) Promise.then

D) process.nextTick

👉 Answer: D

---

#### 55. Output?

```
process.nextTick(()=>console.log("A"));
Promise.resolve().then(()=>console.log("B"));
console.log("C");
```

A) C A B

B) A B C

C) C B A

D) B A C

👉 Answer: A

---

#### 56. Which async method is cleanest?

A) Callbacks

B) Promises

C) async/await

D) Timers

👉 Answer: C

---

#### 57. Which assignment demonstrates async behavior best?

A) Printing numbers

B) File read with callback

C) Loop execution

D) Variable assignment

👉 Answer: B

---

#### 58. Which improves Node.js scalability?

A) Blocking code

B) Async I/O

C) CPU loops

D) Synchronous calls

👉 Answer: B

---

#### 59. Which Node.js feature enables high concurrency?

A) Threads

B) Event loop

C) JVM

D) Classes

👉 Answer: B

---

**60. Main advantage of async programming:**

- A) Smaller code
- B) Non-blocking execution
- C) Faster compilation
- D) Easy syntax

👉 Answer: B

## SESSION 16 – NODE.JS MODULES & NPM (WPT – CCEE)

---

### EASY LEVEL (1–20)

**1. A Node.js module is:**

- A) A browser feature
- B) A reusable block of code
- C) A database
- D) A server

👉 Answer: B

---

**2. Node.js uses which module system by default?**

- A) ES Modules
- B) AMD
- C) CommonJS
- D) UMD

👉 Answer: C

---

**3. Which keyword imports a module in Node.js?**

- A) include
- B) import
- C) require
- D) using

👉 Answer: C

---

**4. Which object is used to export data from a module?**

- A) export
- B) exports
- C) module.exports
- D) Both B and C

👉 Answer: D

---

**5. Which is a core Node.js module?**

- A) express
- B) fs
- C) axios
- D) lodash

👉 Answer: B

---

**6. Node.js core modules are:**

- A) Installed via npm
- B) Built into Node.js
- C) Browser-based
- D) Third-party

👉 Answer: B

---

**7. Which file defines project metadata?**

- A) node.json
- B) package.json
- C) module.json
- D) config.js

👉 Answer: B

---

**8. Which file stores exact dependency versions?**

- A) package.json
- B) node\_modules
- C) package-lock.json
- D) npm.lock

👉 Answer: C

---

**9. Which command initializes a Node project?**

- A) npm install
- B) npm start
- C) npm init

D) node init

👉 Answer: C

---

**10. Which folder contains installed npm packages?**

- A) src
- B) lib
- C) node\_modules
- D) packages

👉 Answer: C

---

**11. Which command installs a package locally?**

- A) npm install pkg
- B) npm install -g pkg
- C) npm global pkg
- D) npm add pkg

👉 Answer: A

---

**12. Which command installs a package globally?**

- A) npm install pkg
- B) npm install --save pkg
- C) npm install -g pkg
- D) npm add global pkg

👉 Answer: C

---

**13. Local npm packages are used:**

- A) System-wide
- B) Only in the current project
- C) In browser
- D) In database

👉 Answer: B

---

**14. Global npm packages are mainly used for:**

- A) Project dependencies
- B) CLI tools
- C) Frontend styling
- D) Databases

👉 Answer: B

---

**15. Which command updates packages?**

- A) npm update
- B) npm upgrade
- C) npm refresh
- D) npm sync

👉 Answer: A

---

**16. Which command checks npm version?**

- A) npm --version
- B) node -v
- C) npm version
- D) npm check

👉 Answer: A

---

**17. Which file is mandatory for npm project?**

- A) index.js
- B) package.json
- C) app.js
- D) server.js

👉 Answer: B

---

**18. Which module type starts with `./`?**

- A) Core module
- B) Global module
- C) Local module
- D) Third-party module

👉 Answer: C

---

**19. Node.js module file extension is:**

- A) .node
- B) .npm
- C) .js
- D) .json

👉 Answer: C

---

**20. Modules help mainly in:**

- A) UI design
- B) Code reuse
- C) Faster CPU
- D) Compilation

👉 Answer: B

---

## MEDIUM LEVEL (21–40) — *With Snippets*

### 21. Output?

```
// a.js
module.exports = { x: 10 };
```

What is exported?

- A) Variable x
- B) Object containing x
- C) Function
- D) Nothing

 **Answer: B**

---

### 22. Output?

```
const m = require('./a');
console.log(m.x);
```

- A) undefined
- B) error
- C) 10
- D) null

 **Answer: C**

---

### 23. Output?

```
exports.a = 5;
```

What is exported?

- A) Whole module
- B) Property a
- C) Function a
- D) Nothing

 **Answer: B**

---

### 24. Difference between `exports` and `module.exports` is:

- A) No difference at all
- B) `exports` is a reference to `module.exports`

- C) module.exports is a copy
- D) exports replaces module.exports

👉 Answer: B

---

## 25. Which command installs dependency and saves it?

- A) npm install pkg
- B) npm install pkg --save
- C) npm add pkg
- D) npm get pkg

👉 Answer: B

---

## 26. Output?

```
const fs = require('fs');
```

fs is:

- A) Third-party module
- B) Core module
- C) Local module
- D) Global module

👉 Answer: B

---

## 27. Which section lists dependencies in package.json?

- A) scripts
- B) dependencies
- C) config
- D) meta

👉 Answer: B

---

## 28. Which section lists dev-only dependencies?

- A) dependencies
- B) devDependencies
- C) scripts
- D) optional

👉 Answer: B

---

## 29. Which command installs dev dependency?

- A) npm install pkg
- B) npm install pkg -D
- C) npm install -g pkg

D) npm add dev pkg

👉 Answer: B

---

### 30. Output?

```
console.log(require('./a'));
```

A) Exports of a.js

B) Path of file

C) Error

D) Nothing

👉 Answer: A

---

### 31. Which file should be ignored in git?

A) package.json

B) package-lock.json

C) node\_modules

D) index.js

👉 Answer: C

---

### 32. Which command removes a package?

A) npm delete

B) npm remove pkg

C) npm uninstall pkg

D) npm clear pkg

👉 Answer: C

---

### 33. npm stands for:

A) Node Project Manager

B) Node Package Manager

C) Network Package Manager

D) Node Program Manager

👉 Answer: B

---

### 34. Which command lists installed packages?

A) npm list

B) npm show

C) npm info

D) npm packages

👉 Answer: A

---

**35. What does `main` field in `package.json` specify?**

- A) Entry file
- B) Version
- C) Author
- D) License

👉 Answer: A

---

**36. Output?**

```
require('./test.js');
```

What happens?

- A) File is executed
- B) File is compiled
- C) File is ignored
- D) Error

👉 Answer: A

---

**37. Which module is used for path operations?**

- A) fs
- B) path
- C) os
- D) util

👉 Answer: B

---

**38. Which command installs all dependencies?**

- A) npm start
- B) npm init
- C) npm install
- D) npm update

👉 Answer: C

---

**39. `package-lock.json` ensures:**

- A) Faster UI
- B) Same dependency versions
- C) Smaller bundle
- D) Security only

👉 Answer: B

---

**40. Which type of module is installed via npm?**

- A) Core
- B) Local
- C) Third-party
- D) Built-in

👉 Answer: C

---

 **HARD LEVEL (41–60) — Tricky & Lab-Oriented**

**41. Output?**

```
exports = { a: 10 };
```

Effect?

- A) Exports object
- B) Overrides module.exports
- C) Does NOT export as expected
- D) Error

👉 Answer: C

---

**42. Correct way to export single function:**

- A) exports = func
- B) module.exports = func
- C) export func
- D) require func

👉 Answer: B

---

**43. Output?**

```
console.log(require('./a') === require('./a'));
```

- A) false
- B) true
- C) error
- D) undefined

👉 Answer: B

---

**44. Why above output is true?**

- A) require caches modules
- B) require recompiles

C) Two copies created

D) Global scope

👉 Answer: A

---

**45. Which command installs exact versions from lock file?**

A) npm install

B) npm update

C) npm upgrade

D) npm sync

👉 Answer: A

---

**46. Which field defines CLI command?**

A) scripts

B) bin

C) main

D) exec

👉 Answer: B

---

**47. Which is NOT a valid dependency type?**

A) dependencies

B) devDependencies

C) peerDependencies

D) buildDependencies

👉 Answer: D

---

**48. Which module loads faster?**

A) Core

B) Third-party

C) Local

D) Global

👉 Answer: A

---

**49. Which command installs package globally for CLI use?**

A) npm install pkg

B) npm install -g pkg

C) npm add pkg

D) npm get pkg

👉 Answer: B

---

**50. Which is best practice?**

- A) Commit node\_modules
- B) Ignore node\_modules
- C) Delete package.json
- D) Use globals only

👉 Answer: B

---

**51. Which command runs scripts?**

- A) npm exec
- B) npm run
- C) npm start only
- D) node run

👉 Answer: B

---

**52. Output?**

```
require('fs');
```

fs is resolved from:

- A) node\_modules
- B) Core modules
- C) Local file
- D) Global install

👉 Answer: B

---

**53. Which module system supports `require`?**

- A) ES Modules
- B) CommonJS
- C) AMD
- D) SystemJS

👉 Answer: B

---

**54. Which keyword replaces require in ES modules?**

- A) include
- B) fetch
- C) import
- D) using

👉 Answer: C

---

**55. Which file enables ES modules in Node?**

- A) .node
- B) package.json with type=module
- C) index.mjs only
- D) npm.json

👉 Answer: B

---

**56. Which command checks outdated packages?**

- A) npm old
- B) npm outdated
- C) npm update
- D) npm check

👉 Answer: B

---

**57. Which improves project reproducibility?**

- A) package.json
- B) package-lock.json
- C) node\_modules
- D) globals

👉 Answer: B

---

**58. Which module loads JSON automatically?**

- A) fs
- B) require
- C) path
- D) os

👉 Answer: B

---

**59. require() returns:**

- A) File path
- B) Exports object
- C) Source code
- D) Promise

👉 Answer: B

---

**60. Main benefit of Node modules:**

- A) Faster UI
- B) Code organization & reuse
- C) Multithreading
- D) Compilation

👉 Answer: B



# SESSION 17 – NODE.JS MODULES (`fs` & `http`)

---



## EASY LEVEL (1–20)

1. The `fs` module in Node.js is used for:

- A) Network operations
- B) File system operations
- C) HTTP requests
- D) Database access

👉 Answer: B

---

2. Which module is used to create an HTTP server?

- A) net
- B) url
- C) http
- D) express

👉 Answer: C

---

3. `fs` is a:

- A) Third-party module
- B) Local module
- C) Core Node.js module
- D) Global module

👉 Answer: C

---

4. Which method writes file synchronously?

- A) `writeFile()`
- B) `writeFileAsync()`
- C) `writeFileSync()`
- D) `fs.write()`

👉 Answer: C

---

**5. Which method reads file asynchronously?**

- A) readFileSync()
- B) readSync()
- C) readfile()
- D) openSync()

👉 Answer: C

---

**6. Asynchronous file operations use:**

- A) Return values
- B) Callbacks
- C) Threads
- D) Blocking calls

👉 Answer: B

---

**7. HTTP server listens on:**

- A) URL
- B) IP only
- C) Port
- D) File

👉 Answer: C

---

**8. Which method creates HTTP server?**

- A) http.newServer()
- B) http.createServer()
- C) http.server()
- D) http.listen()

👉 Answer: B

---

**9. Which HTTP method retrieves data?**

- A) POST
- B) PUT
- C) GET
- D) DELETE

👉 Answer: C

---

**10. Which HTTP method sends data to server?**

- A) GET
- B) POST
- C) HEAD
- D) OPTIONS

👉 Answer: B

---

**11. Node.js HTTP server works on:**

- A) Multi-threading
- B) Blocking I/O
- C) Event-driven model
- D) JVM

👉 **Answer: C**

---

**12. Which object represents HTTP request?**

- A) response
- B) req
- C) server
- D) url

👉 **Answer: B**

---

**13. Which object sends response to client?**

- A) req
- B) request
- C) response
- D) server

👉 **Answer: C**

---

**14. Which status code indicates success?**

- A) 404
- B) 500
- C) 200
- D) 301

👉 **Answer: C**

---

**15. fs operations without Sync keyword are:**

- A) Blocking
- B) Asynchronous
- C) Synchronous
- D) Sequential

👉 **Answer: B**

---

**16. HTTP headers are sent using:**

- A) req.send()
- B) res.writeHead()
- C) res.header()

D) http.header()

👉 Answer: B

---

**17. Which method ends HTTP response?**

A) res.write()

B) res.send()

C) res.end()

D) res.close()

👉 Answer: C

---

**18. Node.js web apps can serve:**

A) Only HTML

B) Only JSON

C) Both GET & POST

D) Only files

👉 Answer: C

---

**19. Which port is commonly used for HTTP?**

A) 21

B) 80

C) 443

D) 3000

👉 Answer: D

---

**20. Which is true about async fs methods?**

A) Faster but blocking

B) Slower but blocking

C) Non-blocking

D) Sequential

👉 Answer: C

---

 **MEDIUM LEVEL (21–40) — WITH SNIPPETS**

**21. Output?**

```
const fs = require('fs');
fs.writeFileSync("a.txt", "Hello");
```

Effect?

- A) Reads file
- B) Writes file synchronously
- C) Appends file
- D) Error

👉 Answer: B

---

22. Output?

```
fs.readFile("a.txt", "utf8", (err, data)=>{
  console.log(data);
});
```

This is:

- A) Sync read
- B) Async read
- C) Blocking read
- D) Error

👉 Answer: B

---

23. Output?

```
console.log("A");
fs.readFile("a.txt", ()=>console.log("B"));
console.log("C");
```

Order?

- A) A B C
- B) A C B
- C) B A C
- D) C A B

👉 Answer: B

---

24. Which parameter handles errors in async fs?

- A) data
- B) file
- C) err
- D) result

👉 Answer: C

---

## 25. Output?

```
fs.writeFile("b.txt", "Hi", ()=>console.log("Done"));
```

What happens?

- A) Blocking write
- B) Async write
- C) Error
- D) No write

👉 Answer: B

---

## 26. Which fs method appends data?

- A) writeFile()
- B) appendFile()
- C) open()
- D) write()

👉 Answer: B

---

## 27. Output?

```
const http = require('http');
http.createServer((req, res)=>{
  res.end("Hello");
}).listen(3000);
```

What does this code do?

- A) Creates file
- B) Creates HTTP server
- C) Reads file
- D) Error

👉 Answer: B

---

## 28. Which method reads HTTP request body?

- A) req.body
- B) req.on('data')
- C) req.read()
- D) res.write()

👉 Answer: B

---

## 29. Output?

```
if(req.method === "GET")
```

Checks for:

- A) URL
- B) Request type
- C) Status
- D) Port

👉 Answer: B

---

### 30. Output?

```
res.writeHead(200, {'Content-Type': 'text/plain'});
```

Purpose?

- A) End response
- B) Set headers
- C) Send data
- D) Close server

👉 Answer: B

---

### 31. Which event completes request data read?

- A) data
- B) read
- C) end
- D) finish

👉 Answer: C

---

### 32. Output?

```
req.on('data', chunk => body += chunk);
```

Used for:

- A) Sending response
- B) Reading POST data
- C) Writing file
- D) Closing server

👉 Answer: B

---

### 33. Which fs method checks file existence?

- A) fs.existsSync()

- B) fs.check()
- C) fs.hasFile()
- D) fs.isFile()

👉 Answer: A

---

#### 34. Which fs method deletes file?

- A) remove()
- B) delete()
- C) unlink()
- D) erase()

👉 Answer: C

---

#### 35. Output?

```
fs.unlink("a.txt", ()=>console.log("Deleted"));
```

Effect?

- A) Reads file
- B) Deletes file
- C) Writes file
- D) Error

👉 Answer: B

---

#### 36. HTTP server callback parameters are:

- A) (res,req)
- B) (request,response)
- C) (req,res)
- D) (server,client)

👉 Answer: C

---

#### 37. Which content type is used for JSON?

- A) text/plain
- B) text/html
- C) application/json
- D) application/xml

👉 Answer: C

---

#### 38. Which fs method is blocking?

- A) readfile()
- B) writefile()

- C) `readFileSync()`
- D) `appendFile()`

👉 Answer: C

---

### 39. Node HTTP server is:

- A) Blocking
- B) Event-driven
- C) Multithreaded
- D) Sequential

👉 Answer: B

---

### 40. Which module parses URL data?

- A) `fs`
- B) `http`
- C) `url`
- D) `path`

👉 Answer: C

---

## 🔴 HARD LEVEL (41–60) — LAB & TRICKY

### 41. Output?

```
fs.writeFileSync("x.txt", "A");
fs.readFile("x.txt", "utf8", (e, d)=>console.log(d));
console.log("Done");
```

Order?

- A) A Done
- B) Done A
- C) A only
- D) Error

👉 Answer: B

---

### 42. Why async fs is preferred?

- A) Simpler code
- B) Blocks CPU
- C) Non-blocking I/O
- D) Uses threads

👉 Answer: C

---

#### 43. Output?

```
req.method === "POST"
```

Used to detect:

- A) URL
- B) Header
- C) POST request
- D) Response

👉 Answer: C

---

#### 44. Output?

```
req.url === "/"
```

Checks for:

- A) Port
- B) Root path
- C) Method
- D) Host

👉 Answer: B

---

#### 45. Which HTTP status code for POST success?

- A) 200
- B) 201
- C) 404
- D) 500

👉 Answer: B

---

#### 46. Which fs method creates file if not exists?

- A) readFile
- B) open
- C) writeFile
- D) unlink

👉 Answer: C

---

#### 47. Which order is correct in POST handling?

- A) end → data
- B) data → end
- C) write → read

D) close → data

👉 Answer: B

---

#### 48. Output?

```
let body="";
req.on("data", chunk=>body+=chunk);
req.on("end", ()=>console.log(body));
```

Used for:

- A) GET request
- B) POST body parsing
- C) File read
- D) Header write

👉 Answer: B

---

#### 49. Which fs method returns buffer by default?

- A) readFileSync
- B) readfile
- C) Both A and B
- D) writeFile

👉 Answer: C

---

#### 50. Why encoding is passed in fs.readFile()?

- A) Compress file
- B) Convert buffer to string
- C) Speed up
- D) Secure file

👉 Answer: B

---

#### 51. HTTP server callback executes on:

- A) Every request
- B) Server start
- C) Port bind
- D) Shutdown

👉 Answer: A

---

#### 52. Which makes Node web app scalable?

- A) Blocking I/O
- B) Async fs

C) Multithreading

D) Sync code

👉 Answer: B

---

**53. Which response ends connection?**

A) res.write()

B) res.close()

C) res.end()

D) res.send()

👉 Answer: C

---

**54. Which fs method is deprecated?**

A) fs.exists()

B) fs.readFile()

C) fs.writeFile()

D) fs.unlink()

👉 Answer: A

---

**55. HTTP server handles multiple clients using:**

A) Threads

B) Event loop

C) JVM

D) Forking

👉 Answer: B

---

**56. Which tool tests GET/POST requests?**

A) npm

B) Node REPL

C) Browser / Postman

D) Compiler

👉 Answer: C

---

**57. Which fs method opens file descriptor?**

A) open()

B) readFile()

C) writeFile()

D) close()

👉 Answer: A

---

**58. Which is correct Node HTTP flow?**

- A) Request → Process → Response
- B) Response → Request → Process
- C) Process → Request → Response
- D) Request → Response → Process

👉 Answer: A

---

**59. Which makes fs sync dangerous in server?**

- A) Faster
- B) Blocks event loop
- C) Uses memory
- D) Needs encoding

👉 Answer: B

---

**60. Best practice for Node web apps:**

- A) Use sync fs
- B) Use async fs & HTTP
- C) Avoid modules
- D) Use blocking calls

👉 Answer: B

## SESSION 18 – INTRODUCTION TO EXPRESS (WPT – CCEE)

---

### EASY LEVEL (1–20)

**1. Express is:**

- A) A programming language
- B) A Node.js web framework
- C) A database
- D) A browser

👉 Answer: B

---

**2. Express is built on top of:**

- A) Java
- B) PHP
- C) Node.js

D) Python

👉 Answer: C

---

**3. Express is mainly used for:**

- A) Styling pages
- B) Building web applications and APIs
- C) Database management
- D) UI design

👉 Answer: B

---

**4. Which command installs Express?**

- A) node install express
- B) npm get express
- C) npm install express
- D) npm add node

👉 Answer: C

---

**5. Which function creates an Express application?**

- A) express.new()
- B) express.create()
- C) express()
- D) new express()

👉 Answer: C

---

**6. Express applications run on:**

- A) Browser
- B) Server
- C) Database
- D) OS kernel

👉 Answer: B

---

**7. Which object represents HTTP request in Express?**

- A) response
- B) server
- C) req
- D) app

👉 Answer: C

---

**8. Which object is used to send response to client?**

- A) req
- B) res
- C) app
- D) router

👉 Answer: B

---

**9. Which method defines a GET route?**

- A) app.post()
- B) app.route()
- C) app.get()
- D) app.send()

👉 Answer: C

---

**10. Which method starts Express server?**

- A) app.run()
- B) app.listen()
- C) app.start()
- D) server.listen()

👉 Answer: B

---

**11. Express uses which protocol?**

- A) FTP
- B) SMTP
- C) HTTP
- D) TCP only

👉 Answer: C

---

**12. Which port is commonly used in Express apps?**

- A) 21
- B) 80
- C) 443
- D) 3000

👉 Answer: D

---

**13. Express response is sent using:**

- A) res.write()
- B) res.send()
- C) res.push()
- D) res.end()

👉 Answer: B

---

**14. Which method sends JSON response?**

- A) res.text()
- B) res.html()
- C) res.json()
- D) res.sendFile()

👉 Answer: C

---

**15. Which method sends HTML file?**

- A) res.send()
- B) res.file()
- C) res.sendFile()
- D) res.html()

👉 Answer: C

---

**16. Express supports:**

- A) Only GET
- B) Only POST
- C) GET, POST, PUT, DELETE
- D) Only HTTP/1.0

👉 Answer: C

---

**17. Express routes map:**

- A) Database to UI
- B) URL to handler
- C) File to server
- D) Module to module

👉 Answer: B

---

**18. Middleware is a function that:**

- A) Handles database
- B) Executes between request and response
- C) Styles HTML
- D) Starts server

👉 Answer: B

---

**19. Express simplifies Node.js by:**

- A) Adding blocking code
- B) Providing routing & middleware
- C) Removing HTTP

D) Replacing JavaScript

👉 Answer: B

---

## 20. Express is:

- A) Opinionated framework
- B) Minimal and flexible framework
- C) Database ORM
- D) Frontend framework

👉 Answer: B

---

## 🟡 MEDIUM LEVEL (21–40) — WITH SNIPPETS

### 21. Output?

```
const express = require('express');
const app = express();
```

What does this code do?

- A) Creates server
- B) Creates Express application
- C) Starts server
- D) Handles request

👉 Answer: B

---

### 22. Output?

```
app.get('/', (req, res)=>{
  res.send("Hello");
});
```

This code defines:

- A) POST route
- B) Root GET route
- C) Middleware
- D) Error handler

👉 Answer: B

---

### 23. Output?

```
app.listen(3000);
```

Effect?

- A) Stops server
- B) Starts server on port 3000
- C) Creates route
- D) Creates middleware

👉 Answer: B

---

**24. Which object contains query parameters?**

- A) req.body
- B) req.query
- C) req.params
- D) req.header

👉 Answer: B

---

**25. Which object contains route parameters?**

- A) req.body
- B) req.query
- C) req.params
- D) req.route

👉 Answer: C

---

**26. Output?**

```
app.post('/login',(req,res)=>{});
```

This route handles:

- A) GET request
- B) POST request
- C) PUT request
- D) DELETE request

👉 Answer: B

---

**27. Which middleware parses JSON body?**

- A) express.urlencoded()
- B) express.json()
- C) bodyParser.text()
- D) res.json()

👉 Answer: B

---

## 28. Output?

```
app.use(express.json());
```

Purpose?

- A) Send JSON
- B) Parse JSON request body
- C) Render template
- D) Handle errors

👉 Answer: B

---

## 29. Middleware function signature is:

- A) (req)
- B) (res)
- C) (req,res)
- D) (req,res,next)

👉 Answer: D

---

## 30. Output?

```
app.use((req, res, next)=>{  
  console.log("Hi");  
  next();  
});
```

This middleware:

- A) Ends request
- B) Logs and passes control
- C) Sends response
- D) Throws error

👉 Answer: B

---

## 31. What happens if `next()` is not called?

- A) Request continues
- B) Response sent automatically
- C) Request hangs
- D) Server crashes

👉 Answer: C

---

**32. Which method sends status code?**

- A) res.code()
- B) res.status()
- C) res.sendStatus()
- D) Both B and C

👉 **Answer: D**

---

**33. Output?**

```
res.status(404).send("Not Found");
```

Effect?

- A) Success
- B) Redirect
- C) Client error response
- D) Server crash

👉 **Answer: C**

---

**34. Which template engine is commonly used with Express?**

- A) JSP
- B) Thymeleaf
- C) EJS
- D) Angular

👉 **Answer: C**

---

**35. Which command installs EJS?**

- A) npm get ejs
- B) npm install ejs
- C) npm add template
- D) node install ejs

👉 **Answer: B**

---

**36. Which method renders a view?**

- A) res.send()
- B) res.render()
- C) res.view()
- D) app.render()

👉 **Answer: B**

---

**37. Output?**

```
res.render('home', {name: 'Amol'});
```

Purpose?

- A) Send JSON
- B) Render template with data
- C) Redirect
- D) Serve file

👉 Answer: B

---

**38. Express router is used to:**

- A) Connect database
- B) Organize routes
- C) Style UI
- D) Create server

👉 Answer: B

---

**39. Which method creates router?**

- A) express.newRouter()
- B) express.Router()
- C) app.router()
- D) new Router()

👉 Answer: B

---

**40. Express follows which architecture?**

- A) Monolithic
- B) MVC (commonly)
- C) Peer-to-peer
- D) Event sourcing

👉 Answer: B

---

## 🔴 HARD LEVEL (41–60) — LAB & TRICKY

**41. Output?**

```
app.get('/user/:id', (req, res)=>{  
  res.send(req.params.id);  
});
```

URL `/user/10` returns:

- A) user

- B) id
  - C) 10
  - D) undefined
- 👉 Answer: C
- 

#### 42. Output?

```
app.use('/admin', (req, res)=>{  
    res.send("Admin");  
});
```

Accessed via:

- A) /
  - B) /user
  - C) /admin
  - D) /login
- 👉 Answer: C
- 

#### 43. Difference between app.use and app.get:

- A) No difference
- B) app.use works for all HTTP methods
- C) app.get handles POST
- D) app.use is deprecated

👉 Answer: B

---

#### 44. Middleware order matters because:

- A) Express is synchronous
- B) Middleware executes in sequence
- C) Routes override middleware
- D) Server crashes otherwise

👉 Answer: B

---

#### 45. Output?

```
app.use((req, res, next)=>{  
    res.send("Hello");  
});  
app.get('/', (req, res)=>{  
    res.send("Home");  
});
```

Request to `/` returns:

- A) Home
- B) Hello
- C) Error
- D) Nothing

👉 Answer: B

---

**46. Which middleware handles errors?**

- A) `(req,res,next)`
- B) `(err,req,res,next)`
- C) `(req,err,res)`
- D) `(error)`

👉 Answer: B

---

**47. Which is best practice for large Express apps?**

- A) Single file
- B) No middleware
- C) Use routers and modules
- D) Global variables

👉 Answer: C

---

**48. Express is preferred over plain Node HTTP because:**

- A) Faster CPU
- B) Simplified routing & middleware
- C) Multithreading
- D) Database support

👉 Answer: B

---

**49. Which method redirects client?**

- A) `res.send()`
- B) `res.forward()`
- C) `res.redirect()`
- D) `res.route()`

👉 Answer: C

---

**50. Output?**

```
res.redirect('/login');
```

Effect?

- A) Sends HTML
- B) Forwards internally
- C) Client redirect
- D) Server crash

👉 Answer: C

---

51. Which status code is default for `res.send()`?

- A) 404
- B) 500
- C) 200
- D) 301

👉 Answer: C

---

52. Express template engines help to:

- A) Compile JS
- B) Generate dynamic HTML
- C) Create APIs
- D) Handle database

👉 Answer: B

---

53. Which command starts Express app normally?

- A) npm run dev
- B) node app.js
- C) express start
- D) npm express

👉 Answer: B

---

54. Rebuilding Node assignment using Express improves:

- A) Complexity
- B) Code readability & structure
- C) CPU usage
- D) Memory usage

👉 Answer: B

---

55. Which Express feature replaces manual URL parsing?

- A) Routing
- B) fs module
- C) path module
- D) HTTP headers

👉 Answer: A

---

**56. Which Express object is singleton?**

- A) req
- B) res
- C) app
- D) router

👉 Answer: C

---

**57. Express internally uses which Node module?**

- A) fs
- B) http
- C) net
- D) path

👉 Answer: B

---

**58. Which middleware serves static files?**

- A) express.static()
- B) express.files()
- C) express.public()
- D) app.static()

👉 Answer: A

---

**59. Which folder commonly stores views?**

- A) src
- B) views
- C) templates
- D) html

👉 Answer: B

---

**60. Express improves productivity mainly by:**

- A) Removing JavaScript
- B) Abstracting low-level HTTP handling
- C) Blocking I/O
- D) Using threads

👉 Answer: B



# SESSION 19 – INTRODUCTION TO REACT (WPT – CCEE)

---



## EASY LEVEL (1–20)

**1. React is:**

- A) A programming language
- B) A JavaScript library
- C) A database
- D) A web server

👉 Answer: B

---

**2. React is mainly used for:**

- A) Backend development
- B) Building user interfaces
- C) Database management
- D) Server configuration

👉 Answer: B

---

**3. React was developed by:**

- A) Google
- B) Microsoft
- C) Facebook (Meta)
- D) Amazon

👉 Answer: C

---

**4. React applications are built using:**

- A) Templates
- B) Components
- C) Controllers
- D) Modules only

👉 Answer: B

---

**5. A React element is:**

- A) A real DOM node
- B) A virtual DOM object
- C) A CSS rule

D) A JavaScript class

👉 Answer: B

---

**6. Components in React are:**

- A) HTML files
- B) Reusable UI pieces
- C) Database entities
- D) Server routes

👉 Answer: B

---

**7. Which file extension is commonly used for React components?**

- A) .java
- B) .html
- C) .jsx
- D) .json

👉 Answer: C

---

**8. Props in React are used to:**

- A) Store component state
- B) Pass data to components
- C) Handle events
- D) Access DOM

👉 Answer: B

---

**9. Props are:**

- A) Mutable
- B) Immutable (read-only)
- C) Global
- D) Optional

👉 Answer: B

---

**10. React supports which type of components?**

- A) Function only
- B) Class only
- C) Function and Class
- D) Module only

👉 Answer: C

---

**11. Function components are written using:**

- A) class keyword
- B) function or arrow function
- C) interface
- D) object

👉 **Answer: B**

---

**12. Class components must extend:**

- A) Component
- B) React.Component
- C) React.Class
- D) BaseComponent

👉 **Answer: B**

---

**13. Which method renders JSX to the DOM?**

- A) React.render()
- B) ReactDOM.render()
- C) renderDOM()
- D) DOM.render()

👉 **Answer: B**

---

**14. JSX stands for:**

- A) Java Syntax XML
- B) JavaScript XML
- C) JSON XML
- D) Java Styled XML

👉 **Answer: B**

---

**15. JSX is:**

- A) Mandatory
- B) Optional
- C) Replaced HTML
- D) Server-side only

👉 **Answer: B**

---

**16. React uses which DOM for performance?**

- A) Real DOM
- B) Shadow DOM
- C) Virtual DOM
- D) Browser DOM

👉 **Answer: C**

---

**17. React components must return:**

- A) HTML string
- B) One root element
- C) Multiple root elements
- D) JSON

👉 Answer: B

---

**18. Data flow in React is:**

- A) Two-way
- B) Circular
- C) One-way (parent → child)
- D) Random

👉 Answer: C

---

**19. Which command creates a React app (classic)?**

- A) npm react-app
- B) npx create-react-app
- C) npm init react
- D) react new

👉 Answer: B

---

**20. React is best suited for:**

- A) Static pages
- B) Dynamic and SPA applications
- C) CLI tools
- D) Databases

👉 Answer: B

---

## 🟡 MEDIUM LEVEL (21–40) — WITH SNIPPETS

**21. Output?**

```
function Hello() {  
  return <h1>Hello</h1>;  
}
```

This is an example of:

- A) Class component
- B) Function component

C) React element

D) Hook

👉 Answer: B

---

## 22. Output?

```
const Welcome = (props) => <h1>{props.name}</h1>;
```

`props.name` represents:

- A) Component state
- B) Passed data
- C) Local variable
- D) Event

👉 Answer: B

---

## 23. Output?

```
<Welcome name="Amol" />
```

What is passed to component?

- A) State
- B) Event
- C) Props
- D) DOM

👉 Answer: C

---

## 24. Output?

```
function App() {  
  return <Hello />;  
}
```

This is an example of:

- A) Rendering component
- B) Decluttering component
- C) Using props
- D) Using state

👉 Answer: A

---

## 25. Which statement is correct about props?

- A) Props can be changed
- B) Props are immutable
- C) Props are optional
- D) Props are global

👉 Answer: B

---

## 26. Output?

```
class Hello extends React.Component {  
  render() {  
    return <h1>Hello</h1>;  
  }  
}
```

This is:

- A) Function component
- B) Class component
- C) Hook
- D) Element

👉 Answer: B

---

## 27. What does component composition mean?

- A) Styling components
- B) Nesting components inside others
- C) Removing components
- D) Creating CSS

👉 Answer: B

---

## 28. Output?

```
function App() {  
  return (  
    <>  
      <Header />  
      <Footer />  
    </>  
  );  
}
```

This shows:

- A) Decluttering
- B) Composition
- C) Props
- D) State

👉 Answer: B

---

### 29. Which symbol is used to embed JavaScript in JSX?

- A) ()
- B) []
- C) {}
- D) <>

👉 Answer: C

---

### 30. Output?

```
<h1>{2 + 3}</h1>
```

Rendered output:

- A) {2+3}
- B) 23
- C) 5
- D) Error

👉 Answer: C

---

### 31. Which React feature helps split UI into small parts?

- A) Modules
- B) Components
- C) Templates
- D) Controllers

👉 Answer: B

---

### 32. Which React method updates UI efficiently?

- A) Direct DOM update
- B) Virtual DOM diffing
- C) Reload page
- D) Server rendering

👉 Answer: B

---

---

**33. Decluttering components means:**

- A) Writing everything in one file
- B) Breaking UI into smaller components
- C) Removing JSX
- D) Avoiding props

👉 Answer: B

---

**34. Which is NOT valid JSX?**

- A) <h1>Hello</h1>
- B) <div></div>
- C) <h1>Hello>
- D) <App />

👉 Answer: C

---

**35. React.Fragment is used to:**

- A) Add extra div
- B) Return multiple elements without extra DOM
- C) Style component
- D) Handle events

👉 Answer: B

---

**36. Output?**

```
<>
  <h1>A</h1>
  <h2>B</h2>
</>
```

This renders:

- A) Error
- B) Two elements
- C) One element
- D) Nothing

👉 Answer: B

---

**37. Which method is compulsory in class component?**

- A) constructor()
- B) componentDidMount()
- C) render()
- D) setState()

👉 Answer: C

---

**38. Which file usually renders root React component?**

- A) App.js
- B) index.js
- C) main.js
- D) root.js

👉 Answer: B

---

**39. Which React concept helps reuse UI logic?**

- A) Props
- B) Components
- C) CSS
- D) HTML

👉 Answer: B

---

**40. React Clock app mainly uses:**

- A) Props only
- B) State only
- C) Time APIs + components
- D) Database

👉 Answer: C

---

## 🔴 HARD LEVEL (41–60) — LAB & TRICKY

**41. Output?**

```
function Show(props){  
  return <p>{props.msg}</p>;  
}  
<Show msg="Hi" />
```

Rendered output:

- A) props.msg
- B) Hi
- C) Show
- D) Error

👉 Answer: B

---

**42. Why React prefers components?**

- A) Faster CPU

B) Code reusability and maintainability

C) Database integration

D) Styling

👉 Answer: B

---

#### 43. Which is TRUE about function vs class components?

A) Only class components support props

B) Function components are simpler

C) Class components are deprecated

D) Function components cannot render JSX

👉 Answer: B

---

#### 44. Output?

```
const A = () => <B />;
const B = () => <h1>Hello</h1>;
```

This demonstrates:

A) Props

B) State

C) Component composition

D) Hooks

👉 Answer: C

---

#### 45. Which rule must React components follow?

A) Must use JSX

B) Must start with uppercase letter

C) Must use class

D) Must use state

👉 Answer: B

---

#### 46. What happens if component name starts with lowercase?

A) Treated as React component

B) Treated as HTML tag

C) Compilation error

D) Runtime error

👉 Answer: B

---

#### 47. Which is best practice in React?

A) Large single component

B) Many small reusable components

C) No props

D) Direct DOM manipulation

👉 Answer: B

---

**48. Rebuilding HTML app using React improves:**

A) Page size

B) Code organization

C) Database access

D) CPU speed

👉 Answer: B

---

**49. Which feature makes React fast?**

A) Server rendering

B) Virtual DOM

C) Browser caching

D) Multithreading

👉 Answer: B

---

**50. Output?**

```
function App(){
  return <h1>{true && "Hello"}</h1>;
}
```

Rendered output:

A) true

B) Hello

C) false

D) Error

👉 Answer: B

---

**51. Which React concept enables passing data down tree?**

A) Events

B) Props

C) DOM

D) CSS

👉 Answer: B

---

**52. Which tool bundles React apps?**

- A) npm
- B) Webpack
- C) Node
- D) Babel only

👉 Answer: B

---

**53. Which API gives current time in JS (Clock app)?**

- A) Date.now()
- B) new Date()
- C) Both A and B
- D) Time()

👉 Answer: C

---

**54. Showing time of 3 countries mainly needs:**

- A) Database
- B) Timezone conversion
- C) HTTP server
- D) Redux

👉 Answer: B

---

**55. Which React feature avoids full page reload?**

- A) Routing
- B) Virtual DOM updates
- C) Templates
- D) HTTP

👉 Answer: B

---

**56. Which library renders React to DOM?**

- A) React
- B) ReactDOM
- C) ReactNative
- D) Redux

👉 Answer: B

---

**57. Which concept helps split UI into Header/Footer?**

- A) Props
- B) Composition
- C) State
- D) Hooks

👉 Answer: B

---

**58. React elements are:**

- A) Mutable
- B) Immutable
- C) Editable
- D) Stateful

👉 **Answer: B**

---

**59. React follows which architecture?**

- A) MVC
- B) MVVM
- C) Component-based
- D) Layered

👉 **Answer: C**

---

**60. Main advantage of React for labs:**

- A) Faster typing
- B) Reusable & structured UI
- C) Database support
- D) CLI tools

👉 **Answer: B**

---

## SESSIONS 20, 21 & 22 – REACT (STATE, LIFECYCLE & EVENTS)

---

### EASY LEVEL (1–25)

**1. State in React is used to:**

- A) Pass data to child components
- B) Store dynamic data in a component
- C) Style components
- D) Handle routing

👉 **Answer: B**

---

**2. Props are:**

- A) Mutable

- B) Immutable
- C) Local to component only
- D) Optional always

👉 Answer: B

---

### 3. State belongs to:

- A) Parent component only
- B) Child component only
- C) The component itself
- D) Global scope

👉 Answer: C

---

### 4. Which hook is used to add state in function components?

- A) useEffect
- B) useState
- C) useContext
- D) useReducer

👉 Answer: B

---

### 5. Lifecycle methods are mainly used in:

- A) Function components only
- B) Class components
- C) JSX
- D) Props

👉 Answer: B

---

### 6. Which lifecycle method runs after component mounts?

- A) render()
- B) constructor()
- C) componentDidMount()
- D) componentWillUnmount()

👉 Answer: C

---

### 7. Updating state causes:

- A) Page reload
- B) Re-render of component
- C) Error
- D) DOM removal

👉 Answer: B

---

**8. Which method updates state in class component?**

- A) updateState()
- B) setState()
- C) changeState()
- D) modifyState()

👉 Answer: B

---

**9. Handling click events in React uses:**

- A) onclick
- B) onClick
- C) on-click
- D) click

👉 Answer: B

---

**10. React events are:**

- A) Native DOM events
- B) Synthetic events
- C) Browser-only
- D) Server-side

👉 Answer: B

---

**11. Conditional rendering means:**

- A) Rendering always
- B) Rendering based on condition
- C) Styling conditionally
- D) Removing JSX

👉 Answer: B

---

**12. Which operator is commonly used for conditional rendering?**

- A) +
- B) &&
- C) =
- D) ++

👉 Answer: B

---

**13. Context is used to:**

- A) Store local state
- B) Pass data deeply without props drilling
- C) Handle events
- D) Style components

👉 Answer: B

---

**14. Which hook provides lifecycle-like behavior in function components?**

- A) useState
- B) useEffect
- C) useMemo
- D) useRef

👉 Answer: B

---

**15. componentWillMount() is used to:**

- A) Create component
- B) Update state
- C) Cleanup resources
- D) Render JSX

👉 Answer: C

---

**16. Clock app time updates require:**

- A) Props
- B) State
- C) Context
- D) Redux

👉 Answer: B

---

**17. Which JavaScript API provides current time?**

- A) Time()
- B) Date.now()
- C) new Date()
- D) Both B and C

👉 Answer: D

---

**18. Which React feature avoids full page reload?**

- A) Routing
- B) State update
- C) Virtual DOM
- D) useEffect

👉 Answer: C

---

**19. State updates in React are:**

- A) Immediate
- B) Asynchronous
- C) Blocking

D) Sequential always

👉 Answer: B

---

**20. Close button in Clock app will:**

- A) Reload page
- B) Remove component
- C) Stop server
- D) Clear state only

👉 Answer: B

---

**21. Background color based on AM/PM uses:**

- A) Routing
- B) Conditional rendering
- C) Context
- D) Refs

👉 Answer: B

---

**22. React follows data flow:**

- A) Two-way
- B) One-way
- C) Circular
- D) Random

👉 Answer: B

---

**23. setState() should NOT be called in:**

- A) render()
- B) componentDidMount()
- C) Event handler
- D) componentDidUpdate()

👉 Answer: A

---

**24. Which lifecycle method is called before component removal?**

- A) componentDidMount()
- B) componentWillUnmount()
- C) shouldComponentUpdate()
- D) render()

👉 Answer: B

---

## 25. React Context helps avoid:

- A) State
- B) Hooks
- C) Props drilling
- D) JSX

👉 Answer: C

---

## 🟡 MEDIUM LEVEL (26–50) — WITH SNIPPETS

### 26. Output?

```
const [count, setCount] = useState(0);
```

count represents:

- A) Props
- B) State value
- C) Context
- D) Ref

👉 Answer: B

---

### 27. Output?

```
setCount(count + 1);
```

Effect?

- A) Direct DOM update
- B) State update & re-render
- C) Page reload
- D) Error

👉 Answer: B

---

### 28. Output?

```
<button onClick={handleClick}>Click</button>
```

This is:

- A) Invalid syntax
- B) Event handling
- C) Conditional rendering
- D) State update

👉 Answer: B

---

## 29. Output?

```
{isLoggedIn && <h1>Welcome</h1>}
```

Renders `<h1>` when:

- A) isLoggedIn is false
- B) isLoggedIn is true
- C) Always
- D) Never

👉 Answer: B

---

## 30. Output?

```
useEffect(() => {
  console.log("Mounted");
}, []);
```

Runs:

- A) On every render
- B) Once on mount
- C) On unmount
- D) On state change

👉 Answer: B

---

## 31. Output?

```
useEffect(() => {
  return () => console.log("Cleanup");
}, []);
```

Cleanup runs on:

- A) Mount
- B) Render
- C) Unmount
- D) State update

👉 Answer: C

---

## 32. Output?

```
this.setState({ time: new Date() });
```

Used in:

- A) Function component
- B) Class component
- C) JSX
- D) Context

👉 Answer: B

---

### 33. Which lifecycle method is ideal for timers?

- A) render()
- B) componentDidMount()
- C) componentWillUnmount()
- D) constructor()

👉 Answer: B

---

### 34. Output?

```
clearInterval(this.timer);
```

Used in:

- A) componentDidMount()
- B) render()
- C) componentWillUnmount()
- D) constructor()

👉 Answer: C

---

### 35. Output?

```
onClick={() => removeClock(id)}
```

Purpose?

- A) Styling
- B) Event handling with parameter
- C) Conditional rendering
- D) Context usage

👉 Answer: B

---

### 36. Which hook replaces lifecycle methods mostly?

- A) useState

- B) useEffect
- C) useContext
- D) useRef

👉 Answer: B

---

### 37. Output?

```
{hours < 12 ? "AM" : "PM"}
```

This is example of:

- A) Props
- B) State
- C) Conditional rendering
- D) Context

👉 Answer: C

---

### 38. Which style changes background color dynamically?

- A) Inline style with condition
- B) External CSS only
- C) Context API
- D) Redux

👉 Answer: A

---

### 39. Output?

```
<div style={{ backgroundColor: isAM ? "blue" : "black" }} />
```

Used for:

- A) Event handling
- B) Conditional styling
- C) Routing
- D) Context

👉 Answer: B

---

### 40. Which React feature triggers UI update?

- A) Props change
- B) State change
- C) Context change
- D) All of the above

👉 Answer: D

---

#### 41. Output?

```
this.state = { time: new Date() };
```

Used inside:

- A) render()
- B) constructor()
- C) componentDidMount()
- D) componentDidUpdate()

👉 Answer: B

---

#### 42. Why should state not be modified directly?

- A) Causes error
- B) No re-render
- C) Security issue
- D) Syntax issue

👉 Answer: B

---

#### 43. Which is correct way to update state based on previous state?

- A) setState(value)
- B) setState(prev => prev + 1)
- C) state++
- D) state = new

👉 Answer: B

---

#### 44. Output?

```
this.setState(prev => ({ count: prev.count + 1 }));
```

Purpose?

- A) Direct mutation
- B) Safe state update
- C) Styling
- D) Context

👉 Answer: B

---

#### 45. Which React hook is used for global data?

- A) useState
- B) useContext

C) useEffect

D) useRef

👉 Answer: B

---

#### 46. Handling multiple clocks requires:

A) One component

B) Multiple component instances

C) One state only

D) No state

👉 Answer: B

---

#### 47. Which React feature allows removing clock component?

A) Conditional rendering

B) Routing

C) Context

D) Fragment

👉 Answer: A

---

#### 48. Output?

```
{showClock && <Clock />}
```

Clock renders when:

A) showClock false

B) showClock true

C) Always

D) Never

👉 Answer: B

---

#### 49. Which lifecycle method is deprecated?

A) componentDidMount

B) componentWillMount

C) render

D) componentWillUnmount

👉 Answer: B

---

#### 50. Which approach is preferred today?

A) Class lifecycle methods

B) Function components with hooks

C) Inline JS only

D) No state

👉 Answer: B

---

## 🔴 HARD LEVEL (51–75) — LAB & TRICKY

### 51. Output?

```
useEffect(() => {
  const id = setInterval(updateTime, 1000);
  return () => clearInterval(id);
}, []);
```

Purpose?

- A) One-time render
- B) Timer setup & cleanup
- C) Styling
- D) Context

👉 Answer: B

---

### 52. Why cleanup is important in Clock app?

- A) Styling
- B) Prevent memory leaks
- C) Faster rendering
- D) Reduce props

👉 Answer: B

---

### 53. Output?

```
<button onClick={() => setShow(false)}>X</button>
```

Used for:

- A) Add clock
- B) Close/remove clock
- C) Update time
- D) Change props

👉 Answer: B

---

### 54. Which is correct AM/PM logic?

- A) hours > 12
- B) hours >= 12

C) hours == 12

D) hours < 24

👉 Answer: B

---

#### 55. Which state structure best for multiple clocks?

A) Single object

B) Array of clock objects

C) Global variable

D) Props only

👉 Answer: B

---

#### 56. Output?

```
clocks.map(clock => <Clock key={clock.id} />)
```

Purpose?

A) Conditional render

B) List rendering

C) Event handling

D) Context

👉 Answer: B

---

#### 57. Why **key** prop is important?

A) Styling

B) Efficient reconciliation

C) Routing

D) Context

👉 Answer: B

---

#### 58. Which lifecycle hook runs on every update?

A) componentDidMount

B) componentDidUpdate

C) componentWillUnmount

D) constructor

👉 Answer: B

---

#### 59. What happens if setInterval is not cleared?

A) Error

B) Memory leak

C) Faster app

D) No effect

👉 Answer: B

---

**60. Best way to handle events in React is:**

- A) Inline HTML events
- B) Arrow functions or handlers
- C) addEventListener
- D) jQuery

👉 Answer: B

---

**61. Context is best for:**

- A) Small local state
- B) Deeply shared data
- C) Temporary UI state
- D) Styling

👉 Answer: B

---

**62. Which causes re-render?**

- A) Variable change
- B) State update
- C) Console.log
- D) Ref update

👉 Answer: B

---

**63. Output?**

```
useEffect(() => {
  console.log("update");
});
```

Runs when?

- A) Once
- B) On mount only
- C) On every render
- D) On unmount

👉 Answer: C

---

**64. Why props should not be stored in state?**

- A) Causes duplication
- B) Syntax error

C) Performance issue

D) Styling issue

👉 Answer: A

---

**65. Which pattern removes clock cleanly?**

A) CSS hide

B) Conditional render false

C) Direct DOM remove

D) Reload page

👉 Answer: B

---

**66. React lifecycle is mainly about:**

A) Styling

B) Component creation, update, destruction

C) Routing

D) API calls

👉 Answer: B

---

**67. Which hook mimics componentDidMount + Update?**

A) useState

B) useEffect without deps

C) useEffect with []

D) useContext

👉 Answer: B

---

**68. Which approach is NOT recommended?**

A) Multiple small components

B) State-driven UI

C) Direct DOM manipulation

D) Hooks

👉 Answer: C

---

**69. Which React feature ensures UI consistency?**

A) Virtual DOM

B) State management

C) Context

D) Routing

👉 Answer: A

---

**70. Clock background color logic should be placed in:**

- A) JSX inline style
- B) CSS only
- C) Server
- D) Context

👉 Answer: A

---

**71. Which React concept replaces many lifecycle methods?**

- A) Props
- B) Hooks
- C) JSX
- D) Fragment

👉 Answer: B

---

**72. Output?**

```
const isAM = hours < 12;
```

Used for:

- A) Routing
- B) Conditional logic
- C) Event handling
- D) State update

👉 Answer: B

---

**73. Why React state updates are batched?**

- A) Styling
- B) Performance optimization
- C) Security
- D) Simplicity

👉 Answer: B

---

**74. Which improves scalability of React app?**

- A) Large components
- B) Component-based architecture
- C) Inline JS only
- D) No state

👉 Answer: B

---

**75. Main learning objective of this lab is:**

- A) JSX syntax
- B) State, lifecycle, and events mastery
- C) CSS design
- D) Database integration

👉 Answer: B

---

## SESSIONS 23 & 24 – REACT (LISTS, FORMS, REFS, LIFTING STATE)

---

### EASY LEVEL (1–20)

**1. Lists in React are commonly rendered using:**

- A) for loop only
- B) map()
- C) filter()
- D) reduce()

👉 Answer: B

---

**2. Keys in React lists help to:**

- A) Style elements
- B) Identify elements uniquely
- C) Handle events
- D) Store state

👉 Answer: B

---

**3. Keys should be:**

- A) Random
- B) Index always
- C) Unique and stable
- D) Optional

👉 Answer: C

---

**4. Rendering multiple components means:**

- A) Multiple pages
- B) Rendering list of components
- C) Multiple servers

D) Multiple states

👉 Answer: B

---

**5. A basic list component returns:**

- A) Single element
- B) Multiple elements via map
- C) JSON
- D) CSS

👉 Answer: B

---

**6. Forms in React are used to:**

- A) Render lists
- B) Collect user input
- C) Style UI
- D) Handle routing

👉 Answer: B

---

**7. Controlled components mean:**

- A) DOM controls state
- B) React controls input value
- C) No state used
- D) Only refs used

👉 Answer: B

---

**8. Input value in controlled component comes from:**

- A) DOM
- B) Props only
- C) State
- D) Context

👉 Answer: C

---

**9. Which event handles input change?**

- A) onInput
- B) onChange
- C) onClick
- D) onSubmit

👉 Answer: B

---

**10. Refs are used to:**

- A) Pass props
- B) Directly access DOM elements
- C) Manage state
- D) Render lists

👉 Answer: B

---

**11. Which hook creates refs in function components?**

- A) useState
- B) useEffect
- C) useRef
- D) useContext

👉 Answer: C

---

**12. Lifting state up means:**

- A) Moving state to child
- B) Moving state to parent
- C) Removing state
- D) Global state

👉 Answer: B

---

**13. Why lift state up?**

- A) Styling
- B) Share state between components
- C) Performance
- D) Routing

👉 Answer: B

---

**14. Dropdown in React is created using:**

- A) <input>
- B) <select>
- C) <option> only
- D) <list>

👉 Answer: B

---

**15. “Add” button click is handled using:**

- A) onAdd
- B) onClick
- C) onSubmit
- D) onPress

👉 Answer: B

---

**16. Clock list rendering needs:**

- A) State array
- B) Single variable
- C) Context only
- D) Ref only

👉 Answer: A

---

**17. Each rendered Clock component should have:**

- A) Style
- B) Key prop
- C) Ref
- D) Context

👉 Answer: B

---

**18. Forms submission default behavior should be:**

- A) Allowed
- B) Prevented
- C) Ignored
- D) Reload page

👉 Answer: B

---

**19. Which method prevents form reload?**

- A) stop()
- B) preventDefault()
- C) return false
- D) cancel()

👉 Answer: B

---

**20. Selecting country timezone affects:**

- A) Props
- B) State
- C) Context
- D) DOM only

👉 Answer: B

---



## MEDIUM LEVEL (21–40) — WITH SNIPPETS

**21. Output?**

```
countries.map(c => <Clock key={c.id} />)
```

Purpose?

- A) Conditional render
- B) List rendering
- C) Event handling
- D) Styling

👉 Answer: B

---

## 22. Why not use index as key?

- A) Syntax error
- B) Causes rendering issues
- C) Slower app
- D) Not allowed

👉 Answer: B

---

## 23. Output?

```
<input value={name} onChange={e => setName(e.target.value)} />
```

This is:

- A) Uncontrolled input
- B) Controlled input
- C) Ref-based input
- D) Invalid

👉 Answer: B

---

## 24. Output?

```
<form onSubmit={handleSubmit}>
```

Purpose?

- A) Click handling
- B) Form submission handling
- C) List rendering
- D) Routing

👉 Answer: B

---

## 25. Output?

```
e.preventDefault();
```

Used to:

- A) Stop event
- B) Prevent page reload
- C) Stop rendering
- D) Clear state

👉 Answer: B

---

## 26. Output?

```
const inputRef = useRef(null);
```

Creates:

- A) State
- B) Context
- C) Ref object
- D) DOM node

👉 Answer: C

---

## 27. Output?

```
<input ref={inputRef} />
```

Allows:

- A) Controlled input
- B) Direct DOM access
- C) State update
- D) Context update

👉 Answer: B

---

## 28. Which is correct to focus input?

- A) inputRef.focus()
- B) inputRef.current.focus()
- C) focus(inputRef)
- D) inputRef->focus()

👉 Answer: B

---

## 29. Lifting state up typically involves:

- A) Child managing state

B) Parent managing shared state

C) Context only

D) Redux only

👉 Answer: B

---

### 30. Output?

```
<Child onAdd={addClock} />
```

Purpose?

A) Pass state

B) Pass callback

C) Render list

D) Ref

👉 Answer: B

---

### 31. Which component should hold clocks array?

A) Individual Clock

B) Parent component

C) Ref

D) Context

👉 Answer: B

---

### 32. Output?

```
<select value={zone} onChange={e=>setZone(e.target.value)}>
```

This is:

A) Uncontrolled

B) Controlled select

C) Invalid

D) Ref-based

👉 Answer: B

---

### 33. Adding new clock updates:

A) Props

B) State array

C) Context only

D) DOM directly

👉 Answer: B

---

**34. Which hook is NOT needed here?**

- A) useState
- B) useRef
- C) useEffect
- D) useContext

👉 Answer: D

---

**35. Why use lifting state up instead of refs?**

- A) Refs faster
- B) Predictable data flow
- C) Styling
- D) Simpler JSX

👉 Answer: B

---

**36. Output?**

```
setClocks([...clocks, newClock]);
```

Purpose?

- A) Mutate state
- B) Add new item immutably
- C) Remove item
- D) Replace array

👉 Answer: B

---

**37. Which is incorrect?**

- A) Controlled inputs use state
- B) Uncontrolled inputs use refs
- C) Refs trigger re-render
- D) Keys help reconciliation

👉 Answer: C

---

**38. Which event is used for dropdown change?**

- A) onSelect
- B) onClick
- C) onChange
- D) onInput

👉 Answer: C

---

**39. List rendering performance depends on:**

- A) CSS
- B) Keys
- C) JSX
- D) Refs

👉 Answer: B

---

**40. Parent-to-child data flow uses:**

- A) State
- B) Props
- C) Refs
- D) Context only

👉 Answer: B

---

## 🔴 HARD LEVEL (41–60) — LAB & TRICKY

**41. Output?**

```
{clocks.length === 0 && <p>No clocks</p>}
```

Renders when:

- A) clocks exist
- B) clocks empty
- C) Always
- D) Never

👉 Answer: B

---

**42. Best key for country clocks:**

- A) Index
- B) Random number
- C) Country+timezone ID
- D) Date.now()

👉 Answer: C

---

**43. Why keys must be stable?**

- A) Styling
- B) Correct diffing
- C) Routing
- D) Context

👉 Answer: B

---

#### 44. Output?

```
<input defaultValue="India" />
```

This input is:

- A) Controlled
- B) Uncontrolled
- C) Invalid
- D) Ref-based only

👉 Answer: B

---

#### 45. Which is better for forms in React?

- A) Uncontrolled always
- B) Controlled components
- C) Direct DOM
- D) jQuery

👉 Answer: B

---

#### 46. Which breaks one-way data flow?

- A) Props
- B) State
- C) Direct DOM manipulation
- D) Lifting state up

👉 Answer: C

---

#### 47. Correct flow for Add Clock:

- A) Child updates parent state directly
- B) Child calls parent callback
- C) Parent reads child DOM
- D) Ref updates state

👉 Answer: B

---

#### 48. Output?

```
onSubmit={() => addClock(zone)}
```

Issue?

- A) Correct
- B) Missing preventDefault

- C) Syntax error
  - D) Context misuse
- 👉 Answer: B
- 

**49. Which avoids duplicate clocks?**

- A) Random keys
- B) Check before adding
- C) Refs
- D) CSS

👉 Answer: B

---

**50. Why not store derived data in state?**

- A) Memory
- B) Duplication bugs
- C) Styling
- D) Speed

👉 Answer: B

---

**51. Which hook causes re-render when changed?**

- A) useRef
- B) useState
- C) useContext
- D) Both B and C

👉 Answer: D

---

**52. Selecting timezone affects which component most?**

- A) Clock child
- B) Form component
- C) Parent managing list
- D) Ref

👉 Answer: C

---

**53. Output?**

```
<input ref={r} value={v} />
```

This is:

- A) Valid controlled
- B) Valid uncontrolled
- C) Mixed (not recommended)

D) Error

👉 Answer: C

---

**54. Which improves scalability of clock app?**

- A) Single state object
- B) Lifting state up properly
- C) Refs everywhere
- D) Inline DOM ops

👉 Answer: B

---

**55. Which ensures consistent UI updates?**

- A) setState immutability
- B) Direct mutation
- C) Refs
- D) CSS

👉 Answer: A

---

**56. Removing a clock should use:**

- A) splice() on state
- B) filter() to create new array
- C) pop()
- D) delete

👉 Answer: B

---

**57. Output?**

```
setClocks(clocks.filter(c => c.id !== id));
```

Purpose?

- A) Add clock
- B) Update clock
- C) Remove clock
- D) Sort clocks

👉 Answer: C

---

**58. Which concept ties forms + lists together?**

- A) Context
- B) Lifting state up
- C) Refs

D) JSX

👉 Answer: B

---

**59. Why React discourages direct DOM access?**

- A) Slow browser
- B) Breaks declarative model
- C) Syntax issues
- D) Styling conflict

👉 Answer: B

---

**60. Main lab objective of this session:**

- A) JSX syntax
- B) Lists, forms, refs, shared state
- C) CSS design
- D) Routing

👉 Answer: B

---

## SESSION 25 – REACT (ERROR BOUNDARIES, COMPOSITION, THINKING IN REACT)

---

### EASY LEVEL (1–15)

**1. Error Boundaries in React are used to:**

- A) Handle async errors
- B) Catch JavaScript errors in components
- C) Handle HTTP errors
- D) Validate props

👉 Answer: B

---

**2. Error boundaries catch errors during:**

- A) Event handlers
- B) Rendering and lifecycle methods
- C) API calls only
- D) CSS rendering

👉 Answer: B

---

**3. Error boundaries are implemented using:**

- A) Function components
- B) Class components
- C) Hooks only
- D) Context

👉 Answer: B

---

**4. Which lifecycle method catches errors?**

- A) componentDidMount
- B) componentDidCatch
- C) render
- D) componentWillUnmount

👉 Answer: B

---

**5. Which static method is also used in error boundaries?**

- A) getDerivedStateFromProps
- B) getDerivedStateFromError
- C) shouldComponentUpdate
- D) setState

👉 Answer: B

---

**6. Error boundaries prevent app from:**

- A) Rendering JSX
- B) Crashing completely
- C) Making API calls
- D) Updating state

👉 Answer: B

---

**7. Composition in React means:**

- A) Using inheritance
- B) Combining components
- C) Using classes only
- D) Using hooks

👉 Answer: B

---

**8. React recommends composition over:**

- A) Hooks
- B) Props
- C) Inheritance
- D) JSX

👉 Answer: C

---

**9. Containment means:**

- A) Child controls parent
- B) Parent controls child layout
- C) Passing components as children
- D) Inheriting methods

👉 Answer: C

---

**10. Specialization means:**

- A) Generic component
- B) Component customized via props
- C) CSS styling
- D) Inheritance

👉 Answer: B

---

**11. Thinking in React starts with:**

- A) Writing code
- B) Designing UI mock
- C) Building backend
- D) Writing CSS

👉 Answer: B

---

**12. Thinking in React focuses on:**

- A) DOM manipulation
- B) Component-based approach
- C) Class inheritance
- D) jQuery usage

👉 Answer: B

---

**13. React components should be:**

- A) Large and complex
- B) Small and reusable
- C) Global
- D) Stateless only

👉 Answer: B

---

**14. Which React concept helps reuse UI logic?**

- A) CSS
- B) Composition
- C) Inheritance

D) Refs

👉 Answer: B

---

**15. Error boundaries display:**

- A) Stack trace
- B) Fallback UI
- C) Raw error
- D) Browser alert

👉 Answer: B

---

 **MEDIUM LEVEL (16–30)**

**16. Error boundaries do NOT catch errors in:**

- A) Rendering
- B) Constructors
- C) Lifecycle methods
- D) Event handlers

👉 Answer: D

---

**17. Which component should be wrapped by error boundary?**

- A) Root component only
- B) Every component
- C) Components likely to fail
- D) Only child components

👉 Answer: C

---

**18. Composition allows passing components via:**

- A) State
- B) Props
- C) Context
- D) Refs

👉 Answer: B

---

**19. Output?**

```
<Layout>
  <Header />
  <Footer />
</Layout>
```

This is example of:

- A) Inheritance
- B) Containment
- C) Specialization
- D) Context

👉 Answer: B

---

## 20. Specialization is achieved using:

- A) extends keyword
- B) children prop
- C) Props customization
- D) Context

👉 Answer: C

---

## 21. Why React discourages inheritance?

- A) Slow performance
- B) Less flexible
- C) Harder to reuse
- D) Both B and C

👉 Answer: D

---

## 22. Thinking in React step after UI breakdown:

- A) Add lifecycle
- B) Build static version
- C) Add error boundary
- D) Add context

👉 Answer: B

---

## 23. State should be owned by:

- A) Child
- B) Lowest common ancestor
- C) Root only
- D) Context always

👉 Answer: B

---

## 24. Error boundary state typically stores:

- A) Error message
- B) Boolean error flag
- C) Component tree
- D) Props

👉 Answer: B

---

## 25. Output?

```
if (this.state.hasError) {  
  return <h1>Error</h1>;  
}
```

This is:

- A) Normal render
- B) Fallback UI
- C) Event handling
- D) Context

👉 Answer: B

---

## 26. Composition improves:

- A) Styling
- B) Reusability
- C) Routing
- D) API calls

👉 Answer: B

---

## 27. Thinking in React promotes:

- A) Imperative coding
- B) Declarative UI
- C) DOM access
- D) jQuery

👉 Answer: B

---

## 28. React component tree represents:

- A) DOM tree
- B) UI hierarchy
- C) Database schema
- D) Routing table

👉 Answer: B

---

## 29. Error boundary placement affects:

- A) Styling
- B) Error isolation
- C) Routing
- D) Hooks

👉 Answer: B

---

**30. Which is NOT a Thinking in React step?**

- A) Break UI into components
- B) Build static version
- C) Optimize with Redux
- D) Identify state

👉 Answer: C

---

## 🔴 HARD LEVEL (31–45)

**31. Why error boundaries must be class components?**

- A) Hooks are slower
- B) Lifecycle methods required
- C) JSX limitation
- D) Context issue

👉 Answer: B

---

**32. Error boundaries do NOT catch errors in:**

- A) Child render
- B) Child lifecycle
- C) Async callbacks
- D) Constructors

👉 Answer: C

---

**33. Best practice for error boundaries:**

- A) One global boundary
- B) Many small boundaries
- C) No boundaries
- D) Only root boundary

👉 Answer: B

---

**34. Output?**

```
function Wrapper({ children }) {  
  return <div>{children}</div>;  
}
```

This is example of:

- A) Inheritance
- B) Containment

C) Error boundary

D) Context

👉 Answer: B

---

**35. Why composition scales better than inheritance?**

A) Simpler syntax

B) Flexible combinations

C) Less coupling

D) All of the above

👉 Answer: D

---

**36. Specialization vs Inheritance difference:**

A) Specialization uses props

B) Inheritance uses extends

C) React prefers specialization

D) All of the above

👉 Answer: D

---

**37. Thinking in React identifies state by asking:**

A) Is it reusable?

B) Does it change?

C) Is it CSS?

D) Is it JSX?

👉 Answer: B

---

**38. Which should NOT be state?**

A) Derived data

B) User input

C) Time

D) Toggle flags

👉 Answer: A

---

**39. Error boundary fallback UI should be:**

A) Complex

B) Minimal and safe

C) Interactive

D) Stateful

👉 Answer: B

---

**40. Why not use try-catch in render for errors?**

- A) Syntax error
- B) Doesn't catch lifecycle errors
- C) Breaks React model
- D) All of the above

👉 Answer: D

---

**41. Component hierarchy helps in:**

- A) Routing
- B) State management
- C) UI clarity
- D) All of the above

👉 Answer: D

---

**42. Thinking in React encourages data flow:**

- A) Two-way
- B) Bottom-up
- C) One-way top-down
- D) Circular

👉 Answer: C

---

**43. Error boundary can be reused by:**

- A) Inheritance
- B) Composition
- C) Hooks
- D) Context

👉 Answer: B

---

**44. Which improves maintainability most?**

- A) Inheritance chains
- B) Component composition
- C) Global state
- D) Direct DOM

👉 Answer: B

---

**45. Main goal of “Thinking in React” is to:**

- A) Write faster code
- B) Design correct component architecture
- C) Avoid JSX
- D) Use class components

👉 Answer: B



# SESSIONS 26, 27 & 28 – REACT-REDUX

---

## EASY LEVEL (1–20)

### 1. Redux is mainly used for:

- A) Styling UI
- B) State management
- C) Routing
- D) API calls

👉 Answer: B

---

### 2. Redux is:

- A) A framework
- B) A JavaScript library
- C) A database
- D) A compiler

👉 Answer: B

---

### 3. Redux follows which architecture?

- A) MVC
- B) MVVM
- C) Flux
- D) Layered

👉 Answer: C

---

### 4. Redux store holds:

- A) UI components
- B) Application state
- C) CSS
- D) Routes

👉 Answer: B

---

### 5. Redux state is:

- A) Mutable
- B) Immutable

C) Local only

D) Optional

👉 Answer: B

---

#### 6. Actions in Redux are:

A) Functions

B) Objects

C) Classes

D) Components

👉 Answer: B

---

#### 7. An action must have:

A) value

B) type

C) payload only

D) reducer

👉 Answer: B

---

#### 8. Reducers are:

A) UI components

B) Pure functions

C) Middleware

D) Hooks

👉 Answer: B

---

#### 9. Reducers take which arguments?

A) state, action

B) action, dispatch

C) store, action

D) props, state

👉 Answer: A

---

#### 10. Redux store is created using:

A) createReducer()

B) createStore()

C) useStore()

D) makeStore()

👉 Answer: B

---

**11. Redux supports:**

- A) Multiple stores
- B) Single store
- C) No store
- D) Database store

👉 **Answer: B**

---

**12. React-Redux is used to:**

- A) Replace Redux
- B) Connect React with Redux
- C) Replace React
- D) Handle CSS

👉 **Answer: B**

---

**13. Provider component is used to:**

- A) Create store
- B) Pass store to React app
- C) Dispatch actions
- D) Create reducer

👉 **Answer: B**

---

**14. Redux store is global to:**

- A) One component
- B) One function
- C) Entire application
- D) One page

👉 **Answer: C**

---

**15. Dispatch function is used to:**

- A) Read state
- B) Update UI
- C) Send actions
- D) Create reducer

👉 **Answer: C**

---

**16. Redux data flow is:**

- A) Two-way
- B) Circular
- C) One-way
- D) Random

👉 **Answer: C**

---

**17. Redux improves:**

- A) Styling
- B) State predictability
- C) Routing
- D) Performance only

👉 **Answer: B**

---

**18. Which is NOT a Redux core concept?**

- A) Store
- B) Reducer
- C) Action
- D) Component

👉 **Answer: D**

---

**19. Redux is best suited for:**

- A) Small static apps
- B) Large apps with shared state
- C) Only backend
- D) CSS-heavy apps

👉 **Answer: B**

---

**20. React Clock app Redux usage is mainly for:**

- A) Styling clocks
- B) Managing time zones and clocks state
- C) Routing
- D) API calls

👉 **Answer: B**

---

## 🟡 MEDIUM LEVEL (21–40) — WITH SNIPPETS

**21. Output?**

```
const action = { type: "ADD_CLOCK" };
```

This is an example of:

- A) Reducer
- B) Action
- C) Store

D) Middleware

👉 Answer: B

---

## 22. Output?

```
dispatch({ type: "REMOVE_CLOCK", id: 1 });
```

Purpose?

- A) Read state
- B) Send action to reducer
- C) Create store
- D) Render component

👉 Answer: B

---

## 23. Output?

```
function reducer(state = [], action) {  
  return state;  
}
```

This reducer is:

- A) Invalid
- B) Pure function
- C) Stateful
- D) Async

👉 Answer: B

---

## 24. Reducer must NOT:

- A) Mutate state
- B) Return new state
- C) Handle action
- D) Be pure

👉 Answer: A

---

## 25. Output?

```
const store = createStore(reducer);
```

Creates:

- A) Action

B) Reducer

C) Store

D) Provider

👉 Answer: C

---

## 26. Which hook reads Redux state in React?

A) useState

B) useReducer

C) useSelector

D) useContext

👉 Answer: C

---

## 27. Which hook dispatches Redux actions?

A) useDispatch

B) useAction

C) dispatchAction

D) useStore

👉 Answer: A

---

## 28. Output?

```
const clocks = useSelector(state => state.clocks);
```

Purpose?

A) Update state

B) Read state

C) Dispatch action

D) Create reducer

👉 Answer: B

---

## 29. Output?

```
<Provider store={store}>
  <App />
</Provider>
```

Why needed?

A) Styling

B) Make store available

C) Routing

D) Lifecycle

👉 Answer: B

---

**30. Redux middleware is used for:**

- A) UI logic
- B) Async operations & logging
- C) Styling
- D) Rendering

👉 Answer: B

---

**31. Which middleware is commonly used for async actions?**

- A) redux-log
- B) redux-promise
- C) redux-thunk
- D) redux-sync

👉 Answer: C

---

**32. Output?**

```
dispatch(addClock(clock));
```

`addClock` is likely a:

- A) Reducer
- B) Action creator
- C) Store
- D) Selector

👉 Answer: B

---

**33. Action creators return:**

- A) Reducers
- B) Actions
- C) State
- D) Components

👉 Answer: B

---

**34. Redux state should be updated using:**

- A) Direct mutation
- B) Reducers only
- C) Components

D) Context

👉 Answer: B

---

**35. Which tool helps debug Redux state?**

- A) Chrome DevTools
- B) Redux DevTools
- C) Node REPL
- D) ESLint

👉 Answer: B

---

**36. Combining reducers is done using:**

- A) mergeReducers()
- B) combineReducers()
- C) joinReducers()
- D) useReducer()

👉 Answer: B

---

**37. Redux replaces which React feature in large apps?**

- A) Props
- B) State
- C) Local component state partially
- D) JSX

👉 Answer: C

---

**38. Redux store updates cause:**

- A) Page reload
- B) Re-render of subscribed components
- C) Error
- D) DOM replacement

👉 Answer: B

---

**39. Redux is NOT ideal when:**

- A) Many components share state
- B) App is complex
- C) App is very small
- D) State is global

👉 Answer: C

---

---

**40. Clock app using Redux improves:**

- A) CSS
- B) Centralized state control
- C) Routing
- D) Server speed

👉 Answer: B

---

## 🔴 HARD LEVEL (41–60) — LAB & TRICKY

**41. Redux enforces immutability to:**

- A) Improve styling
- B) Avoid side effects
- C) Speed UI
- D) Reduce code

👉 Answer: B

---

**42. Output?**

```
return [...state, action.payload];
```

This ensures:

- A) Mutation
- B) Immutability
- C) Deletion
- D) Sorting

👉 Answer: B

---

**43. Why reducers must be pure?**

- A) For async calls
- B) For predictable state updates
- C) For routing
- D) For styling

👉 Answer: B

---

**44. Which is NOT allowed in reducer?**

- A) switch(action.type)
- B) API call
- C) Returning new state
- D) Reading action

👉 Answer: B

---

**45. Best state structure for multiple clocks:**

- A) Object
- B) Array of clock objects
- C) Single variable
- D) Context only

👉 Answer: B

---

**46. Clock “Add” button Redux flow:**

- A) Component → Reducer → Action
- B) Component → Dispatch → Reducer → Store
- C) Store → Reducer → Component
- D) Reducer → Component

👉 Answer: B

---

**47. Redux vs Context: Redux is preferred when:**

- A) Few props
- B) App grows large
- C) Only styling needed
- D) No state

👉 Answer: B

---

**48. Redux makes state:**

- A) Implicit
- B) Predictable
- C) Random
- D) Local

👉 Answer: B

---

**49. Removing clock in Redux uses:**

- A) splice()
- B) filter() returning new array
- C) delete keyword
- D) pop()

👉 Answer: B

---

**50. Which ensures component updates only when needed?**

- A) useState
- B) useSelector equality check
- C) Context

D) Refs

👉 Answer: B

---

**51. Redux store subscribes components using:**

- A) useDispatch
- B) useSelector
- C) Provider
- D) Reducer

👉 Answer: B

---

**52. Which Redux concept handles side effects?**

- A) Reducer
- B) Action
- C) Middleware
- D) Store

👉 Answer: C

---

**53. Redux simplifies debugging because:**

- A) CSS based
- B) Single source of truth
- C) Multiple stores
- D) DOM control

👉 Answer: B

---

**54. Clock app Redux migration mainly removes:**

- A) JSX
- B) Multiple local states
- C) Components
- D) Props

👉 Answer: B

---

**55. Which statement is TRUE?**

- A) Reducer can change state directly
- B) State must be immutable
- C) Actions contain logic
- D) Store updates itself

👉 Answer: B

---

**56. Redux follows which principle?**

- A) Multiple sources of truth
- B) Single source of truth
- C) No source of truth
- D) Context only

👉 **Answer: B**

---

**57. Redux store is updated by:**

- A) Components
- B) Reducers
- C) Middleware
- D) Provider

👉 **Answer: B**

---

**58. Which improves scalability most?**

- A) Local state everywhere
- B) Central Redux store
- C) Direct DOM access
- D) Refs

👉 **Answer: B**

---

**59. Redux Clock app advantage:**

- A) Faster timers
- B) Easier add/remove clocks
- C) Less JSX
- D) CSS control

👉 **Answer: B**

---

**60. Main learning objective of this lab:**

- A) JSX practice
- B) Centralized state management using Redux
- C) Routing
- D) Styling

👉 **Answer: B**