**1. What is Express.js?**

**Answer:** Express.js is a **web framework for Node.js** that makes it easier to build web applications and APIs. It simplifies handling HTTP requests (like GET, POST) and routing.

**2. Why use Express.js instead of plain Node.js?**

**Answer:** Express provides **pre-built tools** for routing, middleware, and handling requests/responses, saving time compared to writing everything from scratch in Node.js.

**3. How do you install Express.js?**

**Answer:** Run:

bash

Copy

npm install express

**4. How do you create a basic Express server?**

**Answer:**

javascript

Copy

const express = require('express');

const app = express();

app.listen(3000, () => {

console.log('Server running on port 3000');

});

**5. What is middleware in Express ?**

**Answer:** Middleware is a **function** that runs between receiving a request and sending a response. It can modify requests/responses (like checking authentication).

**6. What is**app.use()**?**

**Answer:** It’s used to **apply middleware** to all routes. Example:

javascript

Copy

app.use(express.json()); // Parses JSON data in requests

**7. How do you handle a GET request?**

**Answer:**

javascript

Copy

app.get('/', (req, res) => {

res.send('Hello World!');

});

**8. How do you handle a POST request?**

**Answer:**

javascript

Copy

app.post('/', (req, res) => {

res.send('Data received!');

});

**9. What is routing in Express?**

**Answer:** Routing defines **how an app responds** to different URLs (e.g., /users vs /products).

**10. How do you extract query parameters?**

**Answer:** From a URL like ?name=John:

javascript

Copy

app.get('/', (req, res) => {

const name = req.query.name; // "John"

});

**11. How do you extract route parameters?**

**Answer:** From a route like /users/:id:

javascript

Copy

app.get('/users/:id', (req, res) => {

const id = req.params.id; // Extracts ID from URL

});

**12. How do you send a JSON response?**

**Answer:**

javascript

Copy

res.json({ message: "Success", data: {} });

**13. What is**res.send()**vs**res.json()**?**

**Answer:**

* res.send() sends **any type** of response (HTML, text).
* res.json() **automatically converts** objects to JSON.

**14. How do you serve static files (CSS, images)?**

**Answer:**

javascript

Copy

app.use(express.static('public')); // Files in "public" folder

**15. What is**express.json()**?**

**Answer:** It **parses JSON data** sent in requests (e.g., from API calls).

**16. What is**express.urlencoded()**?**

**Answer:** It **parses form data** (like from HTML forms).

**17. How do you handle 404 errors?**

**Answer:**

javascript

Copy

app.use((req, res) => {

res.status(404).send("Page not found!");

});

**18. How do you handle errors in Express?**

**Answer:** Use a **error-handling middleware**:

javascript

Copy

app.use((err, req, res, next) => {

res.status(500).send('Something broke!');

});

**19. What is**next()**in Express?**

**Answer:** It **passes control** to the next middleware function.

**20. What are route handlers?**

**Answer:** Functions that handle requests for specific routes (e.g., app.get('/', handler)).

**21. How do you chain multiple route handlers?**

**Answer:** Pass multiple functions:

javascript

Copy

app.get('/', (req, res, next) => {

console.log('First');

next();

}, (req, res) => {

res.send('Done');

});

**22. What is**Router()**in Express?**

**Answer:** It helps **organize routes** into separate files:

javascript

Copy

const router = express.Router();

router.get('/', (req, res) => { ... });

app.use('/api', router); // Mount at "/api"

**23. How do you set cookies in Express?**

**Answer:** Use cookie-parser middleware:

javascript

Copy

const cookieParser = require('cookie-parser');

app.use(cookieParser());

res.cookie('name', 'John');

**24. How do you read cookies?**

**Answer:**

javascript

Copy

const name = req.cookies.name; // "John"

**25. What is CORS? How do you enable it?**

**Answer:** CORS allows **cross-origin requests**. Enable it with:

bash

Copy

npm install cors

javascript

Copy

const cors = require('cors');

app.use(cors());

**26. How do you redirect a request?**

**Answer:**

javascript

Copy

res.redirect('/new-path');

**27. What is**app.set()**?**

**Answer:** It **sets configuration** (e.g., view engine):

javascript

Copy

app.set('view engine', 'ejs');

**28. What is**req.body**?**

**Answer:** Contains **data submitted** in POST requests (needs express.json() or express.urlencoded()).

**29. How do you use templates (like EJS)?**

**Answer:** Install EJS:

bash

Copy

npm install ejs

javascript

Copy

app.set('view engine', 'ejs');

res.render('index', { name: 'John' });

**30. What is**process.env.PORT**?**

**Answer:** It reads the **port from environment variables** (useful for deployment):

javascript

Copy

app.listen(process.env.PORT || 3000);

**Final Tips:**

* Practice **building a small CRUD API** (GET, POST, PUT, DELETE).
* Understand **middleware flow** (order matters!).
* Learn **basic error handling**.