



SETUP & APP

```
npm install koa koa-router
koa-bodyparser
import Koa from 'koa'
const app = new Koa()
app.keys = ['secret-key'] // for signed cookies
app.use(bodyParser())
app.listen(3000)
```

MIDDLEWARE

```
app.use(async (ctx, next) => {
  const start = Date.now()
  await next() // call downstream middleware
  const ms = Date.now() - start
  ctx.set('X-Response-Time', `${ms}ms`)
})
Onion model: middleware wraps around each other
```

CONTEXT OBJECT

```
ctx.request  ctx.response // Koa's enhanced versions
ctx.req      ctx.res       // Node native
ctx.state.user // shared state across middleware
ctx.session.userId // (koa-session)
ctx.app      // app instance
ctx.body = {} // sets response body
ctx.status = 200 // HTTP status code
```

ASYNC/AWAIT FLOW

```
app.use(async (ctx, next) => {
  try {
    await next()
  } catch (err) {
    ctx.status = err.status || 500
    ctx.body = { error: err.message }
    ctx.app.emit('error', err, ctx)
  }
})
```

Error middleware must be first to catch downstream errors

ERROR HANDLING

```
app.on('error', (err, ctx) => {
  console.error('Server Error', err, ctx)
})
// Throw in handler:
const { createError } =
  require('http-errors')
throw createError(404, 'User not found')
throw createError(400, 'Invalid input')
ctx.throw(404, 'Not found') // shorthand
```

BODY PARSING

```
import bodyParser from 'koa-bodyparser'
app.use(bodyParser({
  enableTypes: ['json', 'form'],
  jsonLimit: '5mb',
}))
// Access parsed body:
ctx.request.body // JSON or form data
// File uploads: koa-multer
```

STATIC FILES

```
import serve from 'koa-static'
app.use(serve('./public', {
  maxAge: 86400, // 1 day cache
  gzip: true,
}))
import mount from 'koa-mount'
app.use(mount('/assets',
  serve('./dist')))
```

COOKIES & SESSIONS

```
ctx.cookies.get('session') // unsigned
ctx.cookies.set('token', value, {
  httpOnly: true })
// Signed cookies (requires app.keys):
ctx.cookies.get('session', { signed: true })
// Session middleware:
import session from 'koa-session'
app.use(session({ key: 'sess', maxAge: 86400 }, app))
```

COMMON MIDDLEWARE PACKAGES

@koa/router	routing
koa-bodyparser	JSON/form parsing
koa-static	static file serving
koa-session	session management
koa-compress	gzip compression
koa-logger	request logging
koa-helmet	security headers
koa-jwt	JWT authentication

COMMON GOTCHAS

Error catching middleware must be registered FIRST
await next() is async forgetting await breaks the chain
ctx.body must be set or Koa returns 404 by default
ctx.request.body only set after bodyParser middleware
app.keys required for signed cookies or sessions