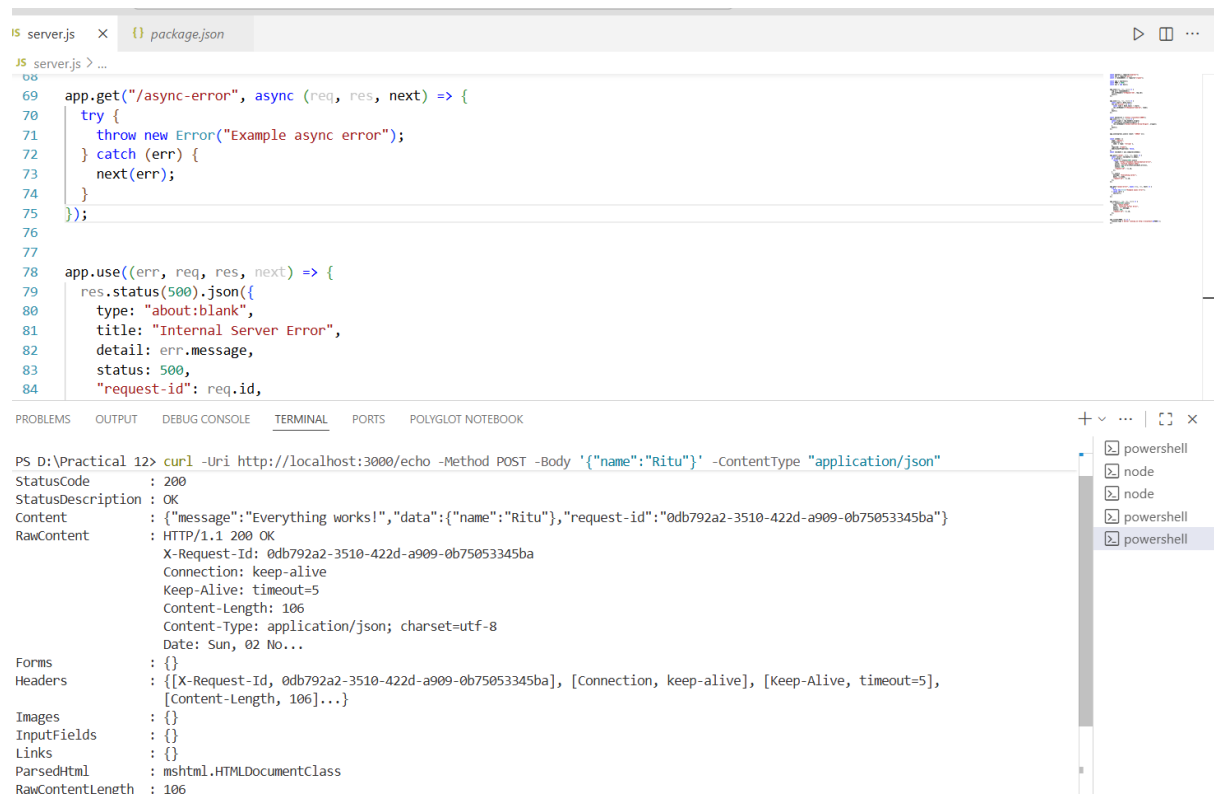


## Practical 12-



```
server.js  x  {} package.json
JS server.js > ...
68
69 app.get("/async-error", async (req, res, next) => {
70   try {
71     throw new Error("Example async error");
72   } catch (err) {
73     next(err);
74   }
75 });
76
77
78 app.use((err, req, res, next) => {
79   res.status(500).json({
80     type: "about:blank",
81     title: "Internal Server Error",
82     detail: err.message,
83     status: 500,
84     "request-id": req.id,
```

```
PS D:\Practical 12> curl -Uri http://localhost:3000/echo -Method POST -Body '{"name":"Ritu"}' -ContentType "application/json"
StatusCode      : 200
StatusDescription: OK
Content         : {"message":"Everything works!","data":{"name":"Ritu"},"request-id":"0db792a2-3510-422d-a909-0b75053345ba"}
RawContent      : HTTP/1.1 200 OK
                  X-Request-Id: 0db792a2-3510-422d-a909-0b75053345ba
                  Connection: keep-alive
                  Keep-Alive: timeout=5
                  Content-Length: 106
                  Content-Type: application/json; charset=utf-8
                  Date: Sun, 02 No...

Forms           : {}
Headers         : {[X-Request-Id, 0db792a2-3510-422d-a909-0b75053345ba], [Connection, keep-alive], [Keep-Alive, timeout=5],
                  [Content-Length, 106]...}
Images          : {}
InputFields     : {}
Links           : {}
ParsedHtml      : mshtml.HTMLDocumentClass
RawContentLength: 106
```

Github- <https://github.com/RituSingh0204/Practical-12-node.git>