

Coverage for app/main.py: 93%

30 statements 28 run 2 missing 0 excluded

« prev ^ index » next coverage.py v7.11.3, created at 2025-11-20 19:17 -0800

```
1 from fastapi import FastAPI
2 from fastapi.middleware.cors import CORSMiddleware
3 from app.utils.search import search_books
4 from app.routers.request_router import router as request_router
5 from app.routers import auth as auth_router
6 from app.routers.rating_router import router as rating_router
7 from app.routers.review_router import router as review_router
8 from app.routers.readinglist_router import router as readinglist_router
9 from app.routers.book_router import router as book_router
10
11
12
13 app = FastAPI()
14
15 app.add_middleware(
16     CORSMiddleware,
17     allow_origins=["*"],
18     allow_methods=["*"],
19     allow_headers=["*"],
20     allow_credentials=False,
21 )
22
23 @app.get("/")
24 def health_check():
25     return {"status": "ok"}
26
27 app.include_router(request_router)
28 app.include_router(auth_router.router, prefix="/auth", tags=["auth"])
29 app.include_router(review_router)
30 app.include_router(book_router)
31 app.include_router(readinglist_router)
32 app.include_router(rating_router)
33
34
```

```
35
36
37 @app.get("/")
38 def read_root():
39     return {"message": "Welcome to my API!"}
40
41 @app.get("/search/{q}")
42 def search(q: str):
43     query = q.lower()
44     result = search_books(q)
45     if not result:
46         return {"result": [], "message": "No matching books found"}
47     return {"results": result}
48
49 # For having it search as they type into textfield
50 # @app.get("/search")
51 # def search(q: str = ""):
52 #     query = q.lower()
53 #     result = search_books(q)
54 #     if not result:
55 #         return {"result": [], "message": "No matching books found"}
56 #     return {"results": result}
57
58
59
```

« prev ^ index » next coverage.py v7.11.3, created at 2025-11-20 19:17 -0800