

Coverage for **app/main.py**: 96%



27 statements

26 run

1 missing

0 excluded

« prev ^ index » next coverage.py v7.11.3, created at 2025-11-21 23:38 -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
24 app.include_router(request_router)
25 app.include_router(auth_router.router)
26 app.include_router(review_router)
27 app.include_router(book_router)
28 app.include_router(readinglist_router)
29 app.include_router(rating_router)
30
31
```

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