

Coverage for **app/repositories/csv_repository.py**: 97%



31 statements

30 run

1 missing

0 excluded

« prev ^ index » next coverage.py v7.11.3, created at 2025-11-21 23:38 -0800

```
1 import csv
2 import os
3 import threading
4 from typing import List, Dict
5 from app.repositories.base_repository import BaseRepository
6
7 class CSVRepository(BaseRepository):
8     """Thread-safe CSV handler shared across all models."""
9     _locks: Dict[str, threading.Lock] = {}
10
11     def _get_lock(self, path: str) -> threading.Lock:
12         if path not in self._locks:
13             self._locks[path] = threading.Lock()
14         return self._locks[path]
15
16     def read_all(self, path: str) -> List[Dict]:
17         """
18         Reads all rows from a CSV file and returns them as a list of dictionaries.
19         """
20         if not os.path.exists(path):
21             return []
22         lock = self._get_lock(path)
23         with lock, open(path, 'r', newline='', encoding='utf-8') as f:
24             return list(csv.DictReader(f))
25
26     def write_all(self, path: str, fieldnames: List[str], rows: List[Dict]):
27         """
28         Overwrites the entire CSV file with the given rows.
29         """
30         lock = self._get_lock(path)
31         with lock, open(path, 'w', newline='', encoding='utf-8') as f:
32             writer = csv.DictWriter(f, fieldnames=fieldnames)
33             writer.writeheader()
34             writer.writerows(rows)
35
36     def append_row(self, path: str, fieldnames: List[str], row: Dict):
37         """
38         Add a single row to the CSV file safely.
39         """
40         file_exists = os.path.exists(path)
41         lock = self._get_lock(path)
42         with lock, open(path, 'a', newline='', encoding='utf-8') as f:
43             writer = csv.DictWriter(f, fieldnames=fieldnames)
```

```
44         if not file_exists:
45             writer.writeheader()
46             writer.writerow(row)
```

« prev ^ index » next coverage.py v7.11.3, created at 2025-11-21 23:38 -0800