Bank

- name: str
- customers: List<Customer>

+ add_customer(customer: Customer) -> None
+ get_customer_by_id(customer_id: int) -> Customer
+ get_account_by_number(account_number: str) -> BankAccount

Customer

- customer_id: int
- first_name: str
- last_name: str
- address: str
- accounts: List[BankAccount]

+ add_account(account: BankAccount) -> None
+ get_account(account_number: str) -> BankAccount

BankAccount

- account_number: str
- account_type: str
- balance: float
- customer: Customer

+ deposit(amount: float) -> None
+ withdraw(amount: float) -> None
+ transfer(to_account: BankAccount, amount: float) -> None

Customer 1 — owns — 0..* BankAccount

records transactions — 1 / 0..*

Logger

- log_file: str (File path for logs)

+ log_transaction(transaction: Transaction) -> None
+ log_error(error: str) -> None
+ log_transaction(transaction: Transaction) -> None

Transaction

- accounts: List<BankAccount>
- transaction_id: int
- account: BankAccount
- transaction_type: str (Deposit, Withdrawal, Transfer)
- amount: float
- timestamp: datetime

+ get_transaction_details() -> str

stores logs — 1 / 0..*

Transaction Log

- log_id: int (Primary Key)
- transaction_id: int (Foreign Key to Transaction)
- account_number: str (Foreign Key to BankAccount)
- log_type: str ("Deposit", "Withdrawal", "Transfer", "Error")
- log_message: str (Descriptive message)
- timestamp: datetime (Log entry timestamp)

1 — has log entry — 1