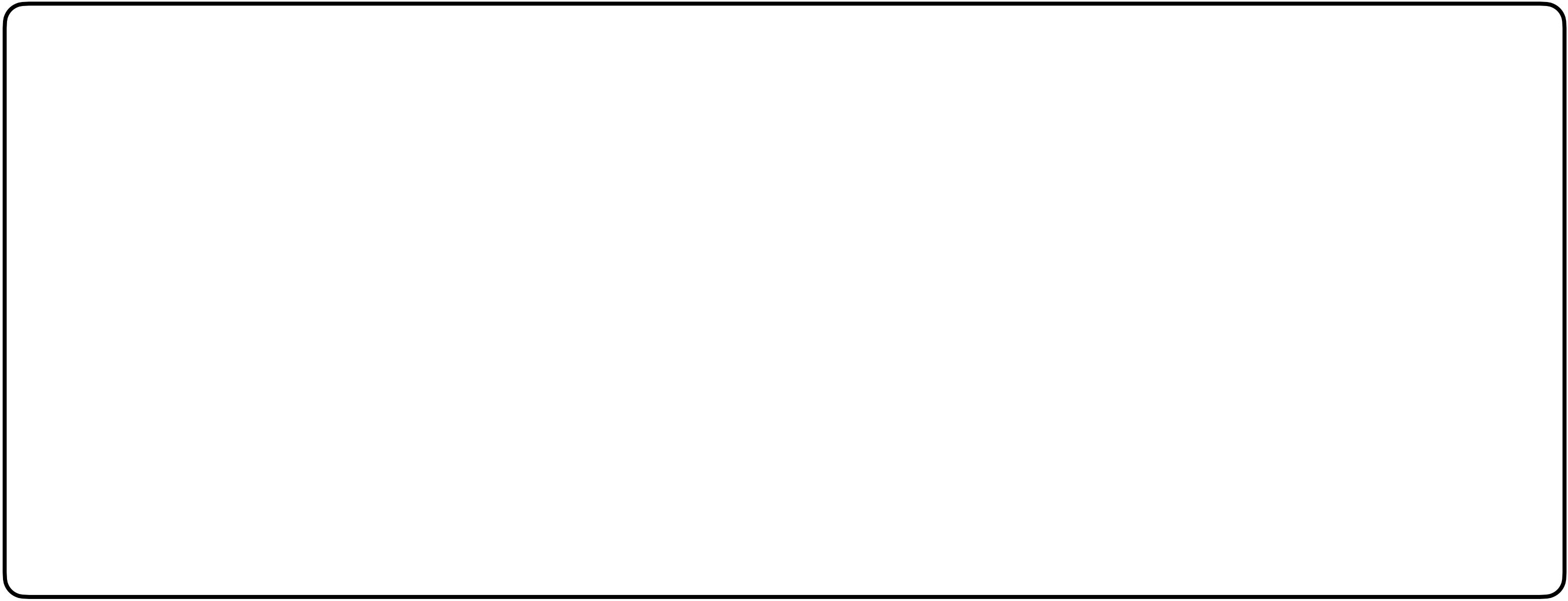



```
public void transfer(Account from,  
                    Account to,  
                    String refId,  
                    Amount amount) {  
  
    accounts.withdraw(from, amount, refId);  
    accounts.deposit(to, amount, refId);  
  
}
```





Transfer Completed

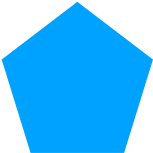


Withdraw

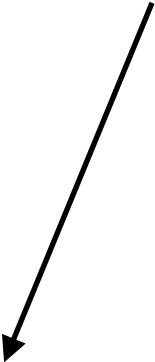
[illegible]

MoneyTransfer









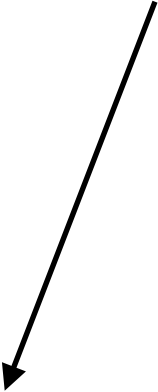


Withdraw

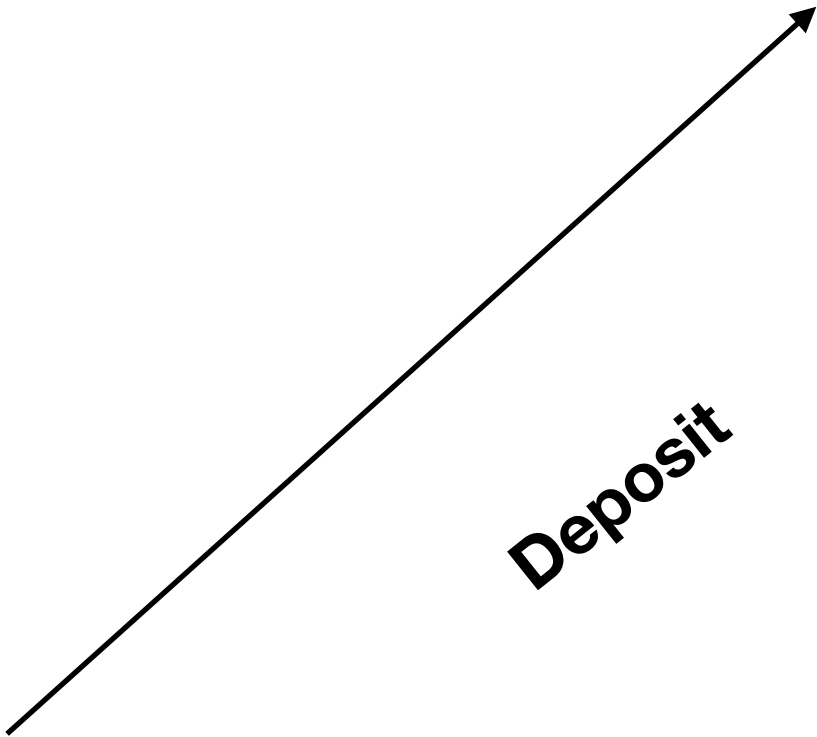


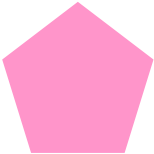




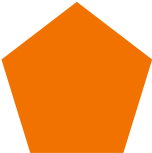






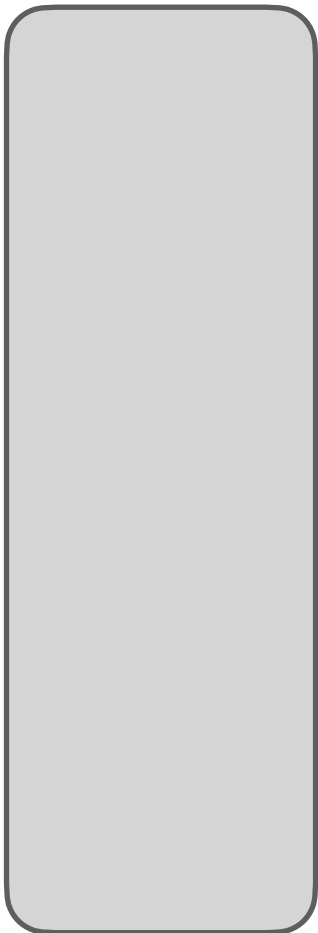








WF History



workflow

Activity

```
public class TransferActivitiesImpl implements TransferActivities {  
  
    @Override  
    public void withdraw(Account account, Amount amount, String refId) {  
        // ...  
    }  
  
    @Override  
    public void deposit(Account account, Amount amount, String refId) {  
        // ...  
    }  
}
```

Activity

```
public class TransferActivitiesImpl implements TransferActivities {  
  
    @Override  
    public void withdraw(Account account, Amount amount, String refId) {  
        // ...  
    }  
  
    @Override  
    public void deposit(Account account, Amount amount, String refId) {  
        // ...  
    }  
}
```









**Complete
Withdraw**

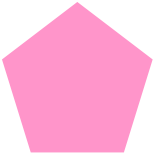




**Complete
Deposit**











Completed





Start Transfer





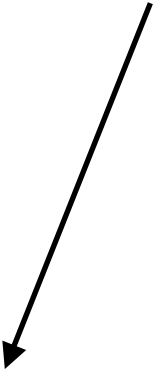
Client

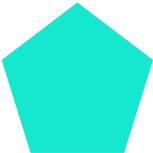


Banking Service



Banking Service







Completed!





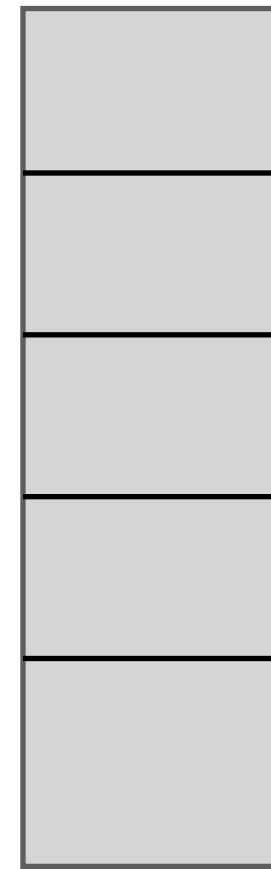
Client

Transfer Completed

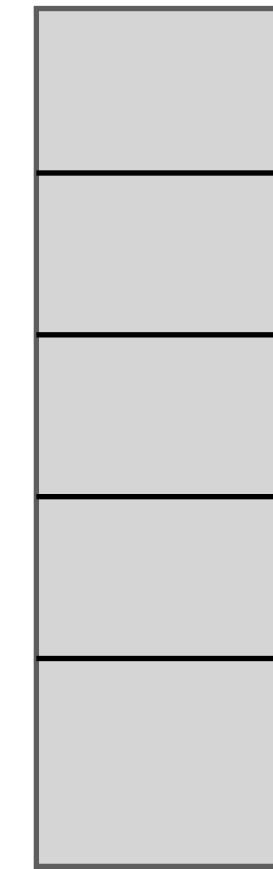


Temporal Server

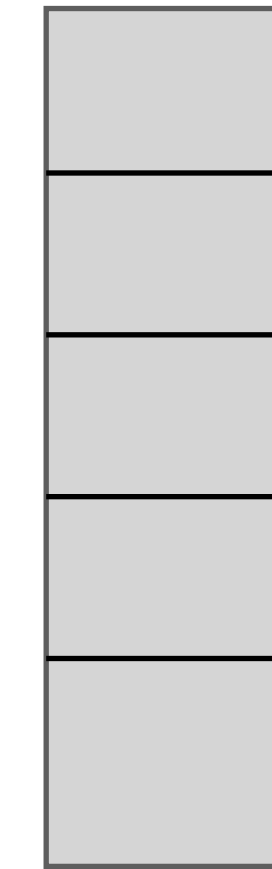
MoneyTransfer



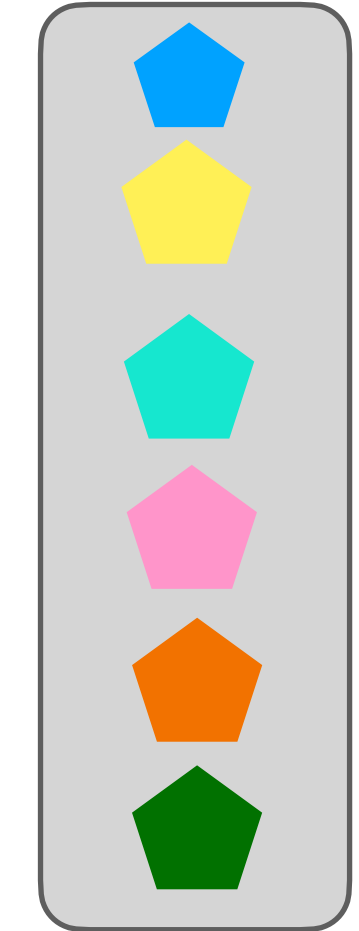
Withdraw



Deposit



WF History



Workflow

```
public void transfer(Account from,  
    Account to,  
    String refId,  
    Amount amount) {
```

```
    accounts.withdraw(from, amount, refId);  
    accounts.deposit(to, amount, refId);
```

Completed!

```
}
```

Activity

```
public class TransferActivitiesImpl implements TransferActivities {  
  
    @Override  
    public void withdraw(Account account, Amount amount, String refId) {  
        // ...  
    }  
  
    @Override  
    public void deposit(Account account, Amount amount, String refId) {  
        // ...  
    }  
}
```

Activity

```
public class TransferActivitiesImpl implements TransferActivities {  
  
    @Override  
    public void withdraw(Account account, Amount amount, String refId) {  
        // ...  
    }  
  
    @Override  
    public void deposit(Account account, Amount amount, String refId) {  
        // ...  
    }  
}
```



Banking Service



Banking Service

Resuming execution after failure

Client Application

Workflow

```
public void transfer(Account from,  
                    Account to,  
                    String refId,  
                    Amount amount) {  
  
    accounts.withdraw(from, amount, refId);  
    accounts.deposit(to, amount, refId);  
  
}
```

Commands

Execution History