Extra Practice

- What type of logging is being used to create the log at right?
  redo-only
- At the start of recovery, what are the possible on-disk values?
  - pins values in memory
    ➔ *can't* go to disk until commit
  - at commit, unpins values
    but does *not* force them to disk
    ➔ older values are still possible

original values:
D1=1000, D2=3000

LSN	*record contents*
100	txn: 1; BEGIN
150	txn: 1; item: D1; new: 2500
350	txn: 2; BEGIN
400	txn: 2; item: D2; new: 6780
470	txn: 1; item: D1; new: 2750
550	**txn: 1; COMMIT**
585	txn: 2; item: D1; new: 1300

	in-memory	possible on-disk
D1:	~~2500~~ **2750**	1000, **2750**
D2:	6780	3000

Extra Practice

- What type of logging is being used to create the log at right?
  redo-only
- At the start of recovery, what are the possible on-disk values?
  - pins values in memory
    ➔ *can't* go to disk until commit
  - at commit, unpins values
    but does *not* force them to disk
    ➔ older values are still possible

original values:
D1=1000, D2=3000

LSN	*record contents*
100	txn: 1; BEGIN
150	txn: 1; item: D1; new: 2500
350	txn: 2; BEGIN
400	txn: 2; item: D2; new: 6780
470	txn: 1; item: D1; new: 2750
550	txn: 1; COMMIT
585	txn: 2; item: D1; new: 1300

	in-memory	possible on-disk
D1:	~~2500~~ ~~2750~~ 1300	**1000, 2750**
D2:	6780	**3000**

Extra Practice

- What if the DBMS were using ***undo-only*** logging instead?

- At the start of recovery, what are the possible on-disk values?
  - does *not* pin values in memory
    ➔ *may* go to disk at any time
  - at commit, forces dirty data values written by that txn to disk
    ➔ older values are no longer possible

original values:
D1=1000, D2=3000

LSN	*record contents*
100	txn: 1; BEGIN
150	txn: 1; item: D1; new: 2500
350	txn: 2; BEGIN
400	txn: 2; item: D2; new: 6780
470	txn: 1; item: D1; new: 2750
550	**txn: 1; COMMIT**
585	txn: 2; item: D1; new: 1300

	in-memory	possible on-disk
D1:	~~2500~~ **2750**	~~1000~~, ~~2500~~, **2750**
D2:	6780	3000, 6780

Extra Practice

- What if the DBMS were using *undo-only* logging instead?

- At the start of recovery, what are the possible on-disk values?
  - does *not* pin values in memory
    ➔ *may* go to disk at any time
  - at commit, forces dirty data values written by that txn to disk
    ➔ older values are no longer possible

original values:
D1=1000, D2=3000

LSN	*record contents*
100	txn: 1; BEGIN
150	txn: 1; item: D1; new: 2500
350	txn: 2; BEGIN
400	txn: 2; item: D2; new: 6780
470	txn: 1; item: D1; new: 2750
550	txn: 1; COMMIT
585	txn: 2; item: D1; new: 1300

	in-memory	possible on-disk
D1:	~~2500~~ ~~2750~~ 1300	~~1000~~, ~~2500~~, **2750, 1300**
D2:	6780	**3000, 6780**

Extra Practice

- What if the DBMS were using ***undo-redo*** logging instead?

- At the start of recovery, what are the possible on-disk values?
  - does *not* pin values in memory
    ➔ *may* go to disk at any time
  - at commit, does *not* force dirty data to disk
    ➔ older values are still possible

original values:
D1=1000, D2=3000

LSN	*record contents*
100	txn: 1; BEGIN
150	txn: 1; item: D1; new: 2500
350	txn: 2; BEGIN
400	txn: 2; item: D2; new: 6780
470	txn: 1; item: D1; new: 2750
550	txn: 1; COMMIT
585	txn: 2; item: D1; new: 1300

	in-memory	possible on-disk
D1:	~~2500~~ ~~2750~~ 1300	**1000, 2500, 2750, 1300**
D2:	6780	**3000, 6780**