

# InnoDB Buffer Pool Flushing Optimization

**목적:** 이 강의에서는 InnoDB Buffer Pool의 플러싱 메커니즘과 이를 최적화하는 방법을 배웁니다.

**포인트:** 변경된 데이터 즉 더티 페이지를 효율적으로 디스크에 저장하는 과정과 그 최적화에 대해 아는 것이 중요합니다.

## ▼ Buffer Pool Flushing Mechanism

**플러싱 개요:** Buffer Pool에 쌓인 더티 페이지(변경된 데이터)들을 디스크에 주기적으로 플러시하는 과정.

**Page Cleaner Thread:** Page Cleaner Thread 가 더티 페이지들을 플러쉬하는 역할을 담당.

- `innodb_page_cleaners` 값을 통해 조정할 수 있음.
- 기본 값은 4이다.

## ▼ Optimization Strategies

**Page Cleaner Thread 조정:** 데이터 변경 작업이 많은 환경에서 Thread 수를 늘려 플러싱 성능 향상.

- 설정할 수 있는 최대값은 `innodb_buffer_pool_instances` 수와 같다.

**플러싱 인접 페이지:** `innodb_flush_neighbors` 값을 조절하여 플러싱할 때 이웃 페이지들을 함께 플러쉬하는 방식 설정. (특히 HDD 스토리지에서 유용하다)

- `innodb_flush_neighbors` 값이 0 이면 이웃 페이지들과 같이 플러쉬 되지 않음.
- `innodb_flush_neighbors` 값이 1이면 연속된 더티페이지들은 같이 디스크에 플러쉬된다.

- `innodb_flush_neighbors` 값이 2이면 같은 Extent 에 있는 더티 페이지들이 같이 디스크에 플러쉬된다.

**LRU 리스트 스캔 최소화:** `innodb_lru_scan_depth` 값을 조절하여 더티 페이지 스캔을 자주하지 않도록 하는 방식 설정. (특히 Wtire Intensive 한 어플리케이션에서 유의미 할 수 있음)

- `innodb_lru_scan_depth` 로 명시된 값만큼 1초마다 LRU 리스트를 스캔한다.