뷰 (그룹) 히스토리 관리 모듈

1. Task
   1. Stack 형태의 저장 공간
   2. TaskManager 을 통하여 push, pop 을 진행하도록 한다 (직접 사용 하지 말 것)
   3. setTaskName, getTaskName 을 통하여 Task의 identify를 가능하게 한다.
2. TaskManager
   1. 클래스가 아닌 객체 (싱글턴으로 이해해도 무관)
   2. TaskManager 는 Task를 관리하는 객체.
   3. 인터페이스
      1. addHistory(뷰그룹)
      2. getLastHistory()
      3. getLastHistoryWithRemove()
      4. addTask()
      5. getCurrentTask()
      6. removeLastTask()
      7. swapTask()
      8. getTaskIndex()
      9. setNeedChangeTask()
      10. needChangeTask()
      11. clearTask()
   4. 기본 로직
      1. addHistory : 뷰그룹이 시작되면 스택에 해당 뷰그룹을 쌓음
      2. getLastHistory : 마지막 뷰그룹을 리턴. 없으면 null 리턴
      3. getLastHistoryWithRemove : 마지막 뷰그룹을 리턴하고 스택에서 삭제함.

없으면 null을 리턴

* + 1. getCurrentTask : 현재 Task 를 return. setTaskName, getTaskName 를 이용할 때 사용
    2. addTask : 현재의 Task는 유지한채 새로운 Task를 생성.
    3. removeLastTask : 현재의 Task 를 삭제
    4. setNeedChangeTask : : 특정한 케이스에 addTask 를 할 수있도록 Flag 설정.
    5. needChangeTask : 필요 시점에 setNeedChangeTask 로 설정된 flag 값을 확인하여 addTask를 할수있도록 도와준다.
    6. clearTask : TaskManager 를 초기화 한다. Task에 저장된 뷰그룹 순서는 초기화 되지만 화면에 그려진 뷰그룹을 화면에서 지우는 것은 아님.