

# 디버깅 (Eclipse)

2018년 1월 11일 목요일    오후 8:33

- 원인을 정확하고 빠르게 파악하여 버그를 해결학 위해 디버거의 사용법을 숙지하고 사용한다.

## 이클립스 디버거 뷰

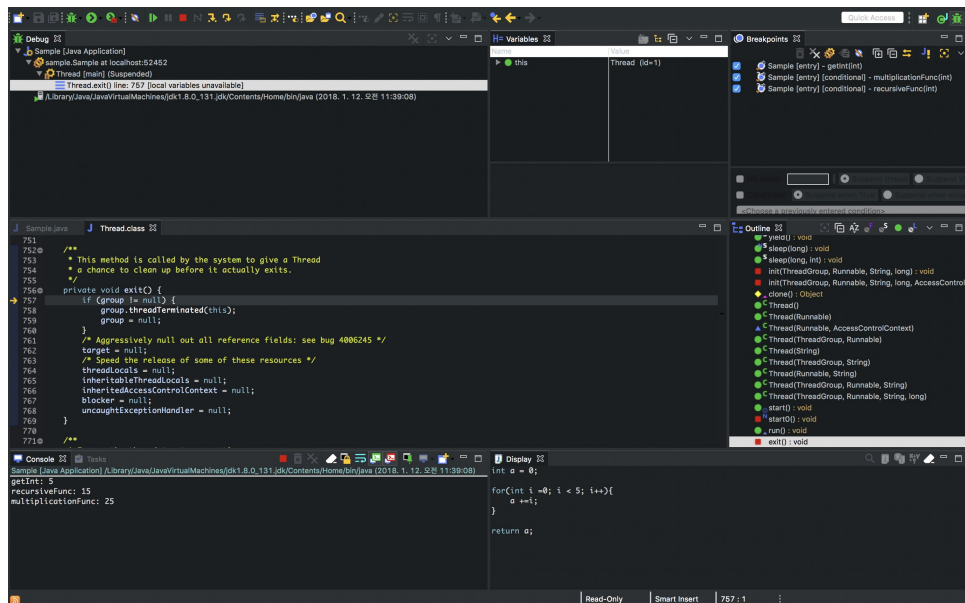
- 이클립스는 디버거 뷰를 제공하여 디버거를 사용할 수 있도록 한다.
- 우측 상단에 벌레 모양

## 디버깅 시작

- 상단 메뉴의 Run에서 프로그램의 시작할 때 Debug를 이용하여 프로그램을 실행하면 디버거가 작동한다.
- Run - shift + fn + command + F11
- Debug - fn + command + F11

## 브레이크 포인트 설정과 뷰

- 디버깅을 할 때 가장 먼저 하는 일은 브레이크포인트를 잡는 일이다.
- 버그 또는 에러가 일어나거나 의심가는 라인, 변수를 추적할 수 있는 라인쯤에 브레이크 포인트를 잡는다.
- 프로그램을 디버깅 할 때, 브레이크 포인트로 지정한 라인이 실행 되면서 디버거가 작동하게 되고, 프로그램을 라인별로 진행해가면서 관찰을 진행할 수 있다.
- 편집기 인쪽 파란 부분(마커 바)에서 더블 클릭 혹은 우클릭 후 Toggle break point를 누른다.



- 브레이크 포인트 뷰에서는 모든 브레이크 포인트들의 정보를 확인할 수 있으며, 이를 수정할 수 있다. ( 활성화/비활성화, 삭제, 조건 등...)
- 디버깅을 진행하고 있는 도중에도 다른 의심이 가는 라인에 브레이크 포인트를 걸 수

있다.

### 스텝 단위 진행

- 지정한 브레이크 포인트에 다르면 동시에 디버거가 작도아게 되고 그 라인부터 스텝 단위로 디버깅이 진행된다.



1. Skip All Breakpoint: 모든 브레이크 포인트를 건너뛴
2. Resume: 다음 브레이크 포인트를 만날 때까지 진행
3. Suspend: 현재 동작중인 스레드 일시 정지
4. Terminate: 현재 진행중인 프로그램 종료
5. Step Into: 메서드가 존재할 경우, 해당 메서드에 진입
6. Step Over: 다음 라인으로 이동하며, 메서드가 있을 경우 이를 무시하고 다음 메서드로 이동
7. Step Return: 동작중인 해당 메서드에서 바로 리턴
8. Drop to Frame : 해당 메서드를 처음부터 다시 실행
9. Use Step Filters: 스텝 필터링

### Variables View (변수 뷰)

- 디버깅 중 변수의 값 또는 객체의 상태를 확인하기 위한 뷰
- 특정 변수에 영향을 끼칠 다른 변수나 객체의 상황을 실시간으로 검사할 필요가 있을 때 사용
- 변수 혹은 객체의 상태를 확인하고, 변수의 현재 상황을 저장할 수 있다. 우클릭 후 Copy Variables를 선택.
- 편집 창으로 돌아와 Command + shift + i를 누르거나 우클릭 후 Inspect를 선택하여 Inspector 창을 불러온다. Inspector 창에서 다시 한 번 단축키를 누르면 해당 변수를 Expression 뷰로 보내게 되며 지속적으로 변수의 상태를 확인할 수 있다.

### Expression View

- 변수 이름을 직접 입력하거나 수행해보고 싶은 명령어를 직접 입력하여 결과 값을 확인
- 결과 값 확인 뿐만 아니라, Expression에 써놓은 변수는 명시적으로 삭제하지 않는 이상 지속적인 확인이 가능하다.
- Expression View에 변수를 추가하는 방법은, 해당 변수를 선택하고 우클릭후 Watch를 통해 추가할 수 있다.

### Display View

- 현 문맥에서 사용할 수 있는 명령어를 실행하거나 변수의 값을 조작할 수 있다.
- Expression에서도 비슷한 기능을 제공하지만, 원본 코드의 수정 없이 편하게 맥락을 수정할 수 있는 장점을 갖는다.

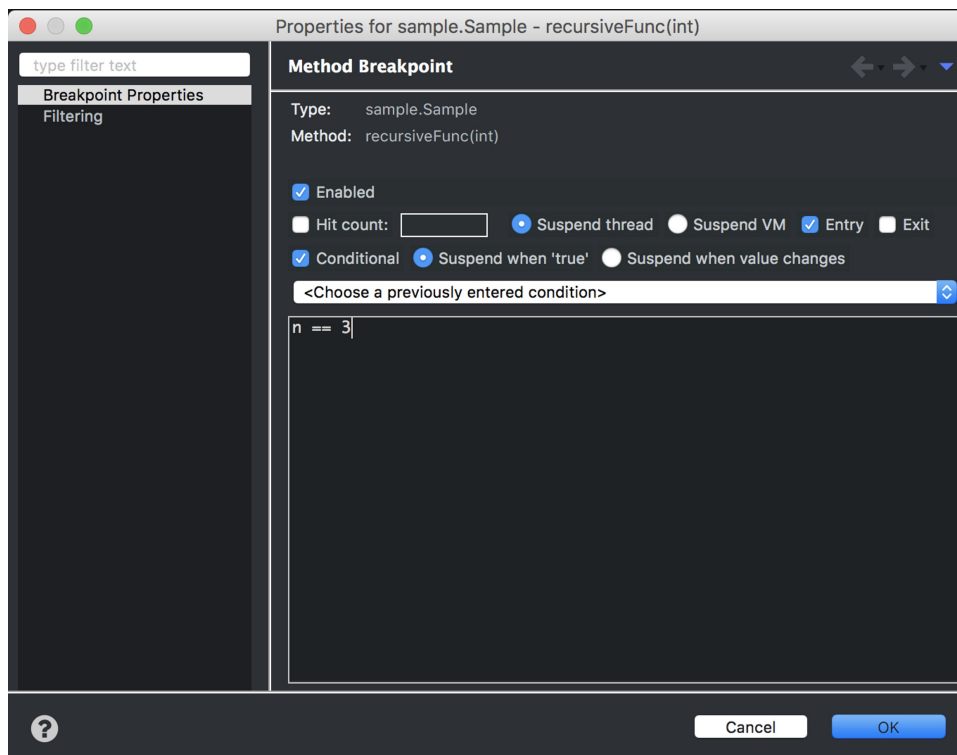
- 필요한 명령어들을 적은 후, 실행하고자 하는 부분만 드래그하여 수정하거나 값을 리턴받을 수 있다.



- 1번은 값을 반환받고 싶을 때 사용하며, 2번은 단순히 실행만 할 경우 사용한다.

## 브레이크 포인트에 조건 걸기

- 반복문 안에 들어가 있는 코드를 디버깅할 때 브레이크 포인트에 조건을 걸면 더 유용하게 사용할 수 있다.
- 모든 상황이 아닌 특정 조건에서의 값을 확인할 수 있으며, 이를 위해 브레이크 포인트에 조건을 건다.
- 해당 브레이크 포인트에서 Breakpoint Properties 옵션에 들어가서 조건문을 설정하여 디버거의 활성화 조건을 설정할 수 있다.



- hit count 옵션은 설정한 값만큼 반복된 후의 디버깅 화면을 출력한다.
- hit count 옵션은 반복문에서 100번 이후에 디버깅을 시작하고 싶은 경우에 사용할 때 유용하다.