

Tugas Pemrograman Berorientasi Objek Sesi 7

Nama : M. Yusgi Padilllah

Kelas : TI21E

NIM : 20210040126

1.

```
package sesitujuh.satu;

public class Outer {
    int number = 0;
    private class Inner{
        public void print(){
            System.out.println("Mengakses inner class yang ke: " + (++number));
        }
    }
    void displayFromMethod(){
        Inner in = new Inner();
        in.print();
    }
}

class Main{
    public static void main(String[] args) {
        Outer out = new Outer();
        out.displayFromMethod();
        out.displayFromMethod();
    }
}
```

- Ada 3 file ekstensi .class, Outer.class, Main.class dan Outer\$Inner.class.
- Program dapat berjalan dengan baik karena tidak ada error dan terdapat internal class Artinya, kelas dalam di dalam kelas luar. Kelas utama akan membuat objek Luar tanpa nama.

2.

```
package sesitujuh.dua;

public class Outer {
    int number = 0;
    void outerMethod(){
        class Inner{
            public void print(){
                System.out.println("mengakses inner class yang ke-: " + (number++));
            }
        }
        Inner inner = new Inner();
        inner.print();
    }
}

class OuterAccess{
    public static void main(String[] args) {
        Outer outer = new Outer();
        outer.outerMethod();
        outer.outerMethod();
        outer.outerMethod();
    }
}
```

- a. Ada perbedaan antara ++ number (awalan) dan number++ (akhiran).

- **++ number**

Saat bertambah, inisialisasi nomor variabel dengan nilai 0 metode cetak, ++ number akan menambahkan 1 ke variabel angka terlebih dahulu sebelum mencetak (muncul di konsol). menghasilkan : 1, 2, 3

```
mengakses inner class yang ke-: 1
mengakses inner class yang ke-: 2
mengakses inner class yang ke-: 3
BUILD SUCCESSFUL (total time: 0 seconds)
```

- **number ++**

Sebaliknya, dengan number ++, variabel angka dengan nilai 0 akan dipanggil Muncul melalui metode cetak sebelum melakukan penambahan menjadi : 0, 1, 2

```
mengakses inner class yang ke-: 0
mengakses inner class yang ke-: 1
mengakses inner class yang ke-: 2
BUILD SUCCESSFUL (total time: 0 seconds)
```

- b. Membuat 3 class yaitu Outer.class, Outer\$1Inner.class, OuterAccess.class