



## 1. Tujuan

- Membuat kelas mereka sendiri
- Mendeklarasikan atribut dan method pada class
- Menggunakan referensi this untuk mengakses instance data
- Membuat dan memanggil overloaded method
- Mengimport dan membuat package
- Menggunakan access modifiers untuk mengendalikan akses terhadap class member

## 2. Latar Belakang

Setelah kita mempelajari penggunaan class dari Java Class Library, kita akan mempelajari bagaimana menuliskan sebuah class sendiri. Pada bagian ini, untuk mempermudah pemahaman pembuatan class, kita akan membuat contoh class dimana akan ditambahkan beberapa data dan fungsi – fungsi lain.

Kita akan membuat class yang mengandung informasi dari Siswa dan operasi – operasi yang dibutuhkan pada record siswa.

Beberapa hal yang perlu diperhatikan pada syntax yang digunakan pada bab ini dan bagian lainnya :

- \* - Menandakan bahwa terjadi lebih dari satu kejadian dimana elemen tersebut diimplementasikan
- <description> - Menandakan bahwa Anda harus memberikan nilai pasti pada bagian ini
- [] - Indikasi bagian optional



## 3. Percobaan

### Percobaan 1 Membuat Class Student Record :

```
public class StudentRecord
{
private String name;
private String address;
private int age;
private double mathGrade;
private double englishGrade;
private double scienceGrade;
private double average;
private static int studentCount;
/**
 * Menghasilkan nama dari Siswa
 */
public String getName(){
return name;
}
/**
 * Mengubah nama siswa
 */
public void setName( String temp ){
name = temp;
}
// area penulisan kode lain
```



```
/**
 * Menghitung rata - rata nilai Matematik, Bahasa Inggris, * * Ilmu
 * Pasti
 */
public double getAverage(){
double result = 0;
result = ( mathGrade+englishGrade+scienceGrade )/3;
return result;
}
/**
 * Menghasilkan jumlah instance StudentRecord
 */
public static int getStudentCount(){
return studentCount;
}
}
```

## **Percobaan 2 Membuat Class Student Record Example :**

```
public class StudentRecordExample
{
public static void main( String[] args ){
//membuat 3 object StudentRecord
StudentRecord annaRecord = new StudentRecord();
StudentRecord beahRecord = new StudentRecord();
StudentRecord crisRecord = new StudentRecord();
//Memberi nama siswa
annaRecord.setName("Anna");
beahRecord.setName("Beah");
crisRecord.setName("Cris");
//Menampilkan nama siswa "Anna"
System.out.println( annaRecord.getName() );
//Menampilkan jumlah siswa
System.out.println("Count="+StudentRecord.getStudentCount()
);
}
}
```



>>> Java Education Network Indonesia

### **Hasil Percobaan 1 dan 2 :**

Output Class Student Record Example dengan memanfaatkan class Student Record:

Problems	Javadoc	Declaration
<terminated>	StudentRecordExa	
Anna		
Count=0		



### **Percobaan 3 Membuat Class Student Record yang telah dimodifikasi:**

```
public class StudentRecord {
    private String name;
    private String address;
    private int age;
    private double mathGrade;
    private double englishGrade;
    private double scienceGrade;
    private double average;
    private static int studentCount;
    public StudentRecord(){ //constructor default
        studentCount++;
    }
    public StudentRecord(String temp){
        this.name = temp;
        studentCount++;
    }
    public StudentRecord(String name, String Address){
        this.name = name;
        this.address = Address;
        studentCount++;
    }
}
```



```
public StudentRecord(double mGrade,double eGrade,double sGrade){
mathGrade = mGrade;
englishGrade = eGrade;
scienceGrade = sGrade;
studentCount++;
}
/**
 * Menghasilkan nama dari siswa
 */
public String getName(){
return name;
}
/**
 * mengubah nama siswa
 */
public void setName(String temp){
name = temp;
}
public String getAddress(){
return address;
}
public void setAddress(String temp){
address = temp;
}
public int getAge(){
return age;
}
```



```
}  
  
public void setAge(int temp){  
    age = temp;  
}  
  
//area penulisan kode lain  
  
/**  
 * Menghitung rata-rata nilai matematika, bahasa inggris, ilmu pengetahuan  
 */  
  
public double getAverage(){  
    double result =0;  
    result = (mathGrade+englishGrade+scienceGrade)/3;  
    return result;  
}  
  
public static int getStudentRecord(){  
    return studentCount;  
}  
  
public void print(String temp){  
    System.out.println("Name : "+name );  
    System.out.println("Address : "+address);  
    System.out.println("Age : "+age);  
}  
  
public void print(double eGrade, double mGrade, double sGrade){  
    System.out.println("Name : "+name);  
    System.out.println("Math Grade : "+mGrade);  
    System.out.println("English Grade : "+eGrade);
```





```
System.out.println("Science Grade : "+sGrade);
}
public double getEnglishGrade() {
return englishGrade;
}
public void setEnglishGrade(double englishGrade) {
this.englishGrade = englishGrade;
}
public double getMathGrade() {
return mathGrade;
}
public void setMathGrade(double mathGrade) {
this.mathGrade = mathGrade;
}
public double getScienceGrade() {
return scienceGrade;
}
public void setScienceGrade(double scienceGrade) {
this.scienceGrade = scienceGrade;
}
}
```

**Percobaan 4 Membuat Class Student Record Example yang telah dimodifikasi :**

```
public class StudentRecordExample {
public static void main(String[] args) {
//membuat 3 object StudentRecord
StudentRecord annaRecord = new StudentRecord();
StudentRecord beahRecord = new StudentRecord();
StudentRecord crisRecord = new StudentRecord();

//versi baru
StudentRecord recordAnna = new StudentRecord("Candra");
StudentRecord recordBeah = new StudentRecord("Bagus",
"Malang");
StudentRecord recordCris = new StudentRecord(80,90,100);

//memberi nama siswa
annaRecord.setName("Anna");
beahRecord.setName("Beah");
crisRecord.setName("Cris");

//menampilkan nama siswa "Anna"
System.out.println(annaRecord.getName());

//menampilkan jumlah siswa
System.out.println("Count="+StudentRecord.getStudentRecord());

StudentRecord anna2Record = new StudentRecord();
anna2Record.setName("Anna");
```



```
anna2Record.setAddress("Philipina");
anna2Record.setAge(15);
anna2Record.setMathGrade(80);
anna2Record.setEnglishGrade(95.5);
anna2Record.setScienceGrade(100);

//overload method
anna2Record.print(anna2Record.getName());
anna2Record.print(anna2Record.getEnglishGrade(),
anna2Record.getMathGrade(),
anna2Record.getScienceGrade());
recordAnna.print(recordAnna.getName());
}
}
```



### Hasil Percobaan 3 dan 4 :

Output membuat Class Student Record Example yang telah dimodifikasi dengan memanfaatkan Class Student Student Record yang telah dimodifikasi:

```
<terminated> StudentRecordExample [Java Applicati  
Anna  
Count=6  
Name : Anna  
Address : Philipina  
Age : 15  
Name : Anna  
Math Grade : 80.0  
English Grade : 95.5  
Science Grade : 100.0  
Name : Candra  
Address : null  
Age : 0
```

### Percobaan 5 Package :

```
package schoolClasses;  
  
public class StudentRecord  
{  
    private String name;  
    private String address;  
    private int age;  
}
```