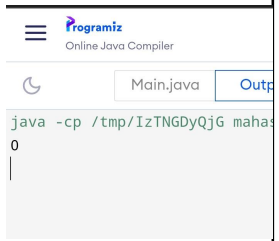
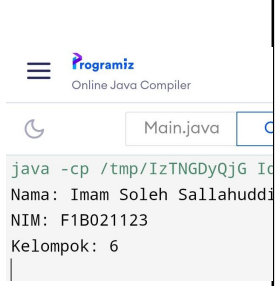

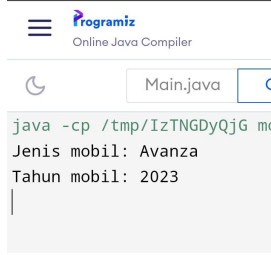



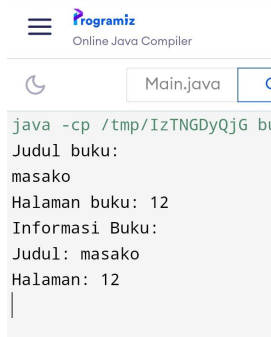
# Jobsheet Constructor


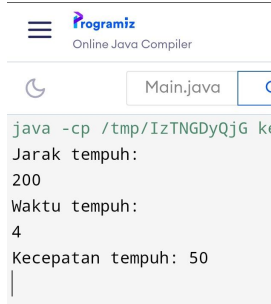
Nama : Imam Soleh Sallahuddin


Nim : F1B021123

No.	Kegiatan	Scrip	Hasil running
1.	Method constructor – membuat constructor tanpa parameter	<pre>public class mahasiswa {     int x;      public mahasiswa() {     }      public static void main(String[] args) {         mahasiswa mhs = new mahasiswa();         System.out.println(mhs.x);     } }</pre>	 <pre> Programiz Online Java Compiler  Main.java Output  java -cp /tmp/IzTNGDyQjG mahas 0   </pre>
2.	Method constructor – membuat constructor dengan parameter	<pre>public class Identitas {     String nama;     String nim;     int kelompok;      public Identitas(String nama, String nim, int kelompok) {         this.nama = nama;         this.nim = nim;         this.kelompok = kelompok;     }      public static void main(String[] args) {         Identitas mahasiswa = new Identitas("Imam Soleh Sallahuddin", "F1B021123", 6);         System.out.println("Nama: " + mahasiswa.nama);         System.out.println("NIM: " + mahasiswa.nim);         System.out.println("Kelompok: " + mahasiswa.kelompok);     } }</pre>	 <pre> Programiz Online Java Compiler  Main.java Output  java -cp /tmp/IzTNGDyQjG Id Nama: Imam Soleh Sallahuddi NIM: F1B021123 Kelompok: 6   </pre>
3.	Method constructor – parameter yang berbeda	<pre>public class nilai {     String mk;     int nilai;      public nilai(String mk, int nilai) {         this.mk = mk;         this.nilai = nilai;     } }</pre>	 <pre> Programiz Online Java Compiler  Main.java Output  java -cp /tmp/IzTNGDyQjG nilai Mata Kuliah: Pemrograman Berorientasi Ob Nilai: 80   </pre>

		<pre>     }      public static void main(String[] args) {         nilai mahasiswa = new         nilai("Pemrograman Berorientasi Objek", 80);         System.out.println("Mata Kuliah: " +         mahasiswa.mk);         System.out.println("Nilai: " +         mahasiswa.nilai);     } } </pre>	
4.	Method Constructor – gabungan	<pre> public class mobil {     private String nama;     private int tahun;      public mobil() {         nama = "Avanza";     }      public mobil(int tahun) {         this.tahun = tahun;     }      public String getNama() {         return nama;     }      public void setNama(String nama) {         this.nama = nama;     }      public int getTahun() {         return tahun;     }      public void setTahun(int tahun) {         this.tahun = tahun;     }      public static void main(String[] args) {         mobil mbl1 = new mobil();         mobil mbl2 = new mobil(2023);          System.out.println("Jenis mobil: " +         mbl1.getNama());         System.out.println("Tahun mobil: " +         mbl2.getTahun());     } } </pre>	

		}	
5.	Method constructor dengan 2 constructor	<pre> public class birthday {     String hari;     String bulan;     int tanggal;     int tahun;      public birthday() {         this.hari = "Senin";         this.tanggal = 1;         this.bulan = "Desember";         this.tahun = 2001;     }      public static void main(String[] args) {         birthday lahir1 = new birthday();          System.out.println("Hari: " + lahir1.hari);         System.out.println("Tanggal: " +         lahir1.tanggal);         System.out.println("Bulan: " +         lahir1.bulan);         System.out.println("Tahun: " +         lahir1.tahun);     } } </pre>	
6.	Method constructor dengan 2 constructor (input secara dinamis)	<pre> import java.util.Scanner;  public class buku {     String judul;     int halaman;      public buku(String judul, int halaman) {         this.judul = judul;         this.halaman = halaman;     }      public static void main(String[] args) {         Scanner input = new         Scanner(System.in);          System.out.println("Judul buku: ");         String judul = input.nextLine();          System.out.println("Halaman buku: ");         int halaman = input.nextInt(); </pre>	

		<pre>         buku myBook = new buku(judul,         halaman);          System.out.println("Informasi Buku:");         System.out.println("Judul: " +         myBook.judul);         System.out.println("Halaman: " +         myBook.halaman);     } } </pre>	
7.	Method constructor dengan perhitungan (menghitung debit air secara dinamis)	<pre> import java.util.Scanner;  public class debit {     int hitungDebit;      public debit(int volume, int waktu) {         hitungDebit = volume / waktu;     }      public int hitungDebit() {         return hitungDebit;     }      public static void main(String[] args) {         Scanner input = new Scanner(System.in);         System.out.println("Volume air: ");         int volume = input.nextInt();         System.out.println("Waktu: ");         int waktu = input.nextInt();          debit nilai = new debit(volume, waktu);         System.out.println("Debit air: " +         nilai.hitungDebit);     } } </pre>	
8.	Methode constructor dengan perhitungan (menghitung kecepatan mobil secara dinamis)	<pre> import java.util.Scanner;  public class kecepatan {     int hitungKecepatan;      public kecepatan(int jarak, int waktu) {         hitungKecepatan = jarak / waktu;     }      public int hitungKecepatan() { </pre>	

		<pre> return hitungKecepatan; }  public static void main(String[] args) {     Scanner input = new Scanner(System.in);     System.out.println("Jarak tempuh: ");     int jarak = input.nextInt();     System.out.println("Waktu tempuh: ");     int waktu = input.nextInt();      kecepatan hasil = new kecepatan(jarak, waktu);     System.out.println("Kecepatan tempuh: " + hasil.hitungKecepatan()); } } </pre>	
9.	<p>Methode constructor dengan perhitungan (menghitung energi potensial secara dinamis)</p>	<pre> import java.util.Scanner;  public class potensial {     double energiPotensial;     final double gravitasi = 9.81;      public potensial(double massa, double tinggi) {         energiPotensial = massa * gravitasi * tinggi;     }      public double getEnergiPotensial() {         return energiPotensial;     }      public static void main(String[] args) {         Scanner input = new Scanner(System.in);         System.out.println("Massa benda: ");         double massa = input.nextDouble(); // Menggunakan nextDouble untuk input nilai desimal         System.out.println("Tinggi: ");         double tinggi = input.nextDouble(); // Menggunakan nextDouble untuk input nilai desimal          potensial energi = new potensial(massa, tinggi);         double hasilEnergiPotensial = energi.getEnergiPotensial(); </pre>	 <p>The screenshot shows the Programiz Online Java Compiler interface. The code area contains a Java program that calculates potential energy. The program takes two inputs: 'Massa benda' (900) and 'Tinggi' (300). The output is 'Energi Potensial: 2648700.0'.</p>

		<pre>        System.out.println("Energi Potensial: " +         hasilEnergiPotensial);     } }</pre>	
--	--	---	--