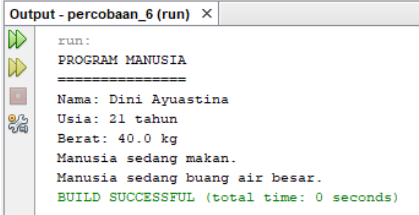
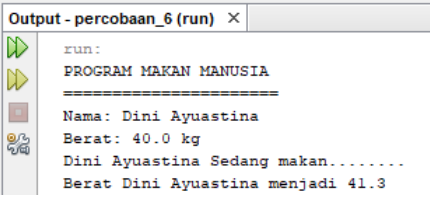
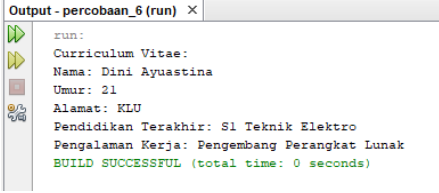






no	Latiha n	Script	Hasil
1	Inherit ance secara static	<pre>package dini_ayuastina.Js1;  public class js1{     public static void     main(String[] args) {         makan mnsi = new makan();          System.out.println("PROGRAM         MANUSIA");         System.out.println("=====         ====");          mnsi.nama = "Dni Ayuastina         mnsi.usia = 21;         mnsi.berat = 40;          mnsi.tampil();         mnsi.makan();         mnsi.pup();         }         }</pre>	
		<pre>package dini_ayuastina.Js1;  public class makan extends         manusia {          void pup(){             System.out.println(nama+"             Sedang puupp.....");         }          void makan(){             System.out.println(nama+"             Sedang makan.....");         }         }</pre>	
		<pre>package dini_ayuastina.Js1;  public class manusia{     String nama;     int usia;     double berat;      void tampil(){         System.out.println("Namanya:         "+this.nama);         System.out.println("Usianya:         "+this.usia);         System.out.println("Beratnya:         "+this.berat);     } }</pre>	

		<pre> } } </pre>	
		<pre> package dini_ayuastina.Js2;  public class js2 {     public static void     main(String[] args) {         makan msia = new makan();          System.out.println("PROGRAM         MANUSIA");         System.out.println("=====         ====");          msia.nama = "Dini Ayuastina";         msia.usia = 21;         msia.berat = 40;          msia.tampil();         msia.makan();         msia.pup();     } } </pre>	
2	Penggunaan inheritance dengan modifier	<pre> package dini_ayuastina.Js2;  public class js2 {     public static void     main(String[] args) {         makan msia = new makan();          System.out.println("PROGRAM         MANUSIA");         System.out.println("=====         ====");          msia.nama = "Dini Ayuastina";         msia.usia = 21;         msia.berat = 40;          msia.tampil();         msia.makan();         msia.pup();     } } </pre>	

		<pre> package dini_ayuastina.Js2;  public class makan extends manusia { void pup(){ System.out.println(nama+" Sedang puupp....."); berat = berat - 0.40; System.out.println("Berat "+nama+" menjadi "+berat); }  void makan(){ System.out.println(nama+" Sedang makan....."); berat = berat + 1.3; System.out.println("Berat "+nama+" menjadi "+berat); } } </pre>	
		<pre> package dini_ayuastina.Js2;  public class manusia { String nama; int usia; double berat;  void tampil(){ System.out.println("Namanya: "+this.nama); System.out.println("Usianya: "+this.usia); System.out.println("Beratnya: "+this.berat); } } </pre>	

3	Inheritance secara statis dengan setter getter	<pre> public class Manusia {     protected int umur;      public void setUmur(int umur) {         this.umur = umur;     }      public int getUmur() {         return umur;     } }  class CV extends Manusia {     private String nama;     private String alamat;     private String pendidikanTerakhir;     private String pengalamanKerja;      public CV(String nama, String alamat, String pendidikanTerakhir, String pengalamanKerja) {         this.nama = nama;         this.alamat = alamat;         this.pendidikanTerakhir = pendidikanTerakhir;         this.pengalamanKerja = pengalamanKerja;     }      public void displayCV() {         System.out.println("Curriculum Vitae:");          System.out.println("Nama: " + nama);          System.out.println("Umur: " + umur);          System.out.println("Alamat: " + alamat);          System.out.println("Pendidikan Terakhir: " + pendidikanTerakhir);          System.out.println("Pengalaman Kerja: " + pengalamanKerja);     }      public static void main(String[] args) {         CV cv = new CV("Dini Ayuastina", "KLU", "S1 Teknik Elektro", "Pengembang Perangkat Lunak");     } } </pre>	
---	--	--	--

		<pre>         cv.setUmur(21);          cv.displayCV();     } } </pre>	
4	Inheritance menggunakan unakan lebih dari 2 class	<pre> public class Bapak {     public static void makan()     {         System.out.println("Bapak makan");     } }  class Anak extends Bapak {     public static void menangis() {          System.out.println("Anak menangis");     } }  class Pewarisan {      public static void main(String[] args) {         Bapak dataSatu = new Bapak();         Anak dataDua = new Anak();          Bapak.makan();         Anak.menangis();          dataDua.makan();     } } </pre>	<div>Output - percobaan_6 (run) ×</div> <pre> run: Bapak makan Anak menangis Bapak makan BUILD SUCCESSFUL (total time: 0 seconds)   </pre>

10	Pengg unaan abstra ct class dan abstra ct metho d secara dinami s	<pre> public abstract class Hewan {     abstract void setName();     abstract void setMakanan();     abstract void setAksi();     abstract void setUmur();     abstract void setBerat();     abstract void setJenis();     abstract void setHabitat();     abstract void setWarna();     abstract void setKaki();     abstract void setSuara();     abstract void setHidupDiAir();     abstract void setHidupDiDarat();     abstract void setHidupDiUdara(); }  public class Kelinci extends Hewan {     public void setName() {  System.out.println("Nama hewan adalah \"KELINCI\");     }      public void setMakanan() {  System.out.println("Makanan kelinci adalah \"WORTEL\");     }      public void setAksi() {  System.out.println("Kelinci dapat \"HOP-HOP\");     }     public void setUmur() {  System.out.println("Kelinci berumur \"2 tahun\");     }     public void setBerat() {  System.out.println("Beratnya \"1.5 kg\");     }     public void setJenis() {  System.out.println("Bejenis \"Herbivora\");     }     public void setHabitat() {  System.out.println("Hbitatnya \"Di Darat\");     }     public void setWarna() { </pre>	<div> <div>Output - percobaan_6 (run) X</div> <div>     </div> <div> <pre> run: Nama hewan adalah "KELINCI" Makanan kelinci adalah "WORTEL" Kelinci dapat "HOP-HOP" Kelinci berumur "2 tahun" Beratnya "1.5 kg" Bejenis "Herbivora" Hbitatnya "Di Darat" Kelinci itu berwarna "Putih" Jumlah Kaki "4" Suaranya "Cicit-cicit" Hidup Di Air "false" Hidup Di Udara "true" BUILD SUCCESSFUL (total time: 0 seconds)   </pre> </div> </div>
----	---	--	---

		<pre> System.out.println("Kelinci itu berwarna \"Putih\");     }     public void setKaki() {  System.out.println("Jumlah Kaki \"4\");     }     public void setSuara() {  System.out.println("Suaranya \"Cicit-cicit\");     }     public void setHidupDiAir() {  System.out.println("Hidup Di Air \"false\");     }     public void setHidupDiUdara() {  System.out.println("Hidup Di Udara \"true\");     }      public static void main(String[] args) {         Kelinci kelinci = new Kelinci();         kelinci.setNama();         kelinci.setMakanan();         kelinci.setAksi();         kelinci.setUmur();         kelinci.setBerat();         kelinci.setJenis();         kelinci.setHabitat();         kelinci.setWarna();         kelinci.setKaki();         kelinci.setSuara();  kelinci.setHidupDiAir();  kelinci.setHidupDiUdara();     }      @Override     void setHidupDiDarat() {         throw new UnsupportedOperationException(" Not supported yet."); //To change body of generated methods, choose Tools   Templates.     } } </pre>	
--	--	--	--