**Soal Pertama**Buatlah sebuah package dengan nama *kampus* yang didalamnya ada Class Mahasiswa dan Dosen, dan diluar package terdapat sebuah Class Main cobalah lakukan pemanggilan kedua Class tersebut didalam Class Main menggunakan

1. Metode *Import All*
2. Metode *Import Each*
3. Metode *Fully Qualified Name*

**Jawaban**

main

|  |
| --- |
| 1. **import** kampus.\*; *//import all* 2. **import** kampus.Mahasiswa; *//import each* 3. **import** kampus.Dosen; *//import each* 4. **import** kampus.baru.mahasiswi;  7. **public** **class** Main { 8. **public** **static** **void** main(String[] args) { 9. System.out.println("Hello world!"); 10. kampus.Mahasiswa mahasiswa = **new** kampus.Mahasiswa(); *//fully qualified name* 11. kampus.Dosen dosen = **new** kampus.Dosen(); *//fully qualified name* 12. } 13. } |

Mahasiswi

|  |
| --- |
| 1. **package** kampus.baru; 3. **public** **class** mahasiswi { 4. **public** **void** test(){ 5. System.out.println("coba"); 6. } 7. } |

Mahasiswa

|  |
| --- |
| 1. **package** kampus; 3. **public** **class** Mahasiswa { 4. **int** nama; 5. } |

Dosen

|  |
| --- |
| 1. **package** kampus; 3. **public** **class** Dosen { 4. **int** namaDosen; 5. } |

**Soal Kedua**Perhatikan Kasus berikut.

Asumsikan bahwa kita telah menulis beberapa Class didalam package mycalculator. Tetapi kita memutuskan bahwa mereka harus dibagi lagi menjadi tiga package, seperti yang tercantum dibawah ini.

1. Baris kode apa yang perlu kita ubuah ke setiap file java setelah menempatkan ke dalam package yang baru?
2. Menurut kamu, apakah perlu membuat perubahan lain pada baris file Main agar dapat dikomplikasi dengan benar? Jika demikian, apa yang perlu dirubah?

**Jawaban**

1. Yang pergegi.java ditambah atasnya package mycalculator.bangundatar

Yang kubus.java ditambah atasnya package mycalculator.bangunruang

Yang deretAngka.java ditambah atasnya package mycalculator.deretangka

1. Perlu, karena kita harus mendifinisikan package yang baru beserta class yang ada didalamnya agar bisa dipanggil.

**Soal Ketiga**Untuk mengakses suatu data class yang dibungkus (encapsulationp) kita dapat menerapkan konsep Acces Modifier

Cobalah untuk membuat sebuah kode program sederhana yang menerapkan :

1. Acces Modifier Protected, dan
2. Acces Modifier Private

Serta jelaskan secara singkat program yang sudah kalian buat.

**Jawaban**

1. Protected

Main

|  |
| --- |
| 1. **package** protected\_access; 3. **public** **class** Main { 4. **public** **static** **void** main(String[] args) { 5. HelloProtected hProtected; 6. hProtected = **new** HelloProtected(); 8. hProtected.helloworld(); 9. hProtected.hellodata(); 10. } 11. } |

HelloProtected

|  |
| --- |
| 1. **package** protected\_access; 3. **public** **class** HelloProtected { 4. **protected** **int** data =10; 5. **protected** **void** helloworld(){ 6. System.out.println("Protect : hello world"); 7. } 8. **public** **void** hellodata(){ 9. helloworld(); 10. System.out.println("Protect : data = "+data); 11. } 12. } |

**Output**

|  |
| --- |
|  |

Penjelasan :yang mana jika memakai protected ini sama aja dengan modifier default, yaitu masih bisa diakses dan tidak terjadi eror, akan tetapi jika packagenya beda baru tidak bisa dan akan terjadi eror

1. Private

Main

|  |
| --- |
| 1. **package** private\_acces; 3. **public** **class** Main { 4. **public** **static** **void** main(String[] args) { 5. Helloprivate hprivate = **new** Helloprivate(); 7. hprivate.helloData(); 8. } 9. } |

Helloprivate

|  |
| --- |
| 1. **package** private\_acces; 3. **public** **class** Helloprivate { 4. **private** **int** data=40; 6. **private** **void** helloWorld(){ 7. System.out.println("Private : hello world"); 8. } 9. **public** **void** helloData(){ 10. helloWorld(); 11. System.out.println("Private: data ="+data); 12. } 13. } |

Output

|  |
| --- |
|  |

Penjelasan : yaitu modifier pada private tidak bisa diakses jika diluar class dia sendiri, dan hanya bisa diakses oleh class dia sendiri

**Soal Keempat**

Perhatikan baris kode dibawah ini.

|  |
| --- |
| 1. **public** **class** Adayangsalah { 2. **private** String nama; 3. **private** String getNama(){ 4. **return** nama; 5. } 6. } |

Apa yang salah dari baris kode diatas? Berikan alas an kalian dan tuliskan baris kode yang benar !

**Jawaban**

|  |
| --- |
| 1. **public** **class** Adayangsalah { 2. **private** String nama; 3. **public** void setNama(String nama) {  this.nama = nama; } 4. **public** String getNama(){ 5. **return** nama; 6. } 7. } |

Penjelasan, karena kalua setter getter itu harus public bukan private, dikarenakan method ini akan diakses diluar class

**Soal kelima**

***PROGRES STUDI KASUS PROJECT AKHIR:***

Berdasarkan Studi Kasus yang telah kalian dapatkan, cobalah untuk memisahkan atau mengelompokkan beberapa class ke dalam sebuah package yang sesuai, dan ubalah juga cara import dari class yang sudah masuk kedalam package pada class yang memanggilnya, selain itu juga tambahkan Acces Modifier ke attribute dan method yang telah kalian buat !

**Jawaban**

Main

|  |
| --- |
| 1. **package** TA; 3. **import** java.util.Scanner; 5. **public** **class** Main { 6. **public** **static** **void** main(String[] args) **throws** Exception { 7. Data obj1 = **new** Data(); 8. Scanner inp = **new** Scanner(System.in); 9. **do** { 10. System.out.println("Menu Pilihan"); 11. System.out.println("1. Daftar Akun"); 12. System.out.println("2. Login "); 13. System.out.println("3. Edit "); 14. System.out.println("4. Hapus"); 15. System.out.println("5. lihat "); 16. System.out.println("6. Exit"); 17. System.out.print("pilih : "); 18. obj1.pilih = inp.nextInt(); 19. **switch** (obj1.pilih){ 20. **case** 1: 21. obj1.Daftar(); 22. **break**; 23. **case** 2: 24. obj1.Login(); 25. **break**; 26. **case** 3: 27. obj1.Update(); 28. **break**; 29. **case** 4: 30. obj1.delete(); 31. **break**; 32. **case** 5: 33. obj1.display(); 34. } 35. } **while** (obj1.pilih != 6); 36. } 37. } |

data1

|  |
| --- |
| 1. **package** TA;  4. **public** **class** data1 { 5. String kursus; 6. data1(String kursus){ 7. **this**.kursus = kursus; 8. } 9. **public** String getKursus() { 10. **return** kursus; 11. } 12. } |

Data

|  |
| --- |
| 1. **package** TA; 3. **import** java.util.Scanner; 4. **import** java.util.ArrayList; 5. **public** **class** Data { 6. Scanner inp = **new** Scanner(System.in); 7. **public** **int** pilih; 8. ArrayList<data1> arr = **new** ArrayList<>(); 9. ArrayList<data2> arrReg = **new** ArrayList<>(); 10. **int** idxLogin; 11. **public** **int** inpDatakursus; 12. **public** String Fname , Lname , Alamat, Tanggal ; 13. **public** String pw , telp; 14. **public** **boolean** log = **false** ; 15. *//getter* 16. **public** String getAlamat() { 17. **return** Alamat; 18. } 19. **public** String getFname() { 20. **return** Fname; 21. } 22. **public** String getLname() { 23. **return** Lname; 24. } 25. **public** String getTanggal() { 26. **return** Tanggal; 27. } 28. **public** String getPw() { 29. **return** pw; 30. } 31. **public** String getTelp() { 32. **return** telp; 33. } 35. **public** **void** Update(){ 36. **if** (log == **true**){ 37. System.out.println("**\n**SISTEM EDIT**\n**"); 38. System.out.println("akun " + arrReg.get(idxLogin).getFnama() + " " + arrReg.get(idxLogin).getLname() + " yang akan diedit"); 39. System.out.println("============================="); 40. System.out.print("nama depan : "); 41. Fname = inp.next(); 42. Fname = getFname(); 43. System.out.print("nama belakang : "); 44. Lname = inp.next(); 45. Lname = getLname(); 46. System.out.print("Alamat : "); 47. Alamat = inp.next(); 48. Alamat = getAlamat(); 49. System.out.print("No telp : "); 50. telp = inp.next(); 51. telp = getTelp(); 52. System.out.print("Password : "); 53. pw = inp.next(); 54. pw = getPw(); 55. viewDatakursus(); 56. System.out.print("pilih kursus : "); 57. inpDatakursus = inp.nextInt(); 58. System.out.println("============================="); 59. System.out.println("nama lengkap : " + getFname()+ " " + getLname()); 60. System.out.println("Alamat : " + getAlamat()); 61. System.out.println("No telp : " + getTelp()); 62. **if**(inpDatakursus == 1){ 63. System.out.println("kursus : " + arr.get(inpDatakursus - 1).getKursus()); 64. } 65. **else** **if** (inpDatakursus == 2 ){ 66. System.out.println("kursus : " + arr.get(inpDatakursus - 1).getKursus()); 67. } 68. **else** **if** (inpDatakursus == 3 ){ 69. System.out.println("kursus : " + arr.get(inpDatakursus - 1).getKursus()); 70. }System.out.println("============================="); 72. arrReg.set(idxLogin, **new** data2(Fname, Lname, Alamat, telp, pw, inpDatakursus)); 73. }**else** **if**(log == **false**){ 74. System.out.println("LOGIN terlebih dahulu yaa hehe"); 75. }**else** **if**(arrReg.isEmpty()){ 76. System.out.println("DAFTAR DULU WOE"); 77. } 78. } 80. **public** **void** Daftar(){ 81. System.out.println("========================================="); 82. System.out.println("Daftar Akun kursus **\n**"); 83. System.out.print("nama depan : "); 84. Fname = inp.next(); 85. Fname = getFname(); 86. System.out.print("nama belakang : "); 87. Lname = inp.next(); 88. Lname = getLname(); 89. System.out.print("Alamat : "); 90. Alamat = inp.next(); 91. Alamat = getAlamat(); 92. System.out.print("No telp : "); 93. telp = inp.next(); 94. telp = getTelp(); 95. System.out.print("Password : "); 96. pw = inp.next(); 97. pw = getPw(); 98. dataKursus(); 99. viewDatakursus(); 100. System.out.print("pilih kursus : "); 101. inpDatakursus = inp.nextInt(); 102. arrReg.add(**new** data2(Fname, Lname, Alamat, telp, pw,inpDatakursus)); 103. System.out.println("**\n**AKUN ANDA BERHASIL DIBUAT"); 104. System.out.println("============================="); 105. System.out.println("nama lengkap : " + getFname()+ " " + getLname()); 106. System.out.println("Alamat : " + getAlamat()); 107. System.out.println("No telp : " + getTelp()); 108. **if**(inpDatakursus == 1){ 109. System.out.println("kursus : " + arr.get(inpDatakursus - 1).getKursus()); 110. } 111. **else** **if** (inpDatakursus == 2 ){ 112. System.out.println("kursus : " + arr.get(inpDatakursus - 1).getKursus()); 113. } 114. **else** **if** (inpDatakursus == 3 ){ 115. System.out.println("kursus : " + arr.get(inpDatakursus - 1).getKursus()); 116. } 117. } 119. **public** **void** viewDatakursus(){ 120. System.out.println("1. " + arr.get(0).getKursus()); 121. System.out.println("2. " + arr.get(1).getKursus()); 122. System.out.println("3. " + arr.get(2).getKursus()); 123. } 124. **public** **void** display() { 125. **if**(arrReg.isEmpty() == **false**){ 126. **for**(**int** i = 0 ; i < arrReg.size() ; i++){ 127. System.out.println("account ke-" + (i+1)); 128. System.out.println("============================="); 129. System.out.println("Nama lengkap : " + arrReg.get(i).getFnama()+ " " + arrReg.get(i).getLname()); 130. System.out.println("Alamat : " + arrReg.get(i).getAddrs()); 131. System.out.println("No telp : " + arrReg.get(i).getNotlp()); 132. **if**(inpDatakursus == 1){ 133. System.out.println("kursus : " + arr.get(inpDatakursus - 1).getKursus()); 134. } 135. **else** **if** (inpDatakursus == 2 ){ 136. System.out.println("kursus : " + arr.get(inpDatakursus - 1).getKursus()); 137. } 138. **else** **if** (inpDatakursus == 3 ){ 139. System.out.println("kursus : " + arr.get(inpDatakursus - 1).getKursus()); 140. }System.out.println("============================="); 141. } 142. } 143. **else** **if** (arrReg.isEmpty() == **true**){ 144. System.out.println("**\n**TIDAK ADA AKUN YANG DIISI"); 145. System.out.println("KOSONG"); 146. } 147. } 148. **public** **int** l; 149. **public** **void** Login() { 150. String login; 151. String pwS; 152. System.out.print("nama depan : "); 153. login = inp.next(); 154. System.out.print("Password : "); 155. pwS = inp.next(); 156. **if** (arrReg.isEmpty()){ 157. System.out.println("Anda Belum Daftar"); 158. } 159. **for**( l = 0 ; l < arrReg.size(); l++){ 160. **if** (login.equals(arrReg.get(l).getFnama()) && pwS.equals(arrReg.get(l).getPw())){ 161. { 162. System.out.println("**\n**--ANDA SUDAH LOGIN--"); 163. System.out.println("============================="); 164. System.out.println("nama lengkap : " + arrReg.get(l).getFnama()+ " " + arrReg.get(l).getLname()); 165. System.out.println("Alamat : " + arrReg.get(l).getAddrs()); 166. System.out.println("No telp : " + arrReg.get(l).getNotlp()); 167. System.out.println("kursus : " + arr.get(inpDatakursus-1).getKursus()); 168. log = **true**; 169. idxLogin = l; 170. } 172. }**else** **if**(login != arrReg.get(l).getFnama() && pwS != arrReg.get(l).getPw()){ 173. System.out.println("data yang anda masukkan salah"); 174. } 175. } 176. } 177. **void** delete(){ 178. **if** (log == **true**){ 179. System.out.println("============================="); 180. System.out.println("ini adalah sistem DELETE"); 181. System.out.println("apakah anda ingin menghapus akun " + arrReg.get(idxLogin).getFnama() 182. + " " + arrReg.get(idxLogin).getLname() + " (y/t)?"); 183. String y = inp.next(); 184. **if** (y.equals("y")){ 185. arrReg.remove(idxLogin); 186. }**else** **if**(y.equals("t")){ 187. System.out.println("Cancelled"); 188. } 189. }**else** **if** (log == **false**){ 190. System.out.println("Daftar terlebih dahulu"); 191. }**else** **if**(arrReg.isEmpty()){ 192. System.out.println("Daftar telebih dahulu"); 193. } 194. } 195. **void** dataKursus(){ 196. String dataK[] = {"Matematika" , "Fisika" , "Kimia"}; 197. **for** (**int** i = 0 ; i < dataK.length ; i++){ 198. arr.add(**new** data1(dataK[i])); 199. } 200. } 201. } |

data2

|  |
| --- |
| 1. **package** TA;  4. **public** **class** data2 **extends** induk { 5. **public** data2(String fname , String lnama , String adres , String tlp , 6. String pw , **int** inpdtkrs ){ 7. **super**.fnama = fname; 8. **super**.lname = lnama; 9. **super**.addrs = adres; 10. **super**.notlp = tlp; 11. **super**.pw = pw; 12. **super**.inpdtkrs = inpdtkrs; 13. } 14. **public** String getAddrs() { 15. **return** addrs; 16. } 17. **public** String getFnama() { 18. **return** fnama; 19. } 20. **public** **int** getInpdtkrs() { 21. **return** inpdtkrs; 22. } 23. **public** String getLname() { 24. **return** lname; 25. } 26. **public** String getNotlp() { 27. **return** notlp; 28. } 29. **public** String getPw() { 30. **return** pw; 31. } 32. } 33. **class** induk { 34. **public** String fnama , lname , addrs , notlp , pw ; 35. **public** **int** inpdtkrs; 36. } |

**Output**

|  |
| --- |
|  |