



NAMA : Dio Aulia Ari Kurnia Sandi  
NIM :2041720086  
KELAS :2C  
MATERI :Jobsheet 11

#### Pertanyaan Percobaan 1

1. Class apa sajakah yang merupakan turunan dari class Employee?  
Jawab:  
InternshipEmployee dan PermanentEmployee
2. Class apa sajakah yang implements ke interface Payable?  
Jawab:  
PermanentEmployee dan ElectricityBill
3. Perhatikan class Tester1, baris ke-10 dan 11. Mengapa e, bisa diisi dengan objek pEmp (merupakan objek dari class PermanentEmployee) dan objek iEmp (merupakan objek dari class InternshipEmployee) ?  
Jawab:  
Karena kedua class tersebut extend pada class employee dimana e adalah object dari class employee
4. Perhatikan class Tester1, baris ke-12 dan 13. Mengapa p, bisa diisi dengan objek pEmp (merupakan objek dari class PermanentEmployee) dan objek eBill (merupakan objek dari class ElectricityBill) ?  
Jawab:  
Karena kedua class tersebut implements pada class Payable dimana p adalah object dari class Payable
5. Coba tambahkan sintaks: p = iEmp; e = eBill; pada baris 14 dan 15 (baris terakhir dalam method main) ! Apa yang menyebabkan error  
Jawab:  
Karena class yang diwakili oleh object iEmp tidak implements pada class Payable, dan class yang diwakili oleh object eBill tidak extend pada class Employee
6. Ambil kesimpulan tentang konsep/bentuk dasar polimorfisme  
Jawab:  
Polimorfisme terjadi ketika ada referensi super class yang digunakan untuk merujuk ke objek dari sub class

#### Pertanyaan Percobaan 2

1. Perhatikan class Tester2 di atas, mengapa pemanggilan e.getEmployeeInfo() pada baris 8 dan pEmp.getEmployeeInfo() pada baris 10 menghasilkan hasil sama?  
Jawab:  
Karena pada baris 7 telah dideklarasikan bahwa e=pEmp
2. Mengapa pemanggilan method e.getEmployeeInfo() disebut sebagai pemanggilan method virtual (virtual method invocation), sedangkan pEmp.getEmployeeInfo() tidak?  
Jawab:  
Karena method yang asli yaitu pEmp.getEmployeeInfo sedangkan e.grtEmployeeInfo hanyalah tiruannya dari deklarasi baris ke 7
3. Jadi apakah yang dimaksud dari virtual method invocation? Mengapa disebut virtual?  
Jawab:  
Virtual Method Invocation adalah jika terjadi polimorfisme dan overriding. Pada saat obyek yang sudah dibuat tersebut memanggil overridden method pada parent class, kompiler Java akan melakukan invocation (pemanggilan) terhadap overriding method pada subclass, dimana yang seharusnya dipanggil adalah overridden method



NAMA : Dio Aulia Ari Kurnia Sandi  
NIM :2041720086  
KELAS :2C  
MATERI :Jobsheet 11

#### Pertanyaan Percobaan 3

1. Perhatikan array e pada baris ke-8, mengapa ia bisa diisi dengan objekobjek dengan tipe yang berbeda, yaitu objek pEmp (objek dari PermanentEmployee) dan objek iEmp (objek dari InternshipEmployee) ?

Jawab:

Karena class-class yang diwakili oleh object pEmp dan iEmp extend pada class Employee yang diwakili oleh object e

2. Perhatikan juga baris ke-9, mengapa array p juga diisi dengan objek-objek dengan tipe yang berbeda, yaitu objek pEmp (objek dari PermanentEmployee) dan objek eBill (objek dari ElectricityBilling) ?

Jawab:

Karena class-class yang diwakili oleh object pEmp dan eBill implement pada class Payable yang diwakili oleh object p

3. Perhatikan baris ke-10, mengapa terjadi error?

Jawab:

Karena class Employee tidak berhubungan dengan class eBill

#### Pertanyaan Percobaan 4

1. Perhatikan class Tester4 baris ke-7 dan baris ke-11, mengapa pemanggilan ow.pay(eBill) dan ow.pay(pEmp) bisa dilakukan, padahal jika diperhatikan method pay() yang ada di dalam class Owner memiliki argument/parameter bertipe Payable?

Jawab:

Karena eBill adalah object dari class Payable

2. Jadi apakah tujuan membuat argument bertipe Payable pada method pay() yang ada di dalam class Owner?

Jawab:

Karena pembayaran dilakukan menggunakan dana kantor yaitu class eBill

3. Coba pada baris terakhir method main() yang ada di dalam class Tester4 ditambahkan perintah ow.pay(iEmp); Mengapa terjadi error?

Jawab:

Karena class yang diwakili oleh object iEmp tidak implements pada class Payable

4. Perhatikan class Owner, diperlukan untuk apakah sintaks p instanceof ElectricityBill pada baris ke-6 ?

Jawab:

Untuk melakukan pemilihan pembayaran listrik

5. Perhatikan kembali class Owner baris ke-7, untuk apakah casting objek disana (ElectricityBill eb = (ElectricityBill) p) diperlukan ? Mengapa objek p yang bertipe Payable harus di-casting ke dalam objek eb yang bertipe ElectricityBill ?

Jawab:

Karena terdapat 2 pilihan sehingga harus dicasting untuk menentukan pilihan tersebut