

Waktu Pengerjaan: 2 jam 30 menit

### Petunjuk Pengumpulan Jawaban

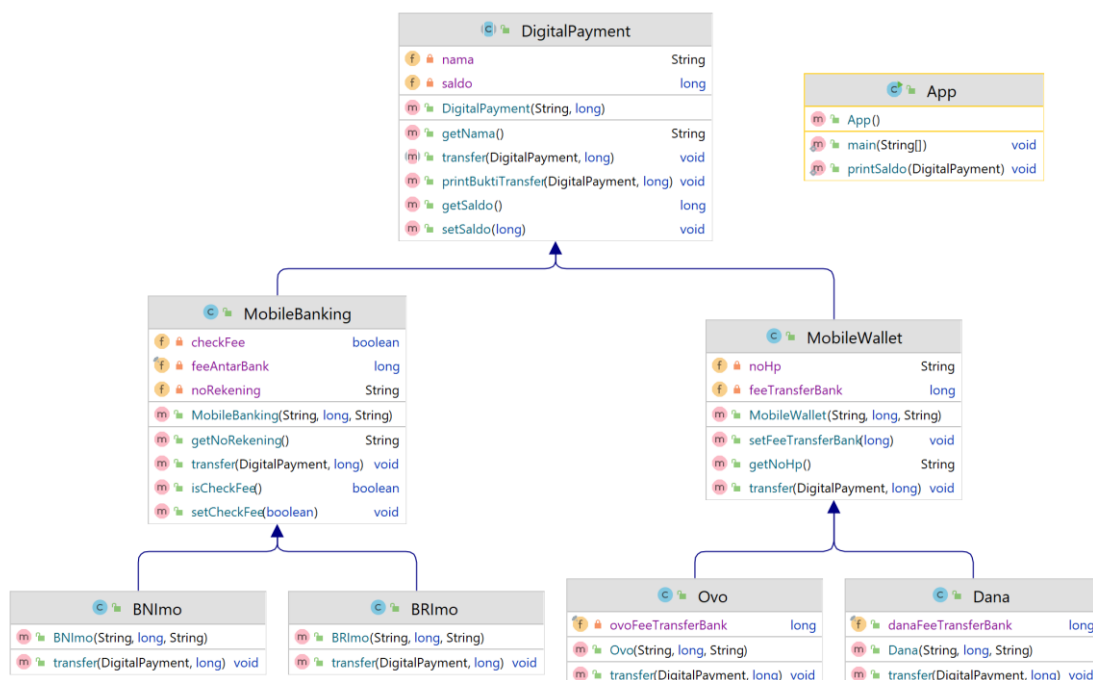
Buatlah folder bernama **UG9\_NIM** sebagai tempat untuk menyimpan jawaban kamu. Selanjutnya, buatlah project sesuai dengan ketentuan soal. Simpan project tersebut ke dalam folder yang baru saja kamu buat. Upload/push folder **UG9\_NIM** berisi project jawaban kamu ke repository UG\_RPLBO\_NIM\_GRUP yang sudah ada di Github kamu.

### Soal (100 Poin)

Buatlah sebuah project Maven dengan ketentuan:

- **GroupId:** com.ug9.etransactionproject
- **Nama Project/ArtifactId:** **NIM**-UG9-eTransactionProject --> Pastikan ini menjadi folder yang berisi jawaban project kamu!
- **Archetype:** maven-archetype-quickstart

Digital payment yang biasa kita gunakan umumnya diklasifikasikan ke dalam dua kategori, yaitu Mobile Banking (e-Banking) dan Mobile Wallet (e-Wallet). Kamu sebagai seorang programmer yang handal diminta oleh perusahaan untuk membuat proyek bernama e-Transaction. Proyek ini memiliki prinsip kerja yang menyerupai QRIS dimana seseorang dapat melakukan transfer ke berbagai macam bentuk digital payment. Perhatikan gambar class diagram di bawah ini untuk membantu kamu dalam mengerjakan proyek tersebut.



Buatlah class yang diperlukan sesuai dengan gambar class diagram di atas. Berikut adalah detail terkait masing-masing class yang perlu kamu perhatikan:

Abstract Class DigitalPayment	
<b>Keterangan:</b> Class ini bersifat abstract dan <b>sudah tersedia di dalam arsip soal</b> . Tujuan dari class ini yaitu menggeneralisasi seluruh bentuk objek alat pembayaran digital.	
DigitalPayment(String nama, long saldo)	Menginisialisasi atribut nama dan saldo sesuai dengan parameter
transfer(DigitalPayment dp, long nominal)	Method ini bersifat abstract dan berfungsi untuk melakukan transfer antar alat pembayaran digital
printBuktiTransfer(DigitalPayment penerima, long nominal)	Method ini berfungsi layaknya bukti transfer dan akan dipanggil apabila proses transfer sukses.

Class MobileBanking	
<b>Keterangan:</b> Salah satu bentuk dari alat pembayaran digital yaitu Mobile Banking (e-Banking).	
noRekening	Atribut ini merupakan identitas untuk akun mobile banking
checkFee	Atribut ini memiliki <b>nilai default</b> yaitu <b>false</b> dan bertujuan untuk memberikan keputusan terkait penambahan fee dalam proses transfer.
feeAntarBank	Atribut ini memiliki <b>nilai default</b> yaitu <b>6000</b> dan berfungsi untuk tambahan biaya apabila terjadi proses transfer antar bank. <b>*Keterangan: Antar bank sama dengan beda bank</b>
MobileBanking(String nama, long saldo, String noRekening)	Menginisialisasi atribut nama, saldo, dan noRekening sesuai dengan parameter.
transfer(DigitalPayment dp, long nominal)	Method ini merupakan override dari method abstract yang ada pada class DigitalPayment. Berikut beberapa hal yang perlu kamu perhatikan: <ul style="list-style-type: none"> <li>• Apabila saldo kurang dari 0, maka transfer gagal dan tampilkan pesan input tidak valid</li> <li>• Apabila saldo tidak mencukupi, maka transfer gagal dan tampilkan pesan kegagalan</li> <li>• Apabila atribut <b>checkFee bernilai true</b>, maka <b>saldo pengirim akan dikurangi dengan nominal dan feeAntarBank</b></li> <li>• <b>Saldo penerima akan selalu bertambah sesuai dengan nominal transfer.</b></li> <li>• Apabila proses transfer berhasil, maka panggil method <b>printBuktiTransfer(DigitalPayment penerima, long nominal)</b> yang ada pada class DigitalPayment.</li> </ul>

Class MobileWallet	
<b>Keterangan:</b> Salah satu bentuk dari alat pembayaran digital yaitu Mobile Wallet (e-Wallet).	
noHP	Atribut ini merupakan identitas untuk akun mobile wallet
feeTransferBank	Atribut ini akan diisi sesuai dengan fee masing-masing MobileWallet untuk melakukan transfer ke bank.
MobileWallet(String nama, long saldo, String noHp)	Menginisialisasi atribut nama, saldo, dan noHp sesuai dengan parameter.
transfer(DigitalPayment dp, long nominal)	<p>Method ini merupakan override dari method abstract yang ada pada class DigitalPayment. Berikut beberapa hal yang perlu kamu perhatikan:</p> <ul style="list-style-type: none"> <li>• Apabila saldo kurang dari 0, maka transfer gagal dan tampilkan pesan input tidak valid</li> <li>• Apabila saldo tidak mencukupi, maka transfer gagal dan tampilkan pesan kegagalan</li> <li>• Apabila object dp merupakan instance dari BNIImo atau BRIImo, maka <b>saldo pengirim akan dikurangi dengan nominal dan feeTransferBank</b></li> <li>• <b>Saldo penerima</b> akan selalu <b>bertambah sesuai dengan nominal transfer</b>.</li> <li>• Apabila proses transfer berhasil, maka panggil method <b>printBuktiTransfer(DigitalPayment penerima, long nominal)</b> yang ada pada class DigitalPayment.</li> </ul>

Class BNIImo	
<b>Keterangan:</b> Salah satu platform mobile banking (e-Banking) yaitu BNI mobile.	
BNIImo(String nama, long saldo, String noRekening)	Menginisialisasi atribut nama, saldo, dan noRekening sesuai dengan parameter.
transfer(DigitalPayment dp, long nominal)	<p>Override dari method yang ada pada class MobileBanking. Berikut beberapa hal yang perlu kamu perhatikan:</p> <ul style="list-style-type: none"> <li>• Apabila object dp merupakan instance dari BRIImo, maka atribut checkFee pada class MobileBanking akan bernilai true.</li> <li>• Proses transfer akan terjadi sesuai dengan aturan yang ada pada class MobileBanking.</li> </ul>

Class BRImo	
<b>Keterangan:</b> Salah satu platform mobile banking (e-Banking) yaitu BRI mobile.	
BRImo(String nama, long saldo, String noRekening)	Menginisialisasi atribut nama, saldo, dan noRekening sesuai dengan parameter.
transfer(DigitalPayment dp, long nominal)	<p>Override dari method yang ada pada class MobileBanking. Berikut beberapa hal yang perlu kamu perhatikan:</p> <ul style="list-style-type: none"> <li>• Apabila object dp merupakan instance dari BNIImo, maka atribut checkFee pada class MobileBanking akan bernilai true.</li> <li>• Proses transfer akan terjadi sesuai dengan aturan yang ada pada class MobileBanking</li> </ul>

Class Dana	
<b>Keterangan:</b> Salah satu platform mobile wallet (e-Wallet) yaitu DANA.	
danaFeeTransferBank	Atribut ini memiliki nilai default yaitu 1000 dan berfungsi sebagai biaya tambahan apabila terjadi proses transfer dari DANA ke rekening bank.
Dana(String nama, long saldo, String noHp)	Menginisialisasi atribut nama, saldo, dan noHp sesuai dengan parameter.
transfer(DigitalPayment dp, long nominal)	<p>Override dari method yang ada pada class MobileWallet. Berikut beberapa hal yang perlu kamu perhatikan:</p> <ul style="list-style-type: none"> <li>• Isi atribut feeTransferBank pada class MobileWallet sesuai dengan danaFeeTransferBank.</li> <li>• Apabila object dp merupakan instance dari Ovo, maka transfer gagal dan tampilkan pesan yaitu akun DANA tidak valid.</li> <li>• Proses transfer akan terjadi sesuai dengan aturan yang ada pada class MobileWallet.</li> </ul>

Class Ovo	
<b>Keterangan:</b> Salah satu platform mobile wallet (e-Wallet) yaitu OVO.	
ovoFeeTransferBank	Atribut ini memiliki nilai default yaitu 2000 dan berfungsi sebagai biaya tambahan apabila terjadi proses transfer dari OVO ke rekening bank.
Dana(String nama, long saldo, String noHp)	Menginisialisasi atribut nama, saldo, dan noHp sesuai dengan parameter.
transfer(DigitalPayment dp, long nominal)	<p>Override dari method yang ada pada class MobileWallet. Berikut beberapa hal yang perlu kamu perhatikan:</p> <ul style="list-style-type: none"> <li>• Isi atribut feeTransferBank pada class MobileWallet sesuai dengan ovoFeeTransferBank.</li> </ul>

	<ul style="list-style-type: none"> <li>• Apabila object dp merupakan instance dari Dana, maka transfer gagal dan tampilkan pesan yaitu akun OVO tidak valid.</li> <li>• Proses transfer akan terjadi sesuai dengan aturan yang ada pada class MobileWallet</li> </ul>
--	---

Selanjutnya, gunakan **class App yang sudah tersedia pada arsip soal** untuk menguji program kamu. Apabila algoritma yang kamu sudah tepat, maka masing-masing skenario akan menghasilkan output yang sesuai dengan gambar di bawah ini:

### Skenario 1 – 9

```

-----Skenario 1: Input Tidak Valid-----
Nominal yang Anda input tidak valid!
Nominal yang Anda input tidak valid!
-----Skenario 2: Saldo Tidak Mencukupi-----
Transfer gagal! Saldo Anda tidak mencukupi.
Transfer gagal! Saldo Anda tidak mencukupi.
-----Skenario 3: Transfer Sesama Bank-----
Transfer ke BRI Mobile atas nama Joyce sebesar Rp 200000 sukses
-----Skenario 4: Transfer Beda Bank-----
Transfer ke BNI Mobile atas nama Felix sebesar Rp 300000 sukses
-----Skenario 5: Bank Transfer ke e-Wallet-----
Transfer ke DANA atas nama Olive sebesar Rp 1000000 sukses
-----Skenario 6: OVO Transfer ke Bank-----
Transfer ke BNI Mobile atas nama Felix sebesar Rp 100000 sukses
-----Skenario 7: DANA transfer ke Bank-----
Transfer ke BRI Mobile atas nama Angel sebesar Rp 100000 sukses
-----Skenario 8: Dana transfer ke Dana-----
Transfer ke DANA atas nama Nola sebesar Rp 150000 sukses
-----Skenario 9: Dana transfer ke OVO-----
Akun Dana tidak valid!

```

### Skenario 10

```

-----Skenario 10: Saldo Akhir-----
=====Rincian Saldo=====
Nama      : Angel
No. Rekening: 250071190490
==> Saldo  : Rp 3594000
=====Rincian Saldo=====
Nama      : Felix
No. Rekening: 250071190490
==> Saldo  : Rp 2900000
=====Rincian Saldo=====
Nama      : Joyce
No. Rekening: 250071190490
==> Saldo  : Rp 1700000
=====Rincian Saldo=====
Nama      : Olive
No. HP      : 081234567891
==> Saldo  : Rp 850000
=====Rincian Saldo=====
Nama      : Adrian
No. HP      : 081578961543
==> Saldo  : Rp 48000
=====Rincian Saldo=====
Nama      : Nola
No. HP      : 081734566789
==> Saldo  : Rp 599000

```

### Komponen Penilaian:

- Program dibuat dengan menggunakan Maven dan struktur project sesuai. **(Poin 5)**
- Class yang dibuat memiliki method, constructor, serta atribut yang lengkap sesuai class diagram dan ketentuan pada soal. **(Poin 25)**
- Override method transfer pada class MobileWallet memiliki algoritma yang tepat dan sesuai dengan ketentuan pada soal. **(Poin 15)**
- Override method transfer pada class MobileRekening memiliki algoritma yang tepat dan sesuai dengan ketentuan pada soal. **(Poin 15)**
- Override method transfer pada class Dana, Ovo, BRIImo, dan BNIImo memiliki algoritma yang tepat dan sesuai dengan ketentuan pada soal. **(Poin 20 = 4 x 5 Poin)**
- Output skenario 1-9 tepat dan sesuai **(Poin 10)**
- Output skenario 10 tepat dan sesuai **(Poin 10)**