1. **DEFINISI MASALAH**
2. Buatlah implementasi sebuah Mesin ATM. Dalam Mesin ATM, password dapat dimasukkan oleh user. Selain itu, sistem memiliki pilihan menu melihat saldo, menarik uang, dan mentransfer ke rekening lain!
3. **SOURCE CODE**

Class ATM :

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121 | package Bab2;  import java.util.Scanner;  class ATM {  private int firstsaldo = 10000000; //saldo awal 10 juta  private String nama;  private int pasword;  int userinput1;  int userinput2;  int penarikan;  int nomtransfer;  String rek;  public int stop = 0;    String namaBank;  Scanner input = new Scanner(System.in);  Scanner input2 = new Scanner(System.in);    public ATM(){  String nm = "PTI";  int pswd = 1234;  pasword = pswd;  nama = nm;  }    public void cekPasword(){  int i = 0;  do{  System.out.print("Masukkan Password Anda : ");  userinput1 = input.nextInt();  if (i > 1){  System.out.println("Maaf, Transaksi Anda Dibatalkan");  System.exit(0);  }  i++;  }  while (userinput1 != pasword);  }  public void cekSaldo(){  System.out.println("Saldo anda sekarang adalah : " + "Rp " + firstsaldo + ",-");  }  int penarikan(){  int i = 0;  do{  System.out.print("Masukkan Nominal Penarikan : ");  penarikan = input.nextInt();  if (penarikan > firstsaldo){  System.out.println("Maaf, Saldo Anda Tidak Mencukupi");  System.out.println("Silahkan Masukkan Nominal yang Lebih Kecil");  }    if (penarikan < 0){  System.out.println("Maaf, Anda Tidak Memasukkan Nominal yang Benar");  System.out.println("Silahkan Masukkan Nominal yang Benar");  penarikan();  }    else {  System.out.println("Penarikan Berhasil, Sebanyak : " + "Rp " + penarikan + ",-");  System.out.println("Terima Kasih^^, Silahkan Ambil Uang Anda ...");  firstsaldo = firstsaldo - penarikan;  }  i++;  }  while (penarikan > firstsaldo);  return firstsaldo;  }  public void transfer(String norek, int money){  rek = norek;  nomtransfer = money;  int i = 0;  do{  System.out.print("Masukkan Nomor Rekening : ");  rek = input2.nextLine();  System.out.print("Masukkan Nominal Transfer : ");  nomtransfer = input.nextInt();    if (nomtransfer > firstsaldo){  System.out.println("Maaf, Saldo Anda Tidak Mencukupi");  System.out.println("Silahkan Masukkan Nominal yang Lebih Kecil");  }    else {  System.out.println("Transfer Berhasil, Sebesar : " + "Rp " + nomtransfer + ",-");    firstsaldo = firstsaldo - nomtransfer;  }  i++;  }  while (nomtransfer > firstsaldo);  }    public void transfer(String norek, int money, String bankname){  rek = norek;  nomtransfer = money;  namaBank = bankname;  int i = 0;  do{  System.out.print("Masukkan Nomor Rekening : ");  rek = input2.nextLine();  System.out.print("Masukan Nama Bank : ");  bankname = input.next();  System.out.print("Masukkan Nominal Transfer : ");  nomtransfer = input.nextInt();    if (nomtransfer > firstsaldo){  System.out.println("Maaf, Saldo Anda Tidak Mencukupi");  System.out.println("Silahkan Masukkan Nominal yang Lebih Kecil");  }    else {  System.out.println("Transfer Berhasil, Sebesar : " + "Rp" + nomtransfer + ",-");  firstsaldo = firstsaldo - nomtransfer;  }  i++;  }    while (nomtransfer > firstsaldo);  }  } |

Main Class ATM :

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  1819  20  21  22  23  24  25  26  27  28  29  30  3132  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71 | package Bab2;  import java.util.Scanner;  class MainATM{  public static void main (String[]args){  Scanner input = new Scanner(System.in);  int pilih;  int pilihBank;  int nominal = 0;  String ulang;  String nmBank = "";  String noRekening = "";    System.out.println("==================================================");  System.out.println("===== Selamat Datang di BANK PTI A =====");  System.out.println("== ATM ini Menggunakan Uang Pecahan Rp 50.000,- ==");  System.out.println("==================================================");  ATM PTI = new ATM();  PTI.cekPasword();  int i = 0;  do{  System.out.println("--- Pilihan Menu ---");  System.out.println("1. Informasi Saldo");  System.out.println("2. Penarikan");  System.out.println("3. Transfer");  System.out.println("4. Keluar");  System.out.print("Masukkan Pilihan Anda : ");  pilih = input.nextInt();  if (pilih == 1)  PTI.cekSaldo();  else if (pilih == 2)  PTI.penarikan();  else if (pilih == 3){  System.out.println("--- Pilih Bank ---");  System.out.println("1. Bank yang Sama");  System.out.println("2. Bank yang Berbeda");  System.out.print("Masukkan Pilihan Anda : ");  pilihBank = input.nextInt();  if (pilihBank == 1){  PTI.transfer(noRekening, nominal);  } if (pilihBank == 2){  PTI.transfer(noRekening, nominal, nmBank);  } else {  System.out.println("Maaf Inputan Tidak Benar");  }  }    else if (pilih == 4){  System.out.println("Terimakasih atas Kunjungan Anda");  break;  }  else{  System.out.println("Maaf Menu Tidak Tersedia");  }    System.out.print("Apakah Anda ingin Bertransaksi Lagi [y/t] : ");  ulang = input.next();  i++;  }    while (ulang.equals("y") || ulang.equals("Y"));  System.out.println("Created by : Albert Wirawan");  }  } |

1. **PEMBAHASAN**

Class ATM:

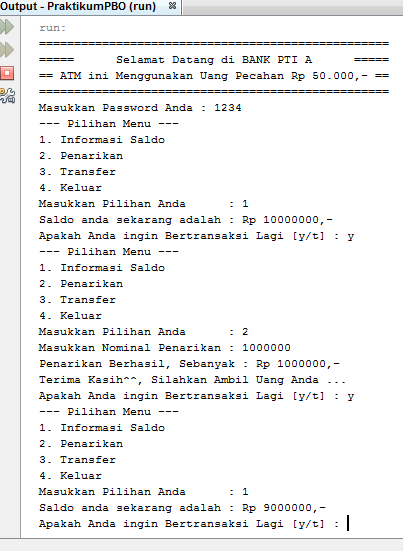
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127 | Letak Package berada di folder Bab2  Mengimport fungsi java utility Scanner  Class yang Bernama ATM  Variabel Saldo Awal bertipe integer dengan nilai awal Rp 10000000,- (10 juta), Bersifat private.  Variabel Nama yang bertipe String, Bersifat private.  Variable password yang bertipe integer, Bersifat private.  Variabel userinput1 yang bertipe integer.  Variabel userinput2 yang bertipe integer.  Variabel Penarikan yang bertipe integer.  Variabel nomtransfer yang bertipe integer.  Variabel rek yang bertipe String, untuk variabel nomor rekening.  Variabel Stop yang bertipe integer, dengan Nilai awal 0, bersifat public.  Variabel namaBank yang bertipe String.  Membuat Fungsi Scanner input yang baru.  Membuat Fungsi Scanner input2 yang baru.  Method void ATM.  Variabel nm yang memiliki Nama PTI, bertipe String.  Variabel pswd yang bertipe integer, dengan settingan paswordnya adalah 1234.  Menginisialisasi kalau password sama dengan pswd.  Menginisialisasi kalau nama sama dengan nm.  Tutup Kurawal untuk Method ATM.  Method untuk Mengecek Inputan Password yang dimasukkan user.  Variabel I bertipe data integer, dengan nilai awal sama dengan 0.  Proses Looping menggunakan do-while  Mencetak perintah Untuk Memasukkan Password user yang sudah disetting yaitu 1234.  Membuat Proses Input dari Variabel userinput1 untuk memasukkan Password.  Kondisi untuk menyeleksi apakah i lebih besar dari 1.  Mencetak yang ada di dalam tanda kutip dua.  Statement  Tutup kurawal untuk Seleksi If.  Statement  Tutup kurawal untuk Perulangan do.  Statement While  Tutup kurawal untuk method Cek Password.  Method untuk Mengecek Saldo.  Mencetak jumlah saldo awal.  Tutup kurawal untuk method cek saldo.  Method untuk Penarikan uang.  Variabel i yang bernilai awal sama dengan 0.  Kondisi Perulangan menggunakan do-while.  Mencetak yang ada di dalam tanda kutip dua. Untuk memasukkan jumlah penarikan yang mau diambil.  Input untuk penarikan.  Kondisi Seleksi if yang pertama.  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Tutup kurawal untuk seleksi if yang pertama.  Kondisi seleksi if yang kedua.  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Memanggil Method penarikan yang sudah dibuat sebelumnya.  Tutup kurawal untuk seleksi if yang kedua.  Kondisi else  Mencetak yang ada di dalam tanda kutip dua. Menampilkan status dan jumlah penarikan.  Mencetak yang ada di dalam tanda kutip dua.  Rumus untuk menghitung sisa saldo.  Tutup kurawal untuk else.  Statement  Tutup Kurawal untuk do.  Kondisi while  Fungsi return untuk mengembalikan nilai saldo awal.  Tutup Kurawal untuk Method Penarikan.  Method transfer yang overload karena memiliki 2 parameter  Inisialisasi  Inisialisasi  Variabel i yang bernilai awal sama dengan 0.  Perulangan do-while  Mencetak yang ada di dalam tanda kutip dua.  Menginput nomor rekening.  Mencetak yang ada di dalam tanda kutip dua.  Menginput nominal yang mau ditransfer.  Kondisi seleksi if else  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Tutup kurawal untuk kondisi if  Kondisi else  Mencetak yang ada di dalam tanda kutip dua. Mencetak status dan jumlahnya.  Rumus untuk menghitung sisa saldo.  Tutup kurawal untuk else  Statement  Tutup kurawal untuk do-while  Kondisi while  Tutup kurawal untuk method Transfer.  Method Transfer yang overload dengan 3 parameter untuk transfer ke bank lain  Inisialisasi  Inisialisasi  Inisialisasi  Variabel i dengan nilai awal 0  Perulangan dengan do  Mencetak yang ada di dalam tanda kutip dua.  Menginput nomor rekening  Mencetak yang ada di dalam tanda kutip dua.  Menginput nama bank  Mencetak yang ada di dalam tanda kutip dua.  Menginput nominal yang mau ditransfer  Kondisi seleksi if  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Tutup kurawal untuk if  Kondisi else  Mencetak yang ada di dalam tanda kutip dua.status dan jumlah nominal yang berhasil dikirim  Rumus Menghitung Sisa saldo  Tutup kurawal untuk else  Statement  Tutup kurawal untuk do  Kondisi while  Tutup kurawal untuk method transfer dengan 3 parameter  Tutup kurawal untuk kelas ATM. |

Main Class ATM:

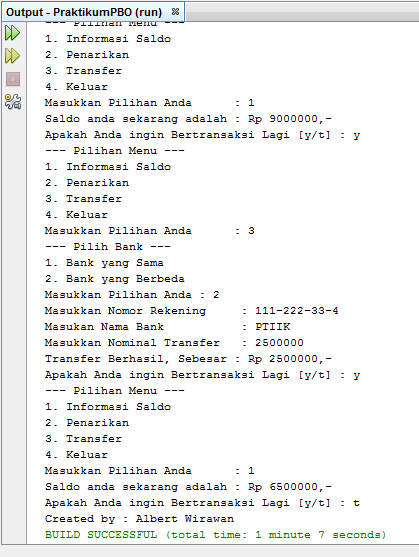
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70 | Letak Package  Mengimport fungsi Scanner  Class MainATM  Main method  Fungsi Scanner  Variabel pilih bertipe integer  Variabel pilihBank bertipe integer  Variabel Nominal bertipe integer dengan nilai awal sama dengan 0  Variabel ulang bertipe String  Variabel nmBank bertipe String  Variabel noRekening bertipe String  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Instansiasi Objek dari kelas ATM yang bernama PTI  Memanggil method cek password dari objek PTI  Variabel i bertipe integer, dengan nilai awal sama dengan 0  Prulangan dengan do-while  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Menginput pilihan menu  Seleksi if jika pilihannya sama dengan 1  Memanggil method cekSaldo  Seleksi jika pilihannya sama dengan 2  Memanggil method penarikan  Seleksi jika pilihannnya sama dengan 3  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Mencetak yang ada di dalam tanda kutip dua.  Menginput pilihan transfer  Seleksi jika pilihan menu transfernya sama dengan 1  Memanggil method transfer dengan 2 parameter  Seleksi jika pilihan menu transfernya sama dengan 2  Memanggil method transfer dengan 3 parameter  Kondisi else  Mencetak yang ada di dalam tanda kutip dua.  Tutup kurawal else  Tutup kurawal else if  Kondisi else if jika memilih menu awal 4  Mencetak yang ada di dalam tanda kutip dua.  Break  Tutup kurawal untuk else if  Kondisi else  Mencetak yang ada di dalam tanda kutip dua.  Tutup kurawal untuk kondisi else  Mencetak yang ada di dalam tanda kutip dua.  Menginput jika mau melakukan transaksi lagi atau tidak  Statement  Tutup kurawal do-while  Kondisi while  Mencetak yang ada di dalam tanda kutip dua.  Tutup kurawal main method’  Tutup kurawal mainATM |

1. **SCREENSHOT PROGRAM**

**Informasi Saldo dan Penarikan Uang Tunai.**

****

**Transfer Uang Ke Bank dan Cek Saldo.**

****

1. **PRAKTIKUM**

A. Constructor

1. Lakukan percobaan constructor diatas dan benahi jika menemukan kesalahan!

Jawab:

Class:

|  |  |
| --- | --- |
| 12  13  14  15  16 | public student(){ // S nya harusnya Besar  name = "";  address = "";  age = 0;  } |

2. Tambahkan constructor pada class Student dengan parameter yang mempunyai parameter masing masing nilai dari mata pelajaran yang ada! Kemudian buat contoh objeknya pada main Class!

Jawab:

3. Tambahkan method dengan nilai balikan berupa boolean pada class student bernama statusAkhir untuk menentukan apakah siswa tersebut remidi atau tidak. Ketentuannya adalah jika nilai lebih dari atau sama dengan 61 adalah lolos sedangkan nilai kurang dari atau sama dengan 60 adakah remidi. Nilai yang dicari adalah nilai rata-rata untuk semua mapel. Kemudian nilai pada method statusAkhir tampilkan pada method displayMessage!

Jawab:

4. Bagaimana cara memasukkan jumlah siswa sesuai dengan keinginan user? Tuliskan kodenya dengan inputan user yang interaktif! (key : menggunakan array)

Jawab:

5. Bagaimana cara menghitung banyaknya objek yang kita buat dari sebuah menginstance objek dari mein class? Tuliskan kodenya kemudian tampilkan informasinya dengan memanggil method jumlahObjek() bertipe void!

Jawab:

B. Instance Method

1. Lakukan percobaan Instance Method diatas dan benahi jika menemukan kesalahan!

Jawab: Tidak ada Kesalahan.

2. Tambahkan method untuk operator <, <=, >= !

Jawab:

3. Ubah method sederhana pada baris 25 – 30 yang awalnya adalah menggunakan while menjadi for!

Jawab:

4. Tambahkan method untuk operasi -, \*, / !

Jawab:

Class:

|  |  |
| --- | --- |
| 1  2  3 |  |

Main Class:

|  |  |
| --- | --- |
| 1  2  3 |  |

1. **KESIMPULAN**
2. jelaskan apa yang disebut dengan konstruktor!

Jawab:

Dalam pemrograman berorientasi objek, kita akan mengenal yang namanya konstruktor.

Konsktruktor adalah method yang pertama kali dijalankan pada saat sebuah objek pertama kali diciptakan.

Perbedaan method konstruktor dengan method lainnya pada java adalah nama method konstruktor harus sama dengan nama kelasnya.

Konstruktor biasanya digunakan untuk menginisialisasi atau memberikan nilai  (melalui parameter) kepada sebuah objek pada saat objek pertama kali diciptakan.

Jika dalam sebuah class tidak terdapat constructor maka secara implisit  Java akan menciptakan **default constructor**. Default constructor adalah constructor tanpa parameter yang akan menginisialisasi variable instance sesuai dengan type datanya. variable numeric akan di set nilainya menjadi 0, string diset nilainya menjadi character(“ ”), reference variable akan diset menjadi **NULL.**

1. jelaskan perbedaan konstruktor dan method pada umumnya!

Jawab:

Konstruktor itu mirip dengan method, perbedaannya:

* tidak memiliki nilai(void, String, int, dll)
* Namanya sama persis dengan nama kelas.

Contructor digunakan untuk inisialisasi sesuatu misalnya variabel,  
Method digunakan untuk menjalankan fungsi fungsi.

1. tuliskan cara mendeklarasikan konstruktor!

Jawab:

Constructor diakses dengan menggunakan keyword new. Contoh:

public class Bike publicBike() {  
}  
Public static void main (Strng args[]) {  
Bike obj=new Bike();  
}  
}

4. jelaskan apa yang disebut dengan instance method!

Jawab:

Instance Method adalah fungsi yg beroperasi pada objek untuk memanipulasi state objek lewat pesan yg diterima dari objek lain. Untuk memanggil method ini harus membuat objek dengan instance class.

Secara singkat instance method adalah methodnya object yaitu method yang hanya bisa dibaca setelah class tersebut diturunkan ke object. Instance method adalah method yang ditakrifkan untuk sesuatu instance (objek suatu kelas) Instance Method bisa dikatakan juga sebagai fungsi yg beroperasi pada objek untuk memanipulasi state objek lewat pesan yg diterima dari obek lain

Bentuk Umum:

**void namaMethodInstance(){**

Masih ada satu lagi, yaitu Method Static, Method Static adalah method yang dapat dipakai tanpa harus menginisialisasi suatu class (maksudnya tanpa menggunakan variabel terlebih dahulu). Ciri method ini adalah dengan menggunakan kata static di depan kata void. Method ini dapat dipanggil dengan membuat objek tanpa Instance Class.  
Bentuk Umum:

**static void namaMethodStatic(){**

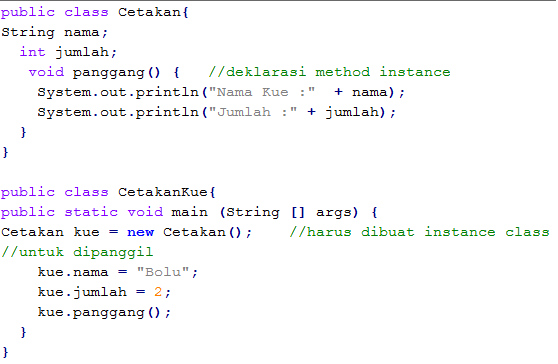
5. bagaimana cara memanggil sebuah instance method!

Jawab:

Untuk memanggil sebuah instance method, kita dapat menuliskan :

**namaObjek.namaMethod();**

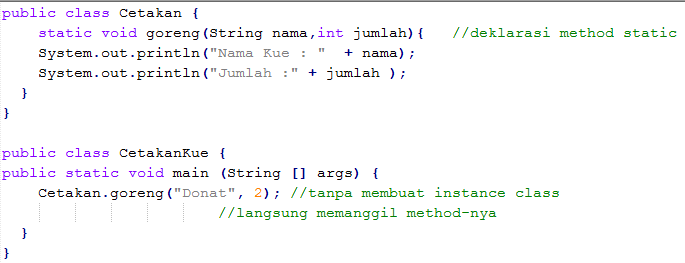
Contoh:



Untuk memanggil sebuah method static, kita dapat menuliskan :

**Namakelas.namaMethodStatic();**

Contoh:

****