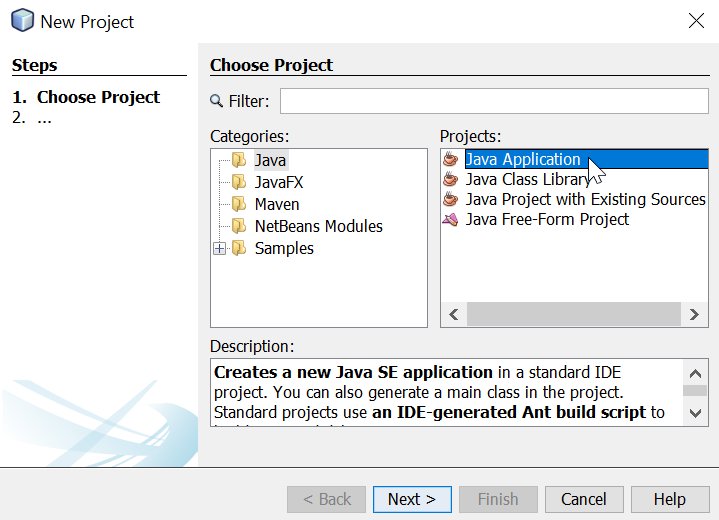
Lembar Praktikum 01

NIM : K3520058  
Nama Mahasiswa : Nur Isnaini Hanifah

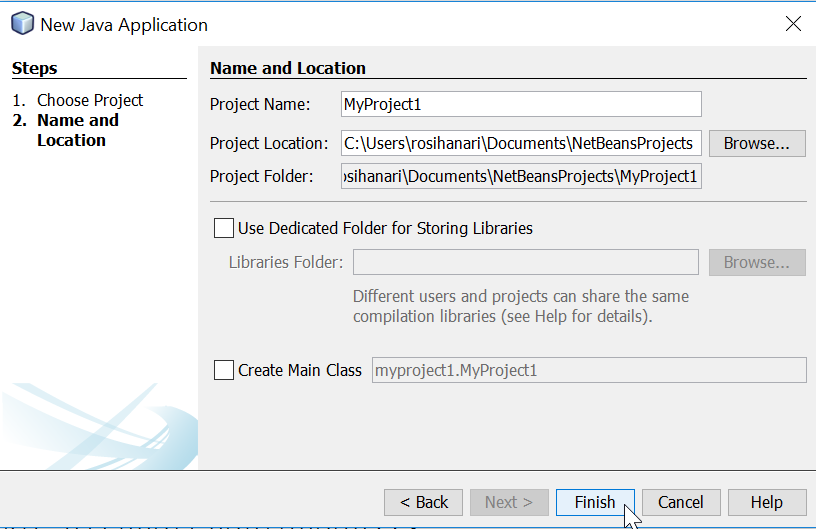
* Mata Pelajaran : Pemrograman Berorientasi Obyek
* Pokok Bahasan : Pendahuluan PBO
* Tujuan Pembelajaran :
  + Melalui serangkaian langkah praktikum dengan Netbeans, peserta didik mampu membuat package dengan benar
  + Melalui serangkaian langkah praktikum dengan Netbeans, peserta didik mampu membuat class dan main-class Java dengan benar
  + Melalui serangkaian langkah praktikum dengan Netbeans, peserta didik mampu mendefinisikan atribut dari sebuah class dengan benar
  + Melalui serangkaian langkah praktikum dengan Netbeans, peserta didik mampu membuat method sederhana di dalam sebuah class dengan benar
  + Melalui serangkaian langkah praktikum dengan Netbeans, peserta didik mampu membuat obyek dengan benar dari class yang telah didefinisikan
  + Melalui serangkaian langkah praktikum dengan Netbeans, peserta didik mampu menjalankan project Java dengan benar

# Langkah Membuat Project Java Baru di Netbeans

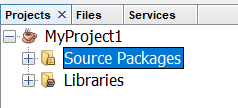
1. Buka Netbeans
2. Klik File - New Project
3. Pilih Categories -> Java, Projects -> Java Application



1. Klik Next
2. Isikan ‘MyProject1’ pada Project Name
3. Tentukan lokasi penyimpanan project
4. Hilangkan tanda centang pada ‘Create Main Class’



1. Klik Finish
2. Selanjutnya akan muncul struktur project di bagian sidebar sebelah kiri



**Kegiatan:**

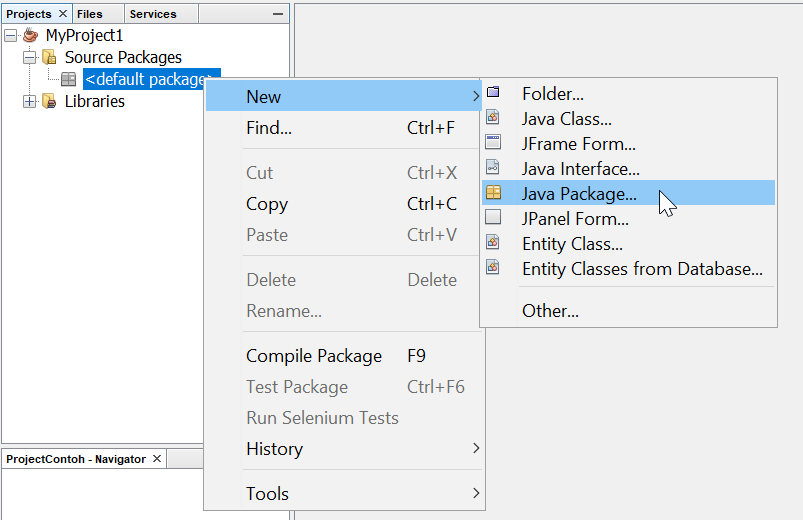
1. Lakukan langkah di atas!
2. Kemudian amati struktur direktori dari project yang sudah dibuat di atas, melalui Windows Explorer! Apa kesimpulan Anda terhadap struktur direktorinya?

Setelah membuat file projek, akan muncul 2 folder utama, yaitu Source Packages dan Libraries. Folder Packages merupakan default package yang akan berisi file-file utama dalam memprogram. Sedangkan folder Libraries berisi file library yang digunakan.

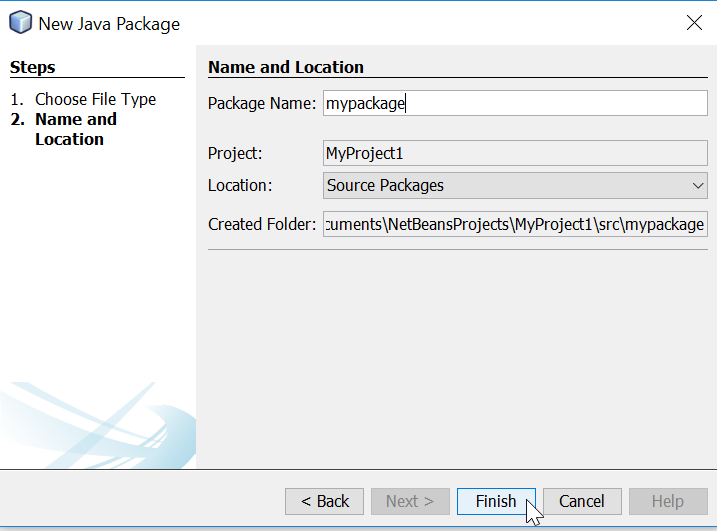
# Langkah Membuat Package

Sebuah package di dalam project, sama seperti folder. Di dalam sebuah package dapat diisi oleh beberapa class. Serta di dalam sebuah project bisa terdiri dari beberapa package.

1. Klik Source Packages
2. Klik kanan pada Default Package – New – Java Package



1. Beri nama package ‘mypackage’ pada ‘Package Name’



1. Klik Finish

**Kegiatan:**

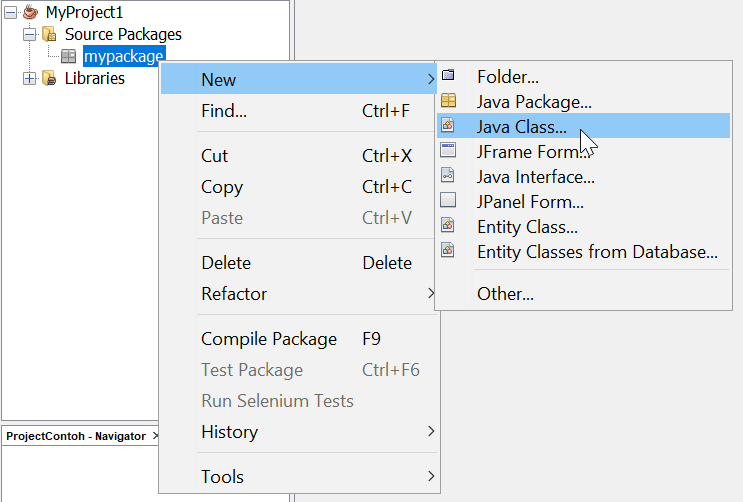
1. Lakukan langkah untuk membuat package di atas
2. Amati kembali efek pembuatan package tersebut di dalam direktori projectnya melalui Windows Explorer! Apa yang terjadi terhadap struktur direktori project setelah pembuatan package?

Tidak ada perubahan yang terlalu terlihat, hanya perubahan pada folder Default Project berubah nama menjadi mypackage.

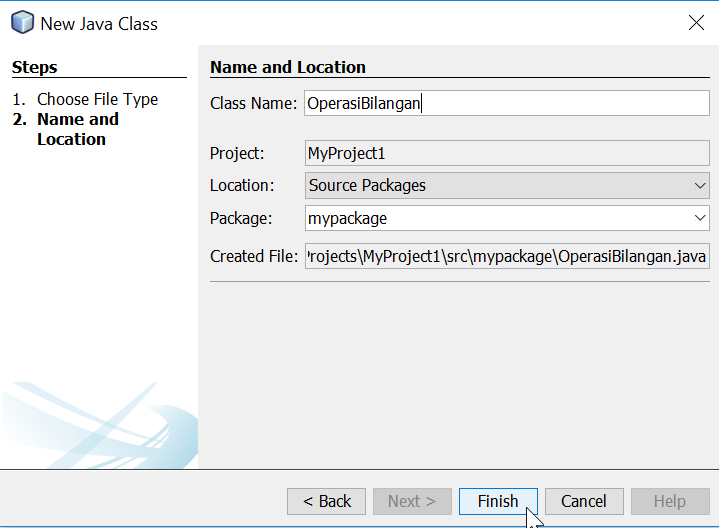
# Langkah Membuat Class

Setelah package dibuat, barulah bisa dibuat class-class di dalamnya

1. Klik kanan pada package yang sudah dibuat
2. Klik New – Java Class



1. Beri nama ‘OperasiBilangan’ pada Class Name



1. Klik Finish

**Kegiatan:**

1. Lakukan langkah membuat class di atas
2. Amati perubahan tampilan di Netbeans setelah sebuah class dibuat! Tampilan apa yang berubah?

Muncul sebuah tampilan yang menjadi tempat untuk menuliskan coding atau program yang akan dibuat

1. Amati perubahan struktur direktori di project melalui Windows Explorer! Perubahan apakah yang terjadi setelah sebuah class dibuat?

File mypackage berubah menjadi sebuah folder yang berisi OperasiBilangan.java dan pada bagian tab navigator muncul sebuah navigator OperasiBilangan

1. Menurut Anda, kira-kira apa kegunaan dari baris perintah

package mypackage;

yang terletak di atas class ‘OperasiBilangan’?

Perintah tersebut berfungsi untuk memerintahkan memanggil yang ada pada package Bernama mypackage. Hal itu juga dapat berfungsi untuk memberitahukan bahwa program yang dikelola berada di dalam package Bernama mypackage

# Menambahkan Atribut dan Method di dalam Class

Setelah sebuah class dibuat, atribut dan method dapat ditambahkan ke dalamnya.

**Kegiatan:**

1. Tambahkan kode berikut ini di dalam class ‘OperasiBilangan’

public class OperasiBilangan {

// atribut

public int bilPertama;

public int bilKedua;

// methods

public void hitungPenjumlahan(){

int hasil = bilPertama + bilKedua;

System.out.println("Hasil penjumlahannya: " + hasil);

}

public void hitungPengurangan(){

int hasil = bilPertama - bilKedua;

System.out.println("Hasil pengurangannya: " + hasil);

}

}

1. Amati cara penulisan atribut dan method di atas. Apa perbedaan mencolok cara pendefinisian keduanya dalam hal penulisan nama dan strukturnya ?

Atribut merupakan ciri atau mendefinisikan dari objek yang dibuat. Sedangkan method merupakan kegiatan atau kemampuan yang dilakukan pada objek yang telah dibuat atau didefinisikan pada sebelumnya.

# Membuat Obyek (Instansiasi)

Setelah sebuah class selesai didefinisikan, selanjutnya dapat dilakukan proses instansiasi yaitu proses pembuatan obyek dari class tersebut.

**Kegiatan:**

1. Buatlah class baru ‘MyClass’ di dalam package ‘mypackage’ pada project ‘MyProject1’
2. Tuliskan kode program berikut ini

public class MyClass {

public static void main(String[] args) {

OperasiBilangan op1 = new OperasiBilangan();

op1.bilPertama = 10;

op1.bilKedua = 20;

op1.hitungPenjumlahan();

op1.hitungPengurangan();

}

}

Keterangan:  
Sebuah class yang di dalamnya terdapat method **main()** dinamakan **main class**.

1. Jalankan project ‘MyProject1’ dengan mengklik Run – Run Project
2. Amati tampilan outputnya. Apa maksud dari program Java yang baru saja dibuat?

Terdapat output dari program yang dibuat berupa

Hasil penjumlahannya: 30

Hasil pengurangannya: -10

1. Dalam class ‘MyClass’, buatlah obyek baru dengan nama ‘op2’. Lalu gunakan obyek ‘op2’ ini untuk mencari hasil penjumlahan serta pengurangan dari 25 dan 55.
2. Buatlah class baru dengan nama ‘MyNewClass’ di dalam package ‘mypackage’ dengan isi class sebagai berikut.

public class MyNewClass {

public static void main(String[] args) {

OperasiBilangan op1 = new OperasiBilangan();

op1.bilPertama = 50;

op1.bilKedua = 60;

op1.hitungPenjumlahan();

op1.hitungPengurangan();

}

}

1. Jalankan project ‘MyProject1’ dengan mengklik Run – Run Project! Amati apa yang terjadi?

Hasil dari program yang dibuat sama seperti sebelumnya, hanya saja bertambah pada hasil program op2. Sedangkan class MyNewClass tidak dijalankan. Tampilan hasil seperti di bawah ini

Hasil penjumlahannya: 30

Hasil pengurangannya: -10

Hasil penjumlahannya: 80

Hasil pengurangannya: -30

1. Bagaimana supaya perintah yang ada di dalam main() method MyNewClass ini dapat dijalankan?

Supaya method MyNewClass dapat dijalankan, dapat mengubah pilihan main class yang akan dijalankan menjadi main.MyNewClass

1. Apa kesimpulan Anda tentang main class dalam sebuah project?

Pemilihan main class yang akan berpengaruh pada class yang akan dijalankan. Jika pemilihan main class adalah MyClass maka program pada MyClass lah yang akan dijalankan bukan MyNewClass. Hal ini juga berlaku untuk sebaliknya.

1. Tukarkan posisi urutan penulisan atribut dan method yang ada di class ‘OperasiBilangan’. Kemudian jalankan project. Apakah ada efeknya terhadap output?

Tidak ada perubahan atau efek pada output yang dihasilkan apabila penulisan atribut dan method ditukar

# Project 1

Sampai tahap ini, Anda sudah bisa membuat project baru, membuat package, class, kemudian mendefinisikan atribut, method, serta membuat obyek, dan juga memahami kegunaan main-class. Selanjutnya kerjakan project berikut ini

1. Dalam class ‘OperasiBilangan’, tambahkan method baru hitungPerkalian() untuk menghitung hasil perkalian kedua bilangan, serta method hitungPembagian() untuk menghitung hasil pembagiannya.

Petunjuk: di method hitungPembagian() gunakan tipe data double

1. Modifikasilah project ‘MyProject’ sedemikian hingga bisa menampilkan hasil penjumlahan, pengurangan, perkalian, dan pembagian dari dua bilangan yaitu 100 dan 150.

# Project 2

1. Buatlah package ‘bangundatar’ di dalam project ‘MyProject1’
2. Buatlah class ‘PersegiPanjang’ di dalam package ‘bangundatar’.
3. Tambahkan atribut ‘panjang’ dan ‘lebar’ di dalam class ‘PersegiPanjang’, keduanya bertipe data integer.
4. Tambahkan method hitungLuas() dan hitungKeliling() yang masing-masing digunakan untuk menghitung luas dan keliling persegi panjang
5. Buatlah main class di dalam package ‘bangundatar’ dengan nama ‘BangunDatarMain’
6. Melalui main class ‘BangunDatarMain’, gunakan class ‘PersegiPanjang’ yang telah dibuat untuk menghitung luas dan keliling persegi panjang jika diketahui panjangnya 25 dan lebarnya 38 satuan panjang.
7. Buatlah class ‘Persegi’, dan ‘Lingkaran’ di dalam package ‘bangundatar’. Rancang dan definisikan atribut yang sesuai untuk masing-masing class, serta tuliskan method hitungLuas() dan hitungKeliling() nya.
8. Selanjutnya gunakan class ‘Persegi’ untuk menghitung luas dan keliling dari beberapa persegi berikut ini:
   1. Sebuah persegi A jika diketahui sisi perseginya 10 satuan panjang
   2. Sebuah persegi B, jika diketahui sisi perseginya 15 satuan panjang
9. Selanjutnya gunakan class ‘Lingkaran’ untuk menghitung luas dan keliling dari beberapa lingkaran berikut ini:
   1. Sebuah lingkaran X jika diketahui jari-jarinya 25 satuan panjang
   2. Sebuah lingkaran Z, jika diketahui jari-jarinya 37 satuan panjang

# Project 3 (Pengayaan)

1. Buat package dengan nama ‘bangunruang’
2. Buat 3 class di dalam package tersebut, masing-masing bernama ‘Bola’, ‘Tabung’, ‘Kerucut’.
3. Untuk tiap-tiap class, definisikan atributnya
4. Untuk tiap-tiap class, buatlah method hitungVol() untuk menghitung volume, serta hitungLuasSelimut() untuk menghitung luas selimut benda ruang

# Pengumpulan Project

* Buatlah repository di Github untuk mempush project hasil project 1, 2, dan 3
* Copy pastekan URL Reponya di bawah ini

URL Repo: ……………………………………………………………