



### MEKANISME PRAKTIKUM

- 1) Buat Sebuah **Project Java Baru** pada IntelliJ IDEA dengan nama Project:  
**"PertemuanX\_NPM AKHIR"**  
Ganti "X" menjadi Pertemuan yang sedang berlangsung.
- 2) Pada saat Praktikum, Jawabanlah Soal Pertanyaan yang memiliki Label **WAJIB** terlebih dahulu Pada Lembar "**Laporan Praktikum**".
- 3) Segala Bentuk **Soal yang memiliki Jawaban** berupa **Kode Program**, maka kode program tersebut harus disimpan pada **File java Project** yang telah dibuat.
- 4) Setiap **File Java** yang dibuat harus mencantumkan Pertanyaan pada bagian atas (baris pertama)
- 5) Simpan **File Laporan Praktikum** yang berupa **DOCX** menjadi **FILE PDF** kemudian ubah nama file PDF menjadi:  
**"PertemuanX\_NPM AKHIR.pdf"**
- 6) Upload File **Laporan Praktikum [PDF]** dan pada form yang sudah disediakan.

### TUGAS PRAKTIKUM

1. Apa yang dimaksud dengan **Data Collection** dan **Encapsulation** (Enkapsulasi) pada java? Jelaskan! Serta sebutkan macam-macam **Data Collection** yang kamu ketahui! [Wajib]
2. Apa perbedaan antara method **Getter** dan **Setter**? serta sebutkan dan jelaskan **Accesss Modifier** apa saja yang terdapat dalam konsep Enkapsulasi! [Wajib]
3. Buatlah sebuah class **Pegawai** yang memiliki atribut nama (gunakan method **getter & setter**). Kemudian, buatlah objek-objek berikut dari class tersebut. Pada kolom "...." isi dengan **nama kalian** sendiri!

NAMA
Rimuru
Ainz
Saitama
Gon
Asta
“....”



**SOAL PRAKTIKUM**  
**PEMROGRAMAN BERORIENTASI OBJEK**  
**PERIODE X**  
Laboratorium Rekayasa Perangkat Lunak, ITATS

---

Setelahnya, buatlah sebuah **ArrayList** untuk menampung objek-objek pegawai tersebut, kemudian cetak hasilnya dengan menggunakan for loop/foreach loop [Wajib]!

4. Merujuk pada soal sebelumnya, buatlah sebuah program Java untuk menambahkan **Objek Pegawai** dari class **Pegawai** yang telah dibuat tadi ke dalam class **Departemen**!

Note : Buatlah Class Departemen yang memiliki method untuk menambahkan daftar pegawai.

5. Buatlah sebuah program Java yang mengimplementasikan relasi **Aggregation** (agregasi) antar kelas. Pilihlah studi kasus yang relevan dimana satu kelas "memiliki" kelas lain, tetapi kelas yang dimiliki dapat berdiri secara independen tanpa kelas yang memiliki. (Berikan studi kasus selain yang terdapat di modul) !