

PRAKTIKUM PBO

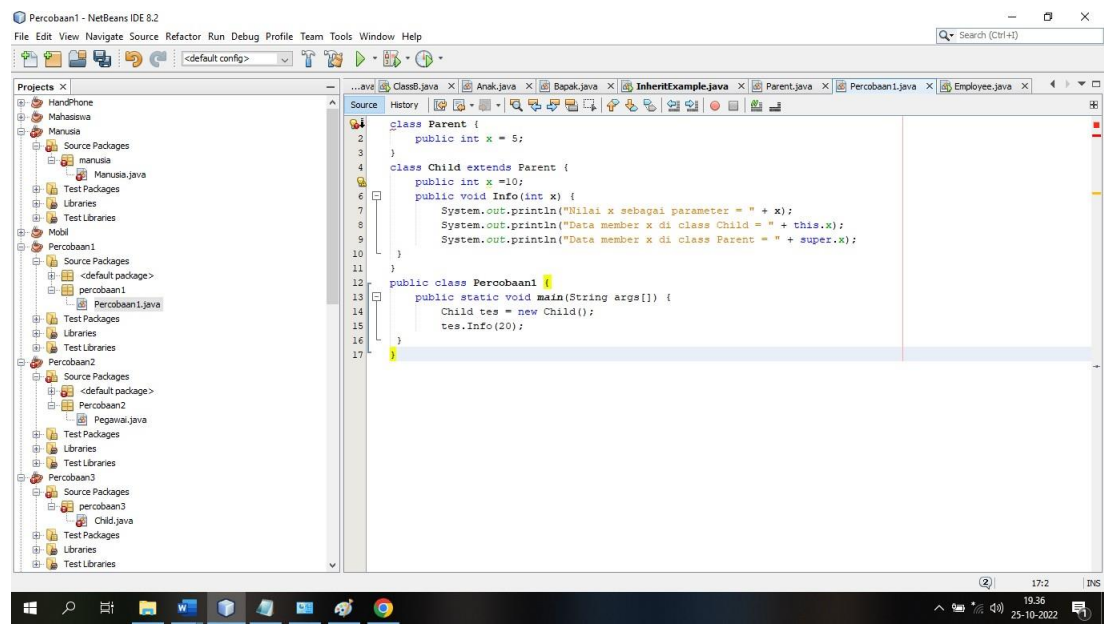
Nam : Moh Azral Fathurrazaq
Nim : 20210040135
Kelas : TI21E
Tugas : Pemrograman Berorientasi Objek

- 1. Kode Program upload ke github dengan nama repository praktikum-inheritance
- 2. Berikan analisa setiap percobaan dalam bentuk File teks pdf dan upload juga ke github praktikum-inheritance

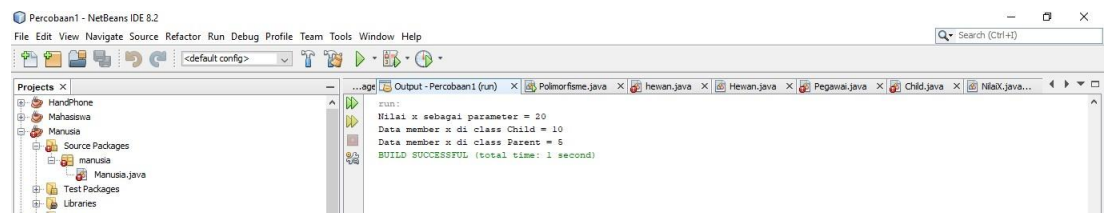
JAWABAN

1. PERCOBAAN 1

➤ INPUT PERCOBAAN 1



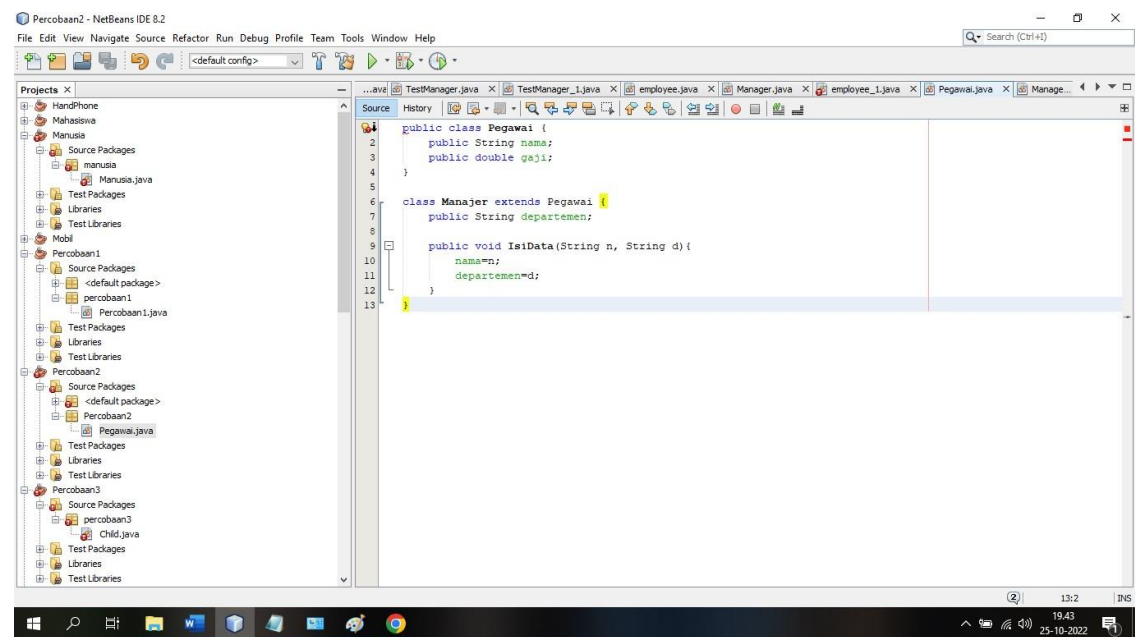
➤ OUTPUT PERCOBAAN 1



. Pada percobaan pertama ini class parent sebagai induk class yang memiliki atribut integer = 5 , Child sebagai sub class dan didalam class child terdapat sebuah nilai parameter 20 karena ditentukan dari tes.info. dan ada data member dari class parent bernilai 5 kenapa nilainya 5 karena “super” mengambil nilai integer dari class parent.

2. PERCOBAAN 2

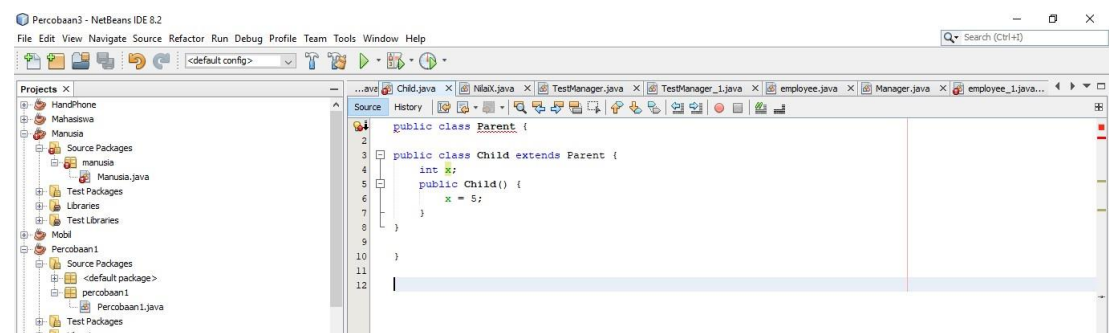
Percobaan berikut ini menunjukkan penggunaan kontrol akses terhadap atribut parent class. Mengapa terjadi error, dan bagaimana solusinya?



Solusi yang saya lakukan yaitu menghapus “public” sebelum kalimat class yang awalnya “public class Pegawai” menjadi class Pegawai begitupun dengan class Manager dan saya juga menambahkan String nama; agar nama=n; di public void isiData tidak terjadi error.

3. PERCOBAAN 3

Percobaan berikut ini menunjukkan penggunaan konstruktor yang tidak diwariskan. Mengapa terjadi error, dan bagaimana solusinya?

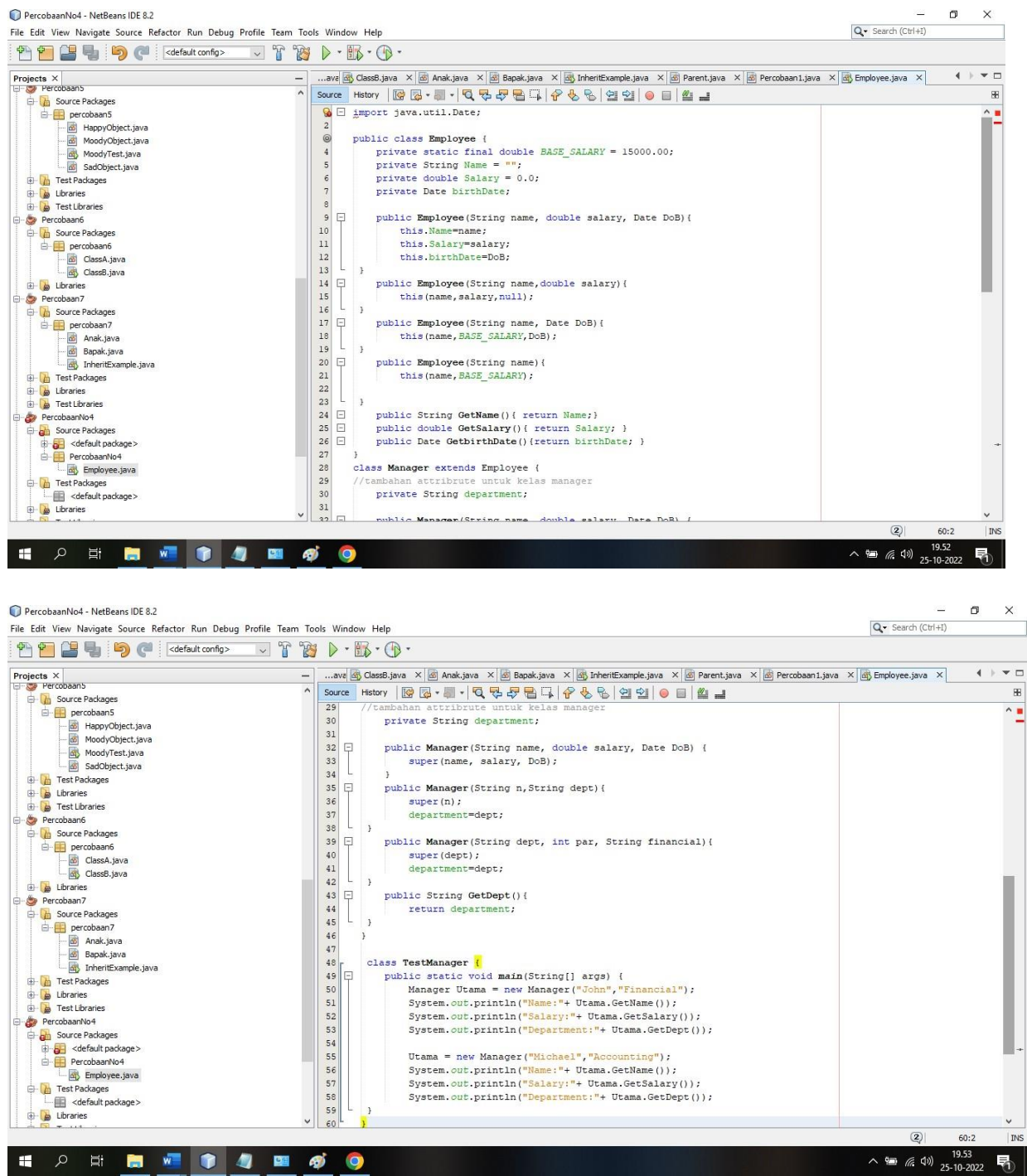


Solusinya saya menambahkan class utama yaitu percobaan3, kenapa costruktur terjadi error karena konstruktor tersebut berada di subclass.

4. PERCOBAAN 4

Percobaan berikut ini menunjukkan penggunaan kelas Employee dan subkelas Manager yang merupakan turunannya. Kelas TestManager digunakan untuk menguji

➤ INPUT PERCOBAAN 4



➤ OUTPUT PERCOBAAN 4

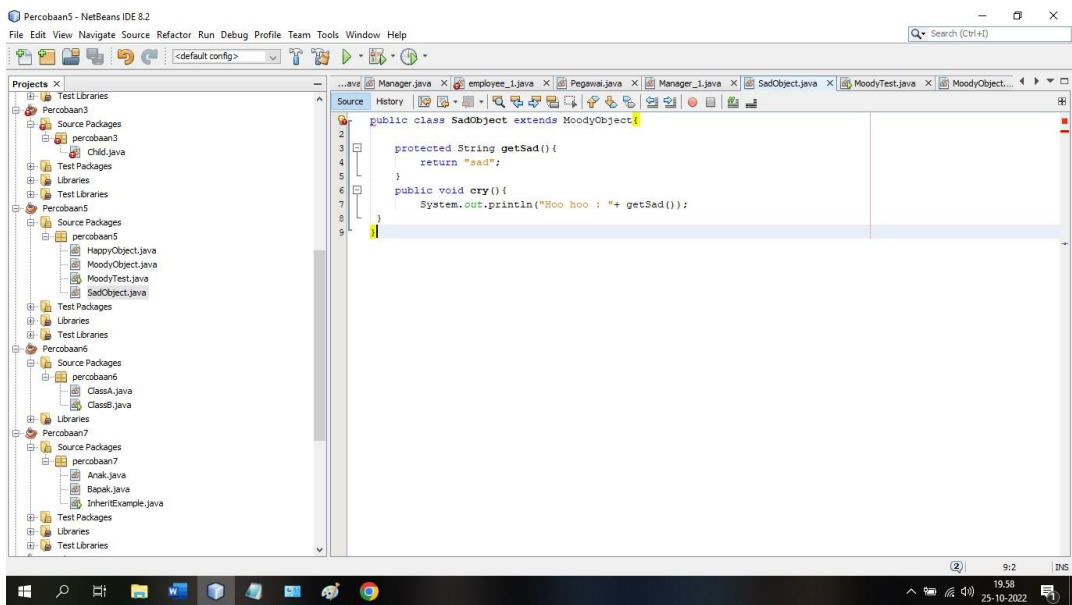


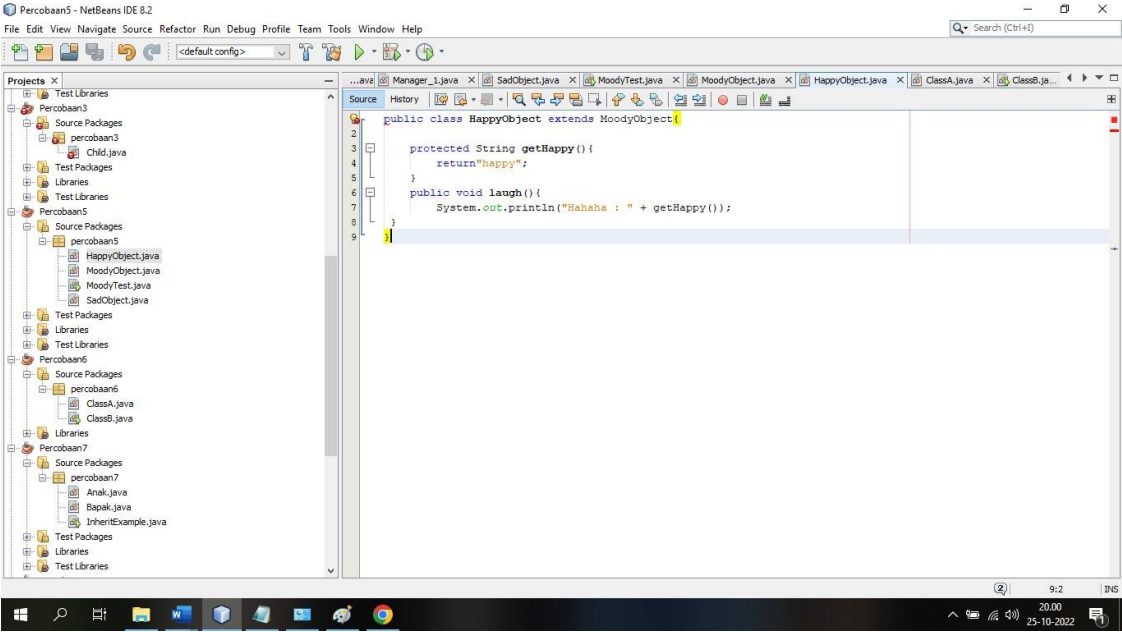
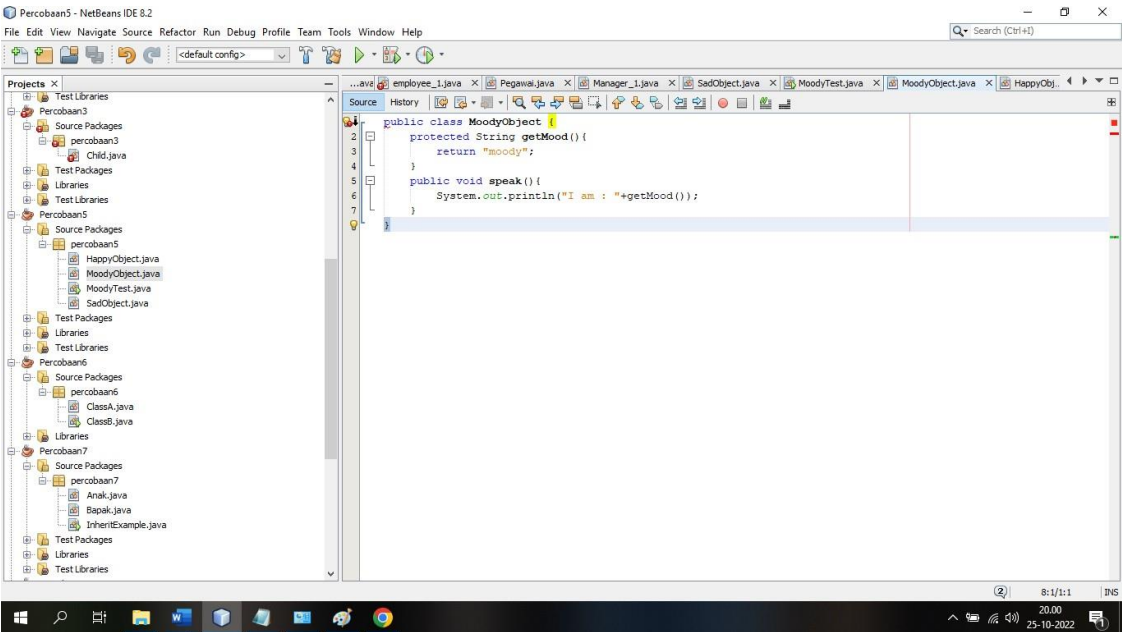
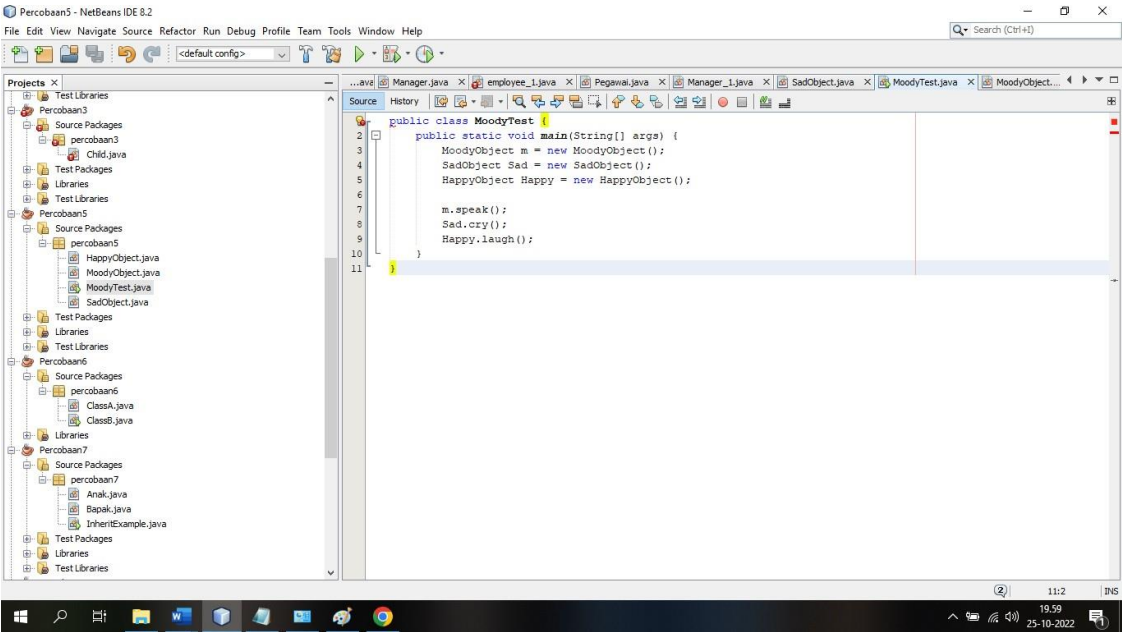
5. PERCOBAAN 5

Percobaan berikut ini menunjukkan penggunaan kelas `MoodyObject` dengan subkelas `HappyObject` dan `SadObject`. Kelas `MoodyTest` digunakan untuk menguji kelas dan subkelas.

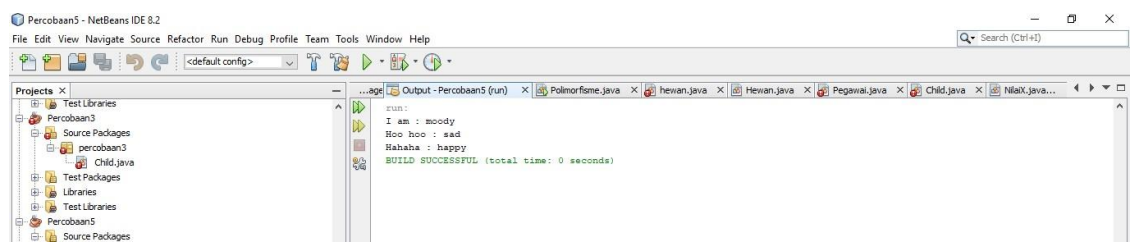
- `SadObject` berisi : `sad`, method untuk menampilkan pesan, tipe `public`
- `HappyObject` berisi : `laugh`, method untuk menampilkan pesan, tipe `public`
- `MoodyObject` berisi :
 1. `getMood`, memberi nilai mood sekarang, tipe `public`, return type `string`
 2. `Speak`, menampilkan mood, tipe `public`

➤ INPUT PERCOBAAN 5





➤ OUTPUT PERCOBAAN 5

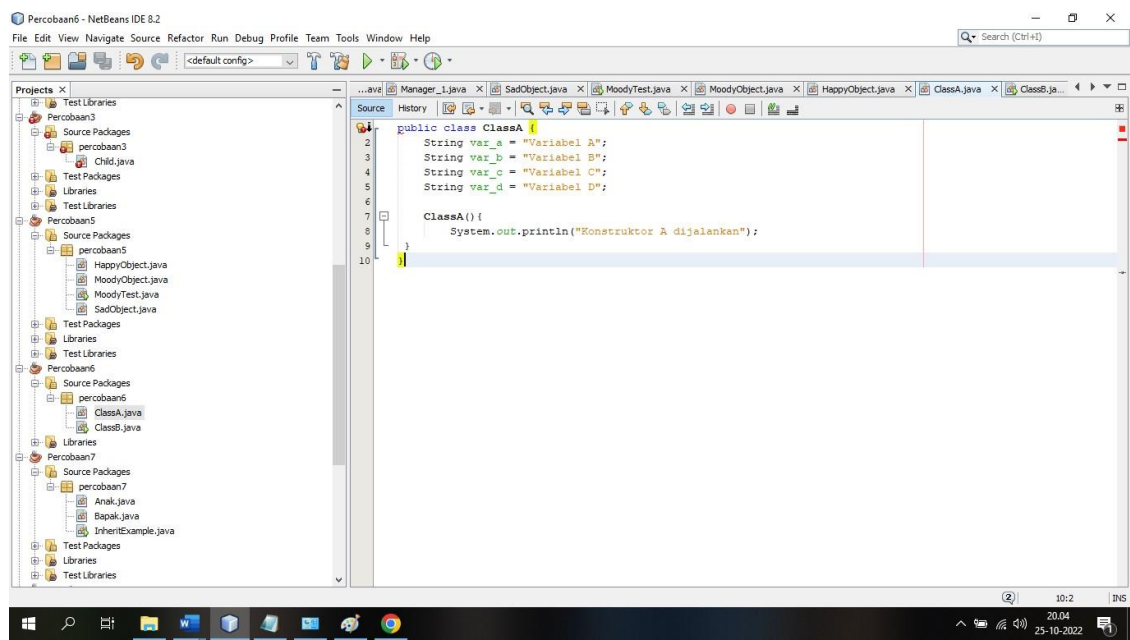


saya memperbaikinya hanya dengan menghapus public di tiap subclass.

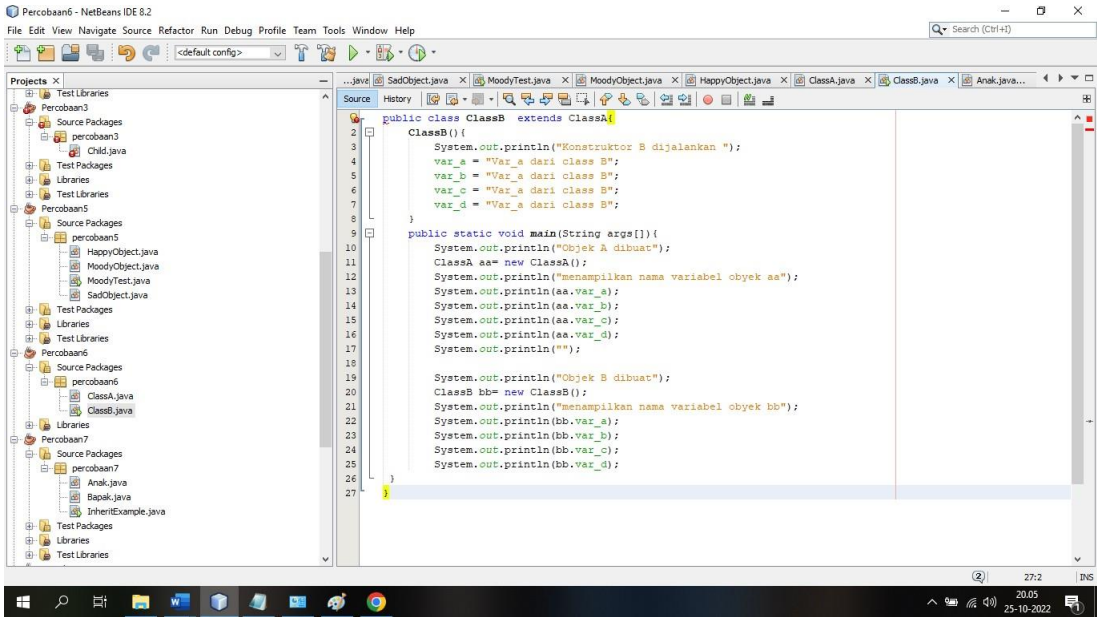
6. PERCOBAAN 6

Percobaan berikut ini menunjukkan penggunaan kelas A dan dengan subkelas B. Simpan kedua kelas ini dalam 2 file yang berbeda (A.java dan B.java) dan dalam satu package. Perhatikan proses pemanggilan konstruktor dan pemanggilan variabel.

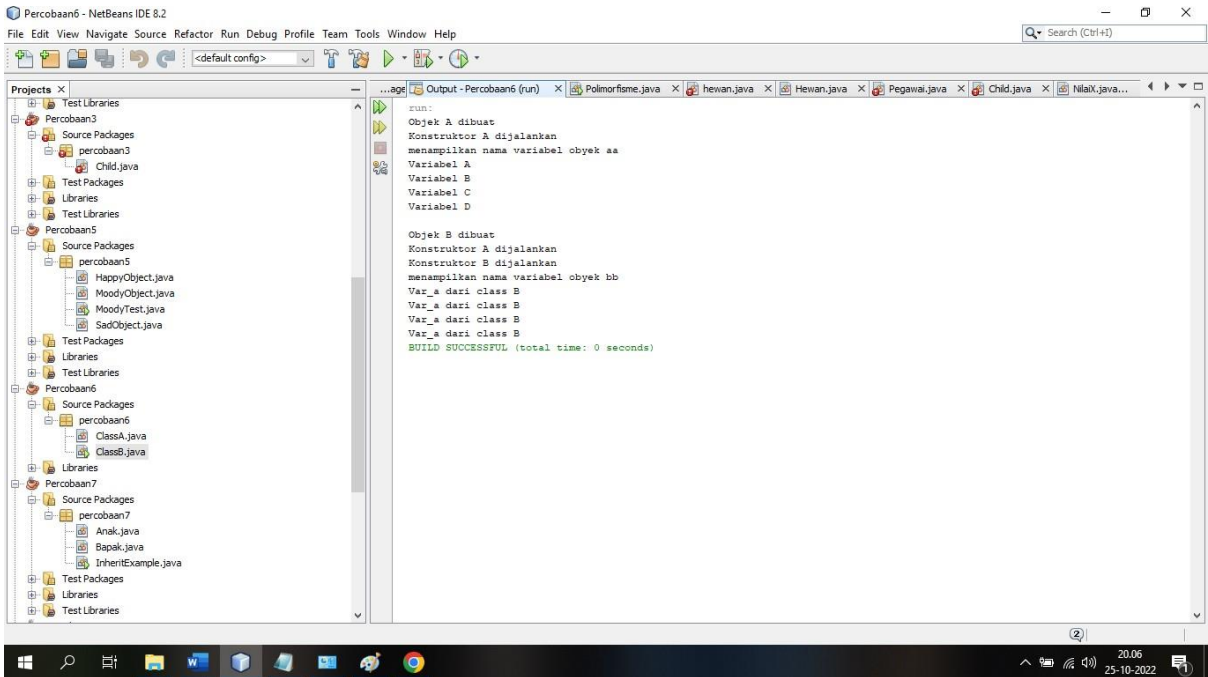
➤ INPUT CLASS A



➤ INPUT CLASS B



➤ OUTPUT PERCOBAAN 6

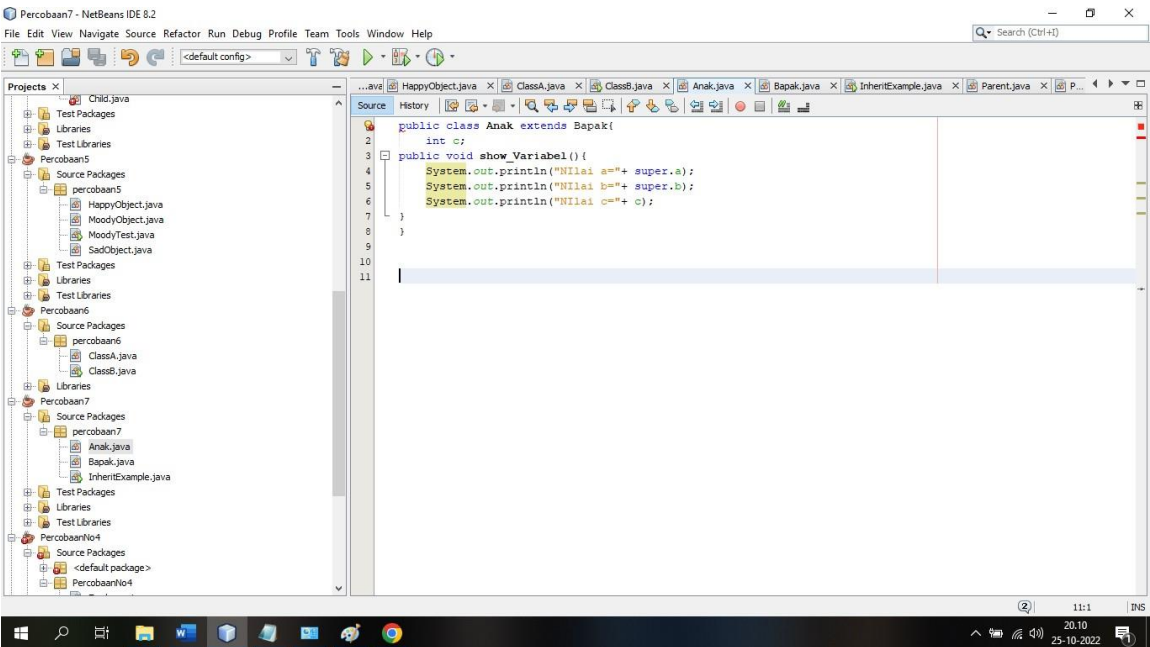
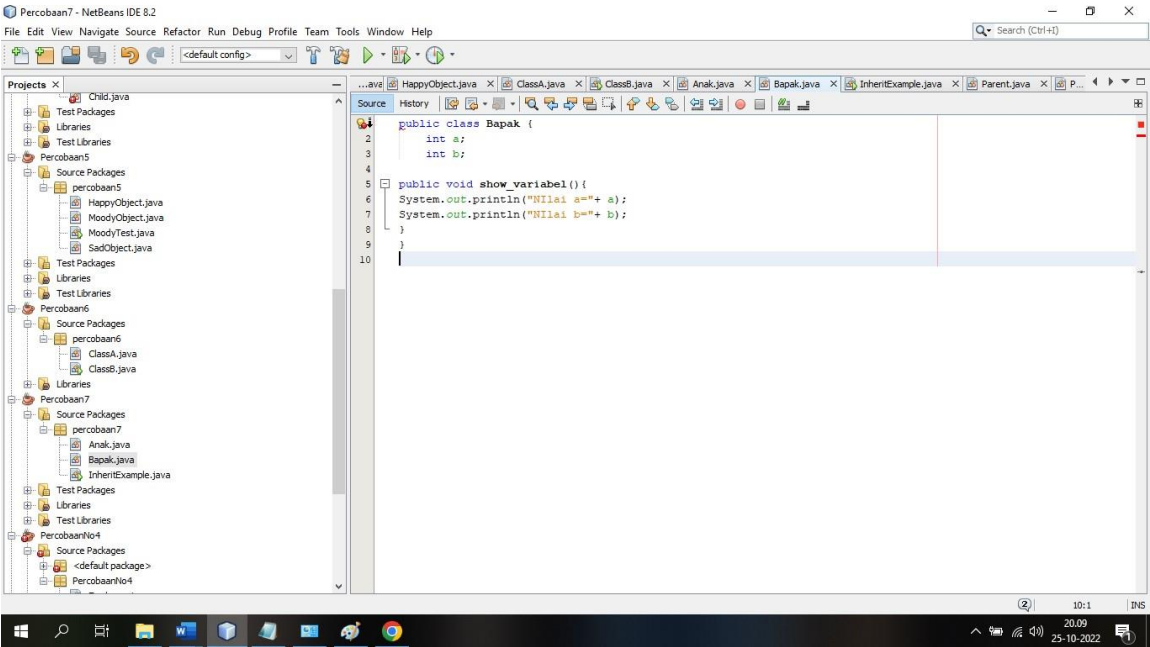
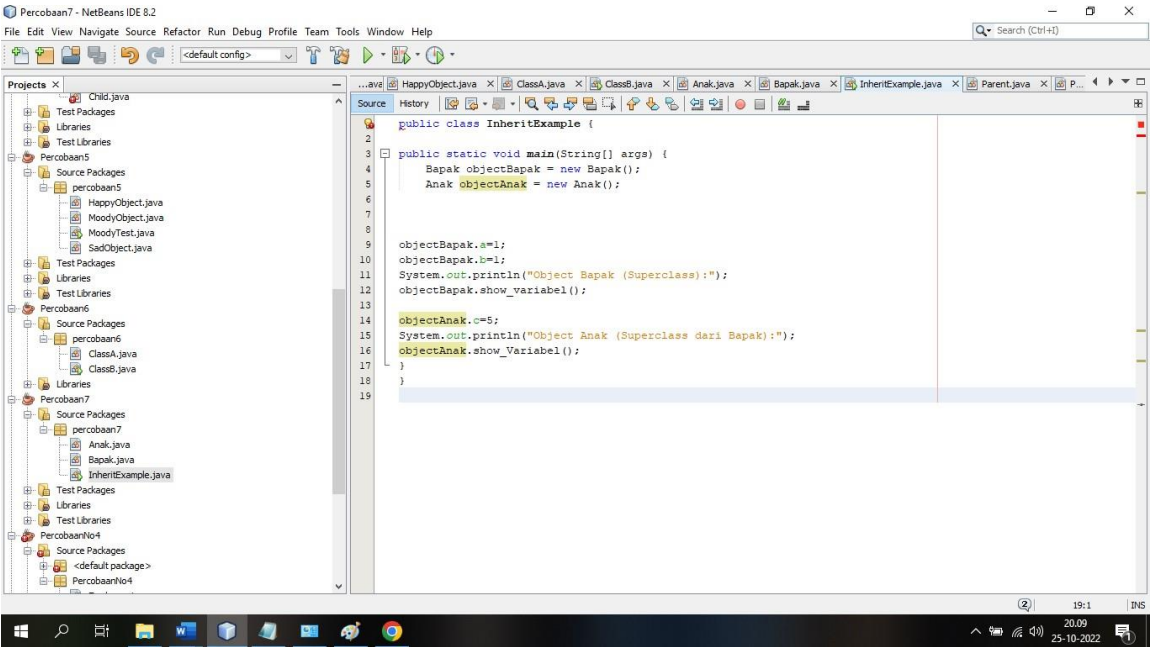


Saya memisahkan subclass B dari kelas A tanpa mengubah apapun dan Cuma sekedar drag and drop dan saya juga sudah memperhatikan konstruktornya

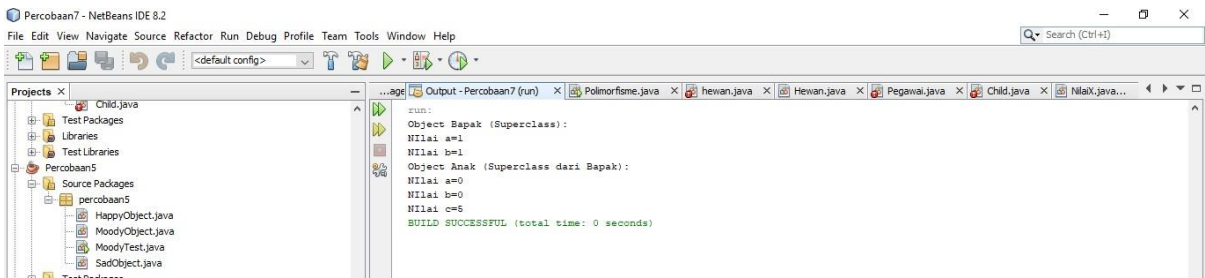
7. PERCOBAAN 7

Percobaan berikut ini menunjukkan penggunaan Inheritance dan Overriding method pada kelas Bapak dan subkelas Anak. Terjadi override pada method show_variabel. Perhatikan perubahan nilai pada variabel a, b, dan c. Kemudian lakukan modifikasi pada method show_variabel() pada class Anak. Gunakan super untuk menampilkan nilai a dan b (memanfaatkan method yang sudah ada pada superclass).

➤ INPUT PERCOBAAN 7



➤ OUTPUT PERCOBAAN 7

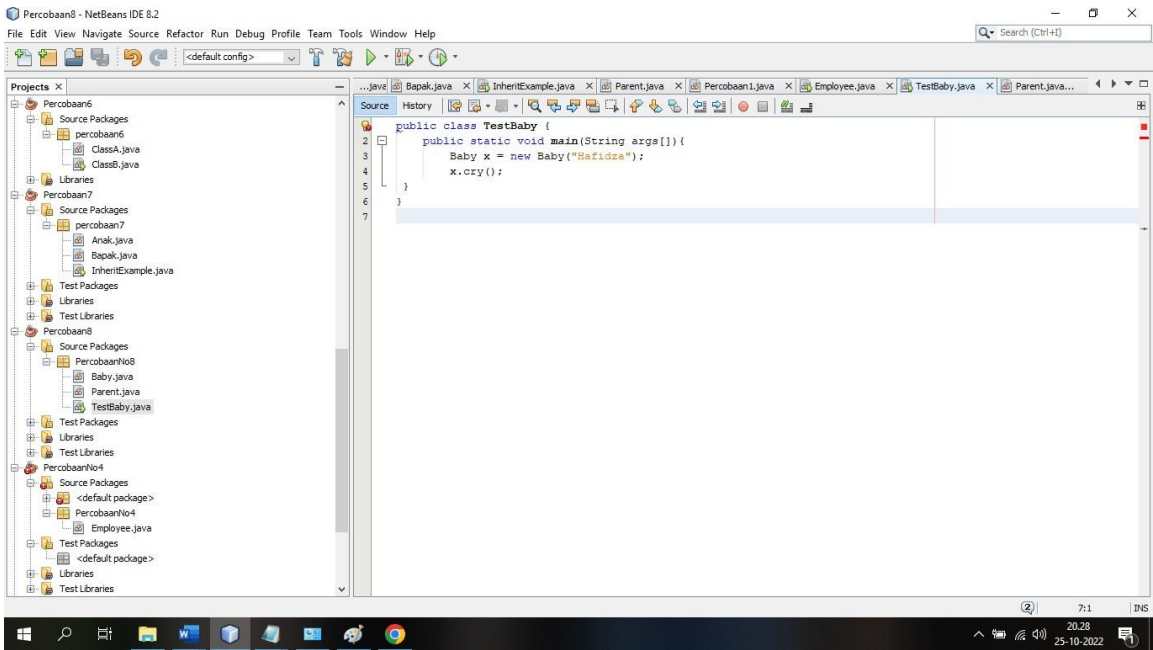


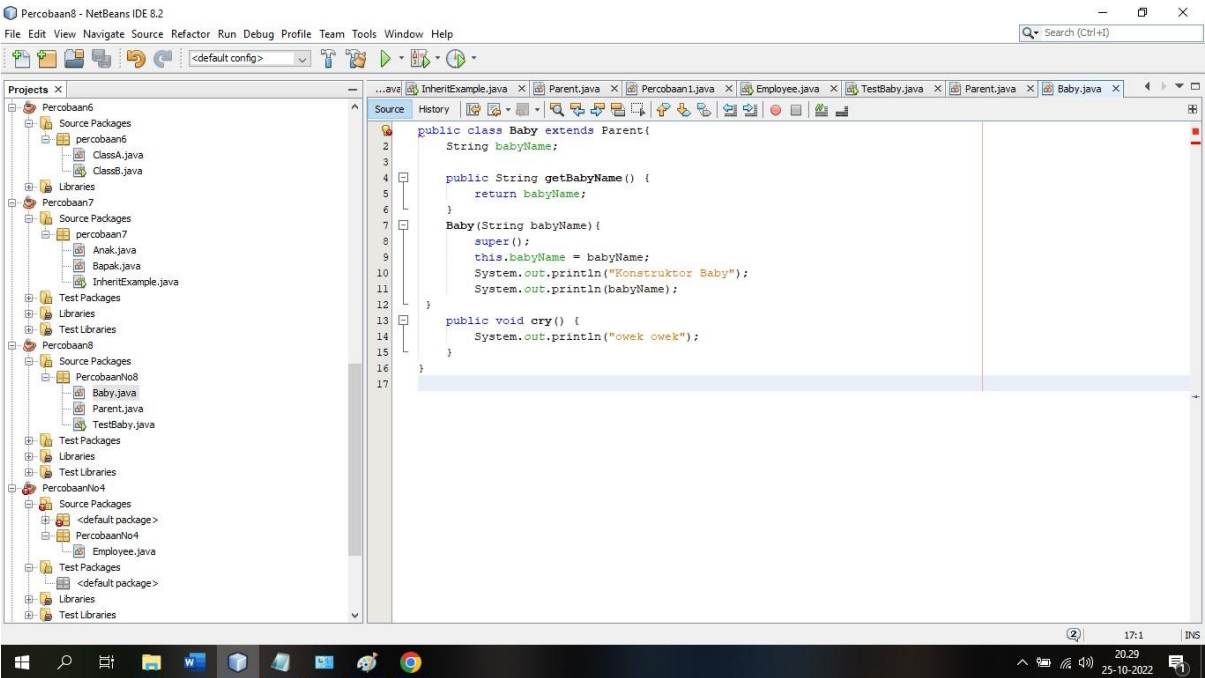
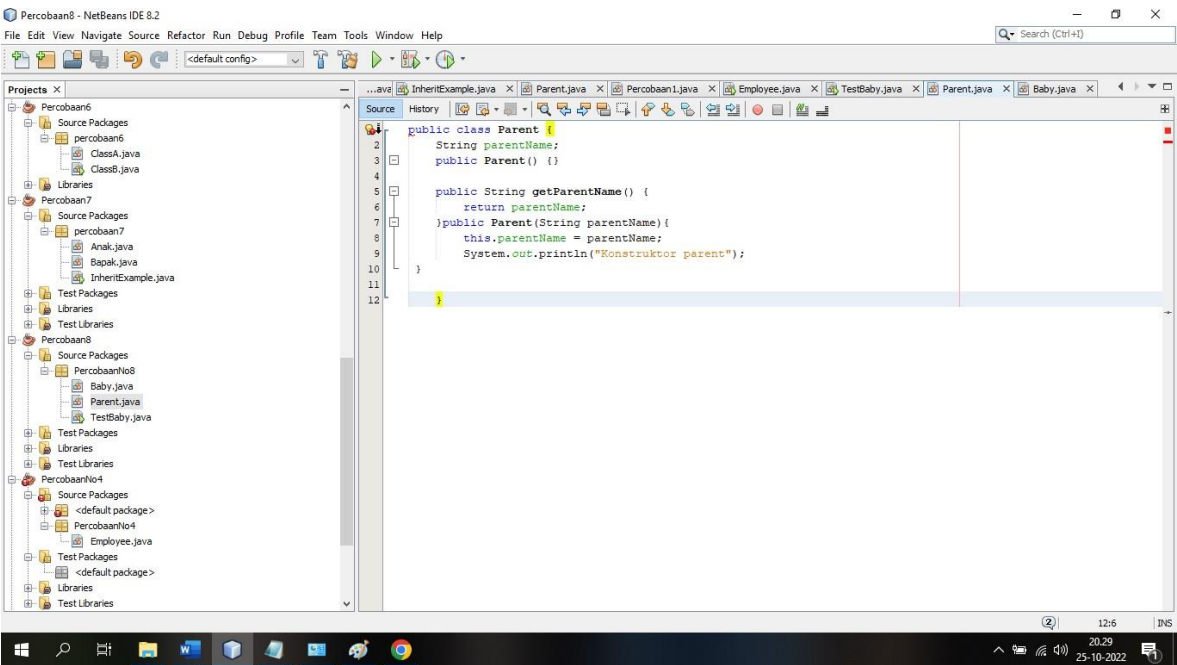
Perbedaan dengan yang sebelum diubah yaitu pada subclass anak nilai a,b yang mewarisi nilai bapak dan c yaitu nilai dari objek sianak atau buka nilai warisan.

8. PERCOBAAN 8

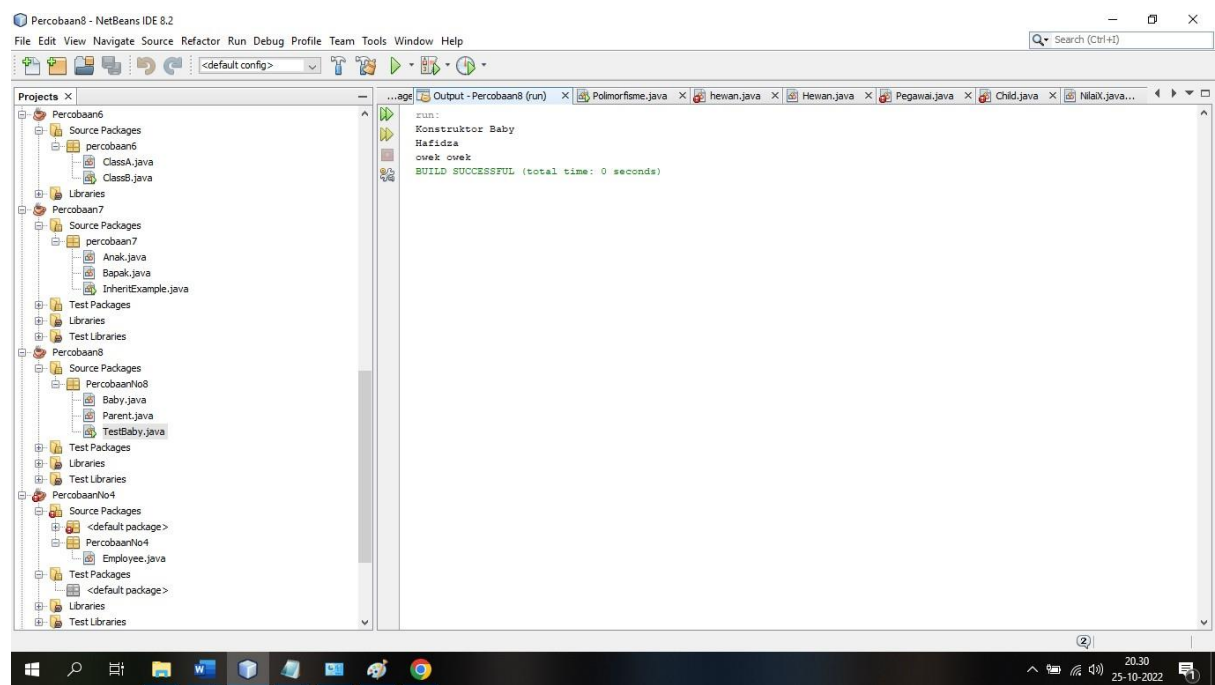
Percobaan berikut ini menunjukkan penggunaan overriding method pada kelas Parent dan subkelas Baby, saat dilakukan pemanggilan konstruktor superclass dengan menggunakan super

➤ INPUT PERCOBAAN 8





➤ OUTPUT PERCOBAAN 8



Dengan begitu programnya berjalan dengan lancar