**Eksamensrapport – PGR112 Objektorientert programmering**

I denne oppgaven har jeg jobbet med arv, innkapsling, filhåndtering og JDBC i et program som håndterer kjøretøy og scrapyards i Java. Jeg har laget en abstrakt klasse Vehicle for å samle felles egenskaper, og latt FossilCar, ElectricCar og Motorcycle arve fra den. Dette gjør det mulig å håndtere ulike kjøretøytyper på en strukturert måte.

Polymorfi er implementert ved hjelp av instanceof, slik at jeg kunne bruke spesifikke metoder for hver kjøretøytype når objektene ble lagret i databasen. For eksempel:

if (v instanceof FossilCar fc) {

stmt.setString(10, fc.getFuelType());

}

Jeg har brukt innkapsling gjennom private felter og gettere, slik at data kun er tilgjengelig gjennom kontrollerte metoder. Dette gir bedre sikkerhet og struktur i koden.

Unntakshåndtering er brukt med try-catch både ved filinnlesing og under kommunikasjon med databasen. Dette beskytter programmet mot feil som manglende linjer i filen eller tilkoblingsproblemer.

En utfordring jeg møtte var å finne ut hvor kjøretøyene startet i tekstfilen. Jeg løste dette ved å lese antall scrapyards først og deretter hoppe over tilsvarende linjer før kjøretøyene behandles. Jeg forutsatte at filen følger en fast struktur, som beskrevet i oppgaveteksten.

Jeg er spesielt fornøyd med menyprogrammet mitt, som gir brukeren mulighet til å hente data fra tre ulike tabeller. Jeg bruker PreparedStatement i alle spørringer, noe som gir både sikkerhet og oversiktlig kode.

Løsningen viser forståelse for arv og polymorfi, abstrakte klasser, innkapsling, unntakshåndtering, JDBC og filhåndtering. Jeg har også brukt en ekstern db.properties-fil for fleksibel konfigurasjon av databaseforbindelsen.

Til slutt valgte jeg å strukturere prosjektet med meningsfulle pakkenavn. Klassene Menu, Main og Importer ligger i pakken ui (user interface), mens kjøretøy og scrapyards klasser ligger i pakken vehicles. Dette gir bedre struktur og lesbarhet i prosjektet.