

Rapport – PGR112 (August 2025)

Jeg har jobbet bevisst med innkapsling i prosjektet ved å bruke private felt og public getter i modellklassene. Dette sørger for å kontrollere tilgang til objektdata, for å sikre trygg kode og enkelt vedlikehold. Et eksempel på dette er hvordan alle feltene i klassene Animal, Dog, Cat og Bird er private og derfor ikke kan endres på, mens verdiene kan leses gjennom egne metoder.

Arv er brukt i prosjektet mitt ved å la Dog, Cat og Bird arve fra superklassen Animal. Animal klassen inneholder alle felles egenskaper for de ulike dyreartene, mens sub-klassene inneholder artsspesifikke egenskaper. Så overstyrer sub-klassene getDetails(), slik at koden skriver ut en beskrivelse som er tilpasset hver art.

Det er også blitt brukt polymorfi ved å ha lister av typen List<Animal>, hvor det lages objekter av de ulike sub-klassene. Dette gjør at jeg kan behandle alle dyr likt i menyen, men samtidig få riktig oppførsel fra koden.

En utfordring jeg støtte på var usikkerhet ved AnimalRepository og hvordan den klassen fungerte. Når jeg så på kodeeksempler fra forberedelse til eksamen og forelesnings kode, skjønnte jeg ikke hvorfor jeg trengte å ha egne insert-metoder for hver art. Men med å se tilbake på kurs tatt på Udemy.com og spørre ChatGPT om en forklaring, forstod jeg at disse metodene var avgjørende for å lagre riktig data i databasen basert på dyretypen.

En annen utfordring jeg støtte på var «Feil ved insertShelter:

[SQLITE_CONSTRAINT_PRIMARYKEY] A PRIMARY KEY constraint failed (UNIQUE constraint failed: Shelter.ShelterID)» error når jeg kjørte koden i starten.

Men jeg hadde støtt på samme error under forberedelse til eksamen og hadde skrevet ned løsningen da. Og jeg måtte legge til den del som fjernet tidligere tabeller ved oppstart av koden i DatabaseConnection.java.

Jeg er veldig fornøyd med hvordan jeg har strukturert koden min, og jeg ser en tydelig forberedning fra tidligere forsøk på eksamen. Jeg er og veldig fornøyd med hvor mye forberedelse og øving på tidligere eksamens oppgaver har gitt meg mye å gå på i løsningen min basert på erfaring fra OOP.

Kildehenvisning:

Jeg har basert koden min på erfaring og kunnskap oppbygget på hva vi har lært fra undervisning, da spesielt fra Master filene for undervisning i 2023 og 2024. Og jeg har jobber med forberedelse til eksamen/personlig forbedring innen OOP med kurset Java and Object-Oriented Programming – Basic to Advanced av Dr. Shahram Taheri for oppfrisking. Punkter med dette er lagt til som kommentarer i selve prosjektet.

Jeg har og brukt verktøyet ChatGPT (GPT-4-turbo) via chat.openai.com som en samtalepartner for å tolke og dobbeltsjekke forståelsen av eksamenskravene. Dette gjaldt spesielt hvordan del 1 skulle løses, hva som kreves for å gå videre til del 2, og hvordan man kan bruke tekstfilen animals.txt riktig. Jeg brukte KI som en trygghet for å validere egne tolkninger, ikke for å få fasitsvar.

Prompt-eksempler:

- «Ok jeg trenger en liten recap for hva jeg må ha gjort i del 1?»

Svar:

Selvfølgelig! Her er en **kort, konkret og oversiktlig oppsummering** av hva du må ha gjort i **Del 1** av eksamen:

Delmål	Oppnådd når du har...
✓ Dyrehjem i databasen	Lest inn og lagret shelters fra animals.txt
✓ Dyr i databasen	Lest inn og lagret dyr fra animals.txt
✓ Korrekt objektstruktur	Brukt Animal-arv, innkapsling og polymorfi
✓ JDBC brukt riktig	Alle inserts skjer via PreparedStatement
✓ Kan hente dyr	Har getAllAnimals() som returnerer List<Animal>
✓ Testet at det funker	Ser utskrift av alle dyr i konsollen via Main.java

- «Men jeg er forvirret på hva insert gjør. Er den for å legge til dyr? Fordi det trenger jeg vel ikke? animls.txt inneholder jo dyrene for meg?»

Svar: En god forklaring på hva insert-metoden gjør og hvorfor jeg trenger den.

- «Bare så jeg er sikker på at jeg har forstått forberedelsen rett. Når jeg skal lage modellene, lager jeg en superklasse som heter Animal. Her har jeg alle de felles faktorene som id, navn, alder, vaksinert, shelterId og dyretype? Også har jeg de dyreart spesifikke i sine egen klasser?»

Svar: Ja, helt riktig forstått! 🏆

Det er akkurat slik du skal bruke **arv og innkapsling** her, og det er også det sensor forventer å se.

Alle med dato: 04.08.2025