

Kandidatnr.: 174

# PGR112 Eksamen

Vår 2024

## Rapport til eksamen

I denne rapporten vil jeg beskrive hvordan jeg håndterte unntak, innkapsling og arv i min løsning av eksamen.

### - Unntak

Unntak fungerer utmerket underveis i kodingen, ettersom det lar meg kunne mer presist finne ut av hvor det kan ha gått feil i koden. For hver main-metode, og metoder passet jeg på å forsøke å innpasse et unntak inn hvor jeg kunne ved senere anledning kunne se hvor det hadde gått galt. Slik jeg har forstått konseptet er det også utmerket å bruke for å unngå at programmet kræsjer, som jeg slettes ikke ville ha ønsket skulle skje under eksamen! Jeg sørget derfor for å bruke try catch på de fleste metoder hvor tilkoblinger til databasen var nødvendig. I tillegg vil noen deler av begynnelsen av Database-klassen være hovedsaklig lånt fra en database undervisning til Marcus, der jeg har kildeført dette, og han har nevnt det var greit vi fikk bruke store deler av denne.

### - Innkapsling

Ved bruk av innkapsuleringer vil vi sikre objekters tilstand slik at de ikke vilkårlig endres fra utsiden av. Jeg brukte derfor private felter i klassene person, museum, funn, mynt, smykke og vaapen. Videre sørget jeg for å bruke private klasse innen database klassen, der metoden Properties properties kun nås og endres gjennom konstruktøren og metoder innenfor 'database'-klassen.

### - Arv

Jeg henvender meg til bruk av arv i klassene Mynt, Smykke og Vaapen der disse arves av Funn klassen. Arv fungerer slik at vi kan arve felt og metoder fra den eksisterende klassen. Vi kan dermed låne feltene fra superklassen, ved å sette de arvede feltene inn i en super(). Et resultat av disse arvede feltene resulterer i mindre dupliserende kode.