

Arbeidskrav 1 PGR101, V18

Frist: 21. mars

Du kan selv velge om du vil levere et BlueJ-prosjekt eller på et annet format. Alle kildefiler må naturlig nok være med. Hvis du leverer noe annet enn et BlueJ-prosjekt: Vennligst legg med en README-fil om hvordan løsningen din kan kjøres/testes.

Husk at formålet med dette arbeidskravet er å lære. Hvis du sliter med å løse den, så jobb videre! Hvis du ikke kommer i mål med løsningen, så gjelder samme regler som i forrige semester: Fortell oss hvordan du har jobbet (hvor mange timer, alene/sammen med andre, hvilke kilder har du brukt), og hva du trenger av hjelp for å komme videre.

- a) En skole trenger et system for å holde orden på samlingen av ulike måleinstrumenter som brukes i naturfagundervisningen. Eksempler på måleinstrumenter er termometer, klokke og vekt.

Om alle måleinstrumentene skal det være registrert et entydig registreringsnummer (f.eks. K1234), en opplysning om instrumentet er i orden eller ikke, og en plasseringskode (hvor instrumentet kan finnes, f.eks. R24H3 – rom 24, hylle 3).

Om et termometer skal det i tillegg være opplysning om laveste og høyeste temperatur som termometeret kan måle (i °C, desimaltall), om en klokke opplysning om minste tidsintervall som klokken kan måle (i sekunder, desimaltall), og om en vekt opplysning om minste og største vekt som vekten kan måle (i gram, desimaltall).

Med utgangspunkt i typene termometer, klokke og vekt skal klassene `Thermometer`, `Clock` og `Weight` skrives. Disse skal være subklasser av en klasse `Meter` (måleinstrument).

Klassene har standard parametrisk og ikke-parametrisk konstruktør i tillegg til standard tilgangsmetoder, `toString`-metode og passende `equals`-metode.

- b) Skriv en klasse `MeterArchive` med metoder for å legge til, hente, endre og slette måleinstrumenter. Klassen bruker en passende `List` for å holde på instrumentene.

Metoden for å legge til instrument skal ha en referanse til et måleinstrument som parameter.

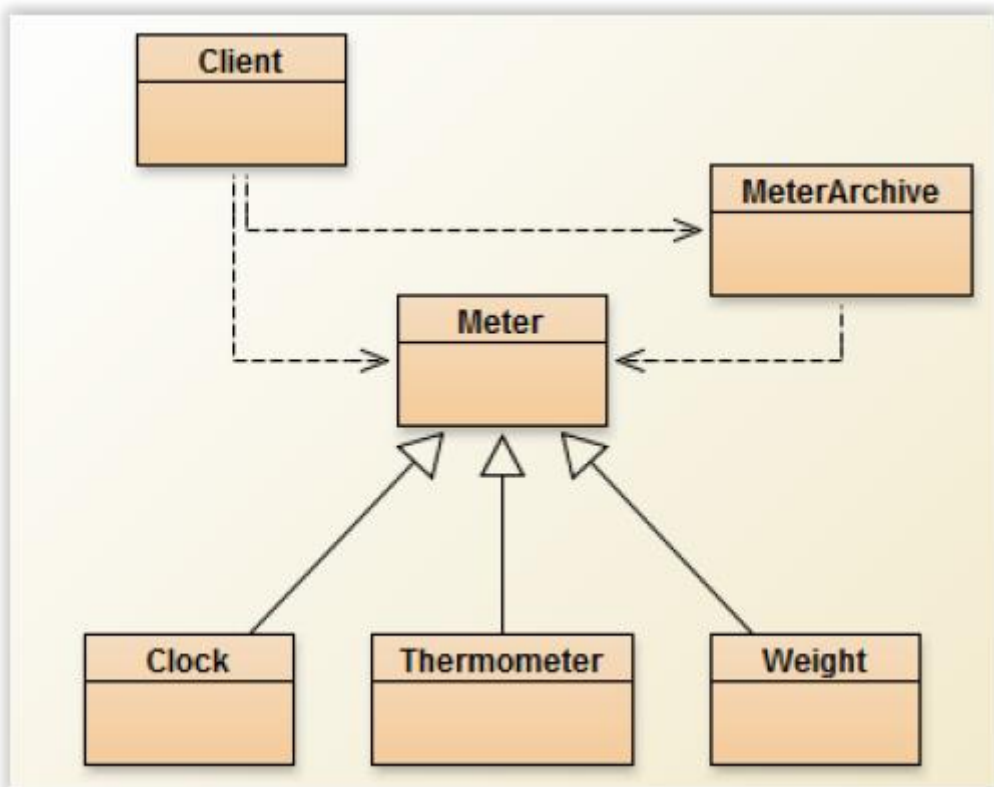
Metoden for å hente og slette et instrument skal ha et registreringsnummer som parameter. Metoden for å hente returnerer `null` hvis instrumentet ikke finnes, og metoden for å slette returnerer `true` hvis slettingen gikk å utføre.

Klassen skal ha to metoder for å endre et instrument. Den ene har en `String` for registreringsnummer og en ny plassering som parameter. Den andre har en `String` for registreringsnummer som parameter og skal endre instrumentet slik at det blir registret som ikke i orden. Begge metodene skal returnere `true` hvis endringen ble utført.

Klassen skal også ha en metode som returnerer alle instrumentene på en passende måte.

- c) Skriv en klasse `Client` med en metode `mainMethod`. Denne metoden oppretter et objekt av klassen `MeterArchive` og bruker dette til å registrere noen måleinstrumenter, og bruker `MeterArchive` sine metoder beskrevet over på *riktig* måte.

Neste side viser et enkelt klassediagram for løsningen.



Din løsning skal vise bruk av alle metodene i `MeterArchive`.

Neste side viser eksempel på utskrift som `mainMethod` kan lage.

Hvis noe er uklart, gjør du egne begrunnede forutsetninger for å løse oppgaven.

Mulig utvidelser (for de som ønsker større utfordringer – dere har ikke nødvendigvis lært det som trengs for å løse disse...):

- Opprett en tekstfil på et egnet format (gjærne `json`, men kan også være et annet format). La programmet ditt lese fra denne filen når `Client` opprettes slik at informasjon om alle instrumentene leses fra filen. Du kan godt bruke dataene på neste side.
- Lag automatiserte tester for metodene dine (særlig de i `MeterArchive`).

Oversikt over alle instrumenter:

Vekt

(Måleintervall: 0.01 - 10.0)

Regnummer: V1000

Plassering: R101H1

Status: i orden

Vekt

(Måleintervall: 0.1 - 100.0)

Regnummer: V1001

Plassering: R101H2

Status: i orden

Vekt

(Måleintervall: 1.0 - 1000.0)

Regnummer: V1002

Plassering: R101H3

Status: i orden

Termometer

(Min- og maks temp: 0.01 - 100.0)

Regnummer: T2000

Plassering: R101H4

Status: i orden

Termometer

(Min- og maks temp: 1.0 - 200.0)

Regnummer: T2001

Plassering: R101H5

Status: i orden

Klokke

(Minste tidsintervall: 1.0E-4 s)

Regnummer: K3000

Plassering: R101H6

Status: i orden

Klokke

(Minste tidsintervall: 0.01 s)

Regnummer: K3001

Plassering: R101H7

Status: i orden

Klokke

(Minste tidsintervall: 1.0 s)

Regnummer: K3002

Plassering: R101H8

Status: i uorden

Sletter instrument med regnummer : T2007

Ukjent regnummer!

Endrer posisjon for instrument med regnummer : T2001

Endret!

Oversikt over alle instrumenter nå:

Vekt

(Måleintervall: 0.01 - 10.0)

Regnummer: V1000

Plassering: R101H1

Status: i orden

Vekt

(Måleintervall: 0.1 - 100.0)

Regnummer: V1001

Plassering: R101H2

Status: i orden

Vekt

(Måleintervall: 1.0 - 1000.0)

Regnummer: V1002

Plassering: R101H3

Status: i orden

Termometer

(Min- og maks temp: 0.01 - 100.0)

Regnummer: T2000
Plassering: R101H4
Status: i orden
Termometer
(Min- og maks temp: 1.0 - 200.0)
Regnummer: T2001
Plassering: R101H6
Status: i orden
Klokke
(Minste tidsintervall: 1.0E-4 s)
Regnummer: K3000
Plassering: R101H6
Status: i orden
Klokke
(Minste tidsintervall: 0.01 s)
Regnummer: K3001
Plassering: R101H7
Status: i orden
Klokke
(Minste tidsintervall: 1.0 s)
Regnummer: K3002
Plassering: R101H8
Status: i uorden
Henter instrument med regnummer T2001:
Termometer
(Min- og maks temp: 1.0 - 200.0)
Regnummer: T2001
Plassering: R101H6
Status: i orden
Setter instrument med regnummer T2001 til ikke i orden:
Termometer
(Min- og maks temp: 1.0 - 200.0)
Regnummer: T2001
Plassering: R101H6
Status: i uorden