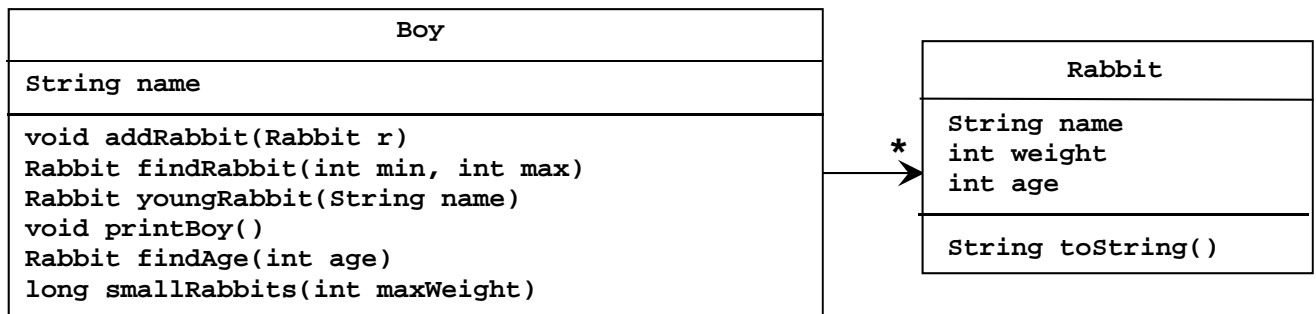


# Rabbit



1. Opret en klasse, *Rabbit*, hvis objekter repræsenterer en kanin. Klassen er specificeret til højre i UML-diagrammet. Feltvariablerne skal initialiseres i en konstruktør (via parametre af passende type). Metoden *toString* skal returnere en tekststreng på formen:

**"Nusse (4 years old - 5 kg) ".**

2. Lav en *TestDriver*-klasse med en klassemetode *test*. Metoden returnerer intet og har ingen parametre.
3. Opret fem velvalgte *Rabbit*-objekter i *test*-metoden, via objektreferencer *r1*, *r2*, *r3*, *r4* og *r5*, og udskriv disse ved hjælp af *toString*-metoden.

**Tilkald en instruktør og demonstrer det, som du har lavet indtil nu (T1).**

4. Opret en ny klasse, *Boy*, hvis objekter repræsenterer en dreng, der har nogle kaniner. Klassen og dens relation til *Rabbit*-klassen, er specificeret i ovenstående UML-diagram.
5. Programmér metoden *addRabbit*, der tilføjer *Rabbit*-objektet *r* til *Boy*-objektet.
6. Opret et objekt af typen *Boy* i *test*-metoden i *TestDriver*-klassen og knyt de allerede oprettede *Rabbit*-objekter hertil.
7. Programmér metoden *findRabbit*. Metoden skal returnere en kanin, hvis vægt ligger mellem de to parametre (begge inklusive). Det kan antages, at den anden parameter er større end eller lig den første. Udvid *Rabbit*-klassen med de nødvendige accessormetoder.
8. Afprøv den skrevne metode i *test*-metoden i *TestDriver*-klassen.

**Tilkald en instruktør og demonstrer det, som du har lavet siden forrige tjekpunkt (T2).**

9. Programmér metoden *youngRabbit*. Metoden skal returnere den yngste kanin, der har det angivne navn. Afprøv den skrevne metode i *test*-metoden.

**Tilkald en instruktør og demonstrer det, som du har lavet siden forrige tjekpunkt (T3).**

10. Programmér metoden *printBoy*. Metoden skal udskrive drengens navn efterfulgt af alle kaniner sorteret efter alder (lavest til højest). Hvis to kaniner har samme alder, sorteres alfabetisk efter navn. Afprøv den skrevne metode i *test*-metoden.

**Tilkald en instruktør og demonstrer det, som du har lavet siden forrige tjekpunkt (T4).**

11. Brug funktionel programmering til at implementere metoden *findAge*. Metoden skal returnere en kanin, der har den angivne alder. Afprøv den skrevne metode i *test*-metoden.

**Tilkald en instruktør og demonstrer det, som du har lavet siden forrige tjekpunkt (T5).**

12. Brug funktionel programmering til at implementere metoden *smallRabbits*. Metoden skal returnere antallet af kaniner, der højst har den angivne vægt. Afprøv den skrevne metode i *test*-metoden.

**Tilkald en instruktør og demonstrer det, som du har lavet siden forrige tjekpunkt (T6).**