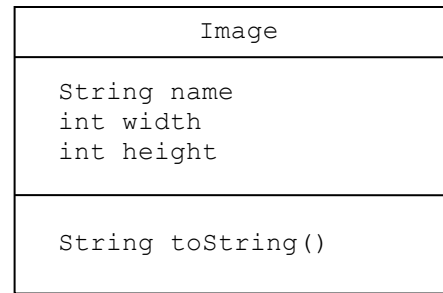


Image

1. Opret en klasse, *Image*, der repræsenterer et billede. Klassen er specificeret i UML-diagrammet til højre. De tre feltvariabler skal initialiseres i en konstruktør (via parametre af passende type). Metoden *toString* skal returnere en streng-repræsentation for en *Image* på formen

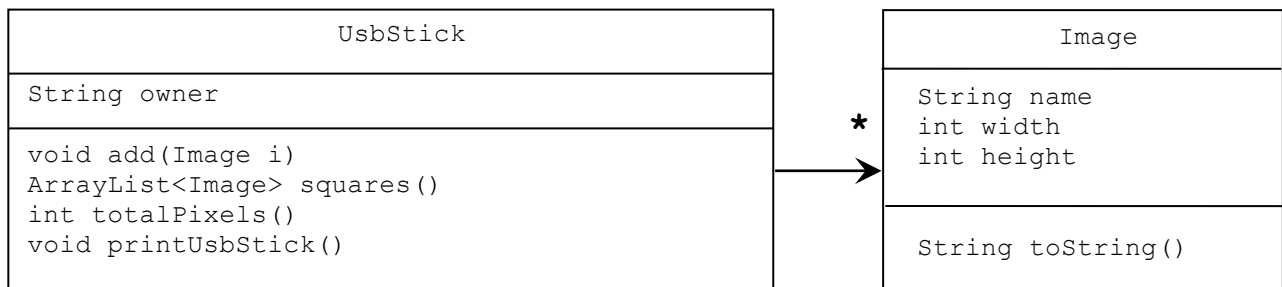
```
"Beautiful lake: 124 x 64 pixels"
"Portrait: 80 x 80 pixels"
```



2. Lav en *Driver*-klasse med en *exam*-metode. Metoden skal være static, have returtype void og være uden parametre.
3. Opret fem velvalgte *Image*-objekter i *exam*-metoden, via objektreferencer *i1*, *i2*, *i3*, *i4* og *i5*, og udskriv disse vha. *toString*-metoden.

Tilkald tilsynsførende og demonstrer det du har lavet indtil nu.

4. Opret en ny klasse, *UsbStick*, der repræsenterer et USB-stik med billeder. Klassen *UsbStick*, og dens relation til klassen *Image*, er specificeret i følgende UML-diagram:



5. Programmér metoden *add*, der tilføjer *Image*-objektet *i* til *UsbStick*-objektet.
6. Opret et objekt af typen *UsbStick* i *exam*-metoden i *Driver*-klassen og knyt de allerede oprettede *Image*-objekter hertil.
7. Programmér metoden *squares*. Metoden skal returnere alle kvadratiske billede. Udvid *Image*-klassen med de nødvendige get-metoder.
8. Afprøv metoden *squares* i *exam*-metoden i *Driver*-klassen.

Tilkald tilsynsførende og demonstrer det du har lavet indtil nu.

9. Programmér metoden *totalPixels*. Metoden skal returnere billedernes samlede antal pixels. Afprøv *totalPixels* i *exam*-metoden.
10. Programmér metoden *printUsbStick*. Metoden skal udskrive ejeren af USB-stikket efterfulgt af alle billeder sorteret efter bredde (lavest til højest). Hvis to har samme bredde sorteres efter højde (lavest til højest). Afprøv *printUsbStick* i *exam*-metoden.

Tilkald tilsynsførende og demonstrer din færdige løsning.