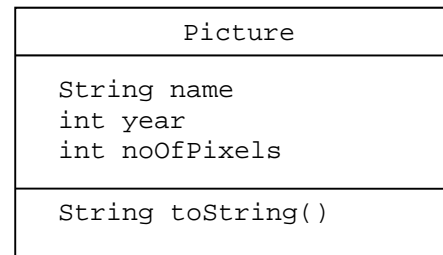


# Picture

1. Opret en klasse, *Picture*, 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 *Picture* på formen

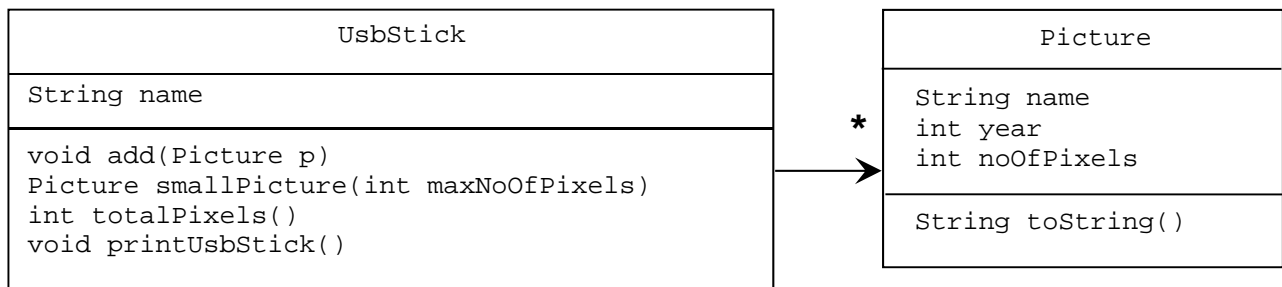
"Beautiful lake from 2017 with 120000 pixels"



2. Lav en *Driver*-klasse med en *exam*-metode. Metoden skal være static, have returtype void og være uden parametre.
3. Opret fem velvalgte *Picture*-objekter i *exam*-metoden, via objektreferencer *p1*, *p2*, *p3*, *p4* og *p5*, og udskriv disse vha. *toString*-metoden.

**Tilkald en instruktør 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 *Picture*, er specificeret i følgende UML-diagram:



5. Programmér metoden *add*, der tilføjer *Picture*-objektet *p* til *UsbStick*-objektet.
6. Opret et objekt af typen *UsbStick* i *exam*-metoden i *Driver*-klassen og knyt de allerede oprettede *Picture*-objekter hertil.
7. Programmér metoden *smallPicture*. Metoden skal returnere et billede, der højst har det angivne antal pixels. Hvis der ikke findes et sådant billede returneres null. Udvid *Picture*-klassen med de nødvendige get-metoder.
8. Afprøv metoden *smallPicture* i *exam*-metoden i *Driver*-klassen.

**Tilkald en instruktør og demonstrer det du har lavet indtil nu.**

9. Programmér metoden *totalPixels*. Metoden skal returnere billedernes samlede antal pixels. Afprøv *totalPixels* i *exam*-metoden.

**Tilkald en instruktør og demonstrer det du har lavet indtil nu.**

10. Programmér metoden *printUsbStick*. Metoden skal udskrive USB-stikkets navn efterfulgt af alle billeder sorteret efter år (lavest til højest). Hvis to billeder er fra samme år sorteres efter antal pixels (højest til lavest). Afprøv *printUsbStick* i *exam*-metoden.

**Tilkald en instruktør og demonstrer din færdige løsning.**