**Terningspil**

Lav nu en ny klasse med navnet **Terning** og gør følgende med klassen:

1. Lav to attributter, som skal være af typen **int**. Attributterne skal hedde **MAX** og **faceValue**. MAX skal tilskrives værdien 6.  
   Lav en tredje attribut med typen **Random** (benyt Random klassen), og giv den et hvilket som helst sigende varibel navn. Ligesom faceValue, tilskrives denne ikke nogen værdi.
2. Lav en konstruktør, som ikke modtager noget (default constructor), og som giver et fremtidigt terninge objekt en tilfældig/random værdi til at starte med. (Hint: Du skal bruge et Random objekt og faceValue)
3. Lav en metode med navnet **roll**(skrives såldes *void roll()*), som ændrer faceValue attributten til en ny random værdi.

Lav en klasse, som hedder **Terningspil** og lav en main-metode hvori du gør følgende:

* + 1. Lav to instanser af klassen **Terning**, du kan kalde dem f.eks. **terning1** og **terning2**.
    2. Prøv at brug **roll**-metoden på de to Terninge objekter og skriv derefter værdien af deres nye faceValue ud. Sikre dig at terningerne opfører sig efter hensigten.

Nu skal du være kreativ! Der findes et terningspil som går ud på at en spiller skal slå to 1’ere for at vinde spillet. Som et sejers brøl råber spilleren ”Snake eyes!” hvorefter spillet officielt afsluttes.

Ved brug af de ting du allerede kender fra de sidste ugers kodning og Regnemaskine-opgaven, skal du prøve at implementere dette scenarie i main-metoden. Prøv endelig lidt forskelligt af. Spillets udførelse kunne se sådan her ud i konsollen.

A black screen with white text

Description automatically generated

Hvis du har ekstra tid kan du tænke over andre terningespil du vil kunne implementere i din løsning.