

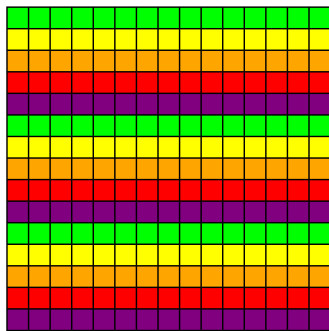
## W3 - L2 Programma

### Nabespreking voorbereiding

### Opgaven

#### ... en nog eens vierkantjes

In opgave Vierkantjes, vierkantjes heb je een raster van rode vierkantjes gemaakt. Pas die code zodanig aan dat er 100 bij 100 vierkantjes worden getekend en er een patroon aan kleuren ontstaat zoals aangegeven in de afbeelding.



- rij nummer (de bovenste rij is rij 0) exact deelbaar door 5: GROEN
- Bij rest 1: GEEL
- Bij rest 2: ORANJE
- Bij rest 3: ROOD
- Bij rest 4: PAARS

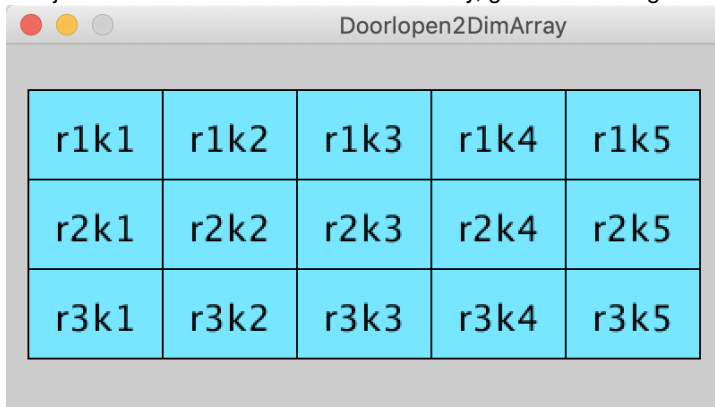
#### Methoden met conditie

1. Schrijf de methode `showMuisheft()`. De methode schrijft een tekst in het console die afhankelijk is van de positie van de muis van de gebruiker. Als de muis zich rechts van de middenlijn van het grafische scherm bevindt, toont het console "Muis bevindt zich RECHTS van het scherm", anders "Muis bevindt zich LINKS van het scherm". Bedenk zelf welke parameters de methode nodig heeft. TIP. Maak gebruik van de methode `isMuisOpRechterHelft` uit de vorige les.

- Schrijf de methode `geefQuotientAlsString()` en test deze. De methode berekent de quotiënt van twee getallen en retourneert het resultaat als een `String`. De deling moet werken voor integers en kommagetallen. Bedenk zelf welke parameters je zou moeten definiëren. TIP: Delen door 0 levert een foutmelding op. Dat zul je moeten oplossen.

### Doorlopen van een twee-dimensionale array

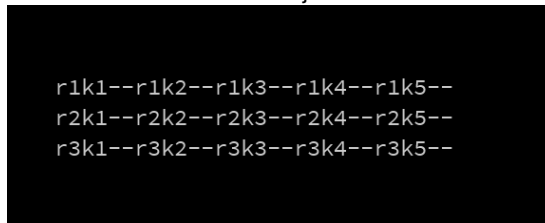
Bekijk onderstaand twee-dimensionaal array, getoond in het grafische scherm van processing.



r1k1	r1k2	r1k3	r1k4	r1k5
r2k1	r2k2	r2k3	r2k4	r2k5
r3k1	r3k2	r3k3	r3k4	r3k5

Implementeer het volgende:

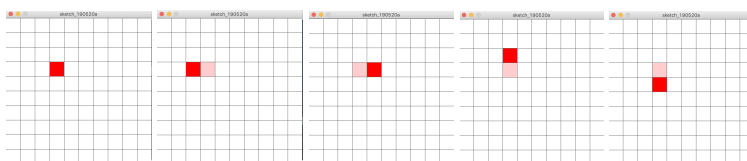
- Maak een functie waarmee je in het console de inhoud van deze array laat zien.



```
r1k1--r1k2--r1k3--r1k4--r1k5--
r2k1--r2k2--r2k3--r2k4--r2k5--
r3k1--r3k2--r3k3--r3k4--r3k5--
```

- Maak daarna een functie waarmee bovenstaande figuur in het grafische scherm laat zien.
- Pas de functie onder 2 zodanig aan dat altijd de laatste kolom een gele achtergrond krijgt.

### Industriële Robot I



Tijdens de lessen gaan we stapsgewijs een applicatie maken waarin je een robot bestuurt die objecten kan oppakken en neerzetten. Daarvoor zetten we nu de eerste stap. Bestudeer bovenstaande afbeeldingen. Het rode vierkant hierboven is de transportrobot dat zich over een grid beweegt. Links staat zijn originele positie. Vervolgens staan er achtereenvolgens de posities afgebeeld van de robot nadat die respectievelijk een stap naar links, recht, boven en beneden gedaan heeft.

Schrijf het algoritme voor het besturen van de robot met de pijltjestoetsen.